IBM Surveillance Insight for Financial Services
Version 2.0.2

IBM Surveillance Insight for Financial Services Installation Guide

Note

Before using this information and the product it supports, read the information in “Notices” on page 59.

Product Information

This document applies to Version 2.0.2 and may also apply to subsequent releases.

Contents

Introduction

Use IBM® Surveillance Insight® for Financial Services to proactively detect, profile, and prioritize non-compliant behavior in financial organizations. The solution ingests unstructured and structured data, such as trade, electronic communication, and voice data, to flag risky behavior. Surveillance Insights helps you investigate sophisticated misconduct faster by prioritizing alerts and reducing false positives, and reduces the cost of misconduct.

Some of the key problems that financial firms face in terms of compliance misconduct include:

- Fraudsters using sophisticated techniques thereby making it hard to detect misconduct.
- Monitoring and profiling are hard to do proactively and efficiently with constantly changing regulatory compliance norms.
- A high rate of false positives increases the operational costs of alert management and investigations.
- Siloed solutions make fraud identification difficult and delayed.

IBM Surveillance Insight for Financial Services addresses these problems by:

- Leveraging key innovative technologies, such as behavior analysis and machine learning, to proactively identify abnormalities and potential misconduct without pre-defined rules.
- Using evidence-based reasoning that aids streamlined investigations.
- Using risk-based alerting that reduces false positives and negatives and improves the efficiency of investigations.
- Combining structured and unstructured data from different siloed systems into a single platform to perform analytics.

IBM Surveillance Insight for Financial Services takes a holistic approach to risk detection and reporting. It combines structured data such as stock market data (trade data) with unstructured data such as electronic emails and voice data, and it uses this data to perform behavior analysis and anomaly detection by using machine learning and natural language processing.

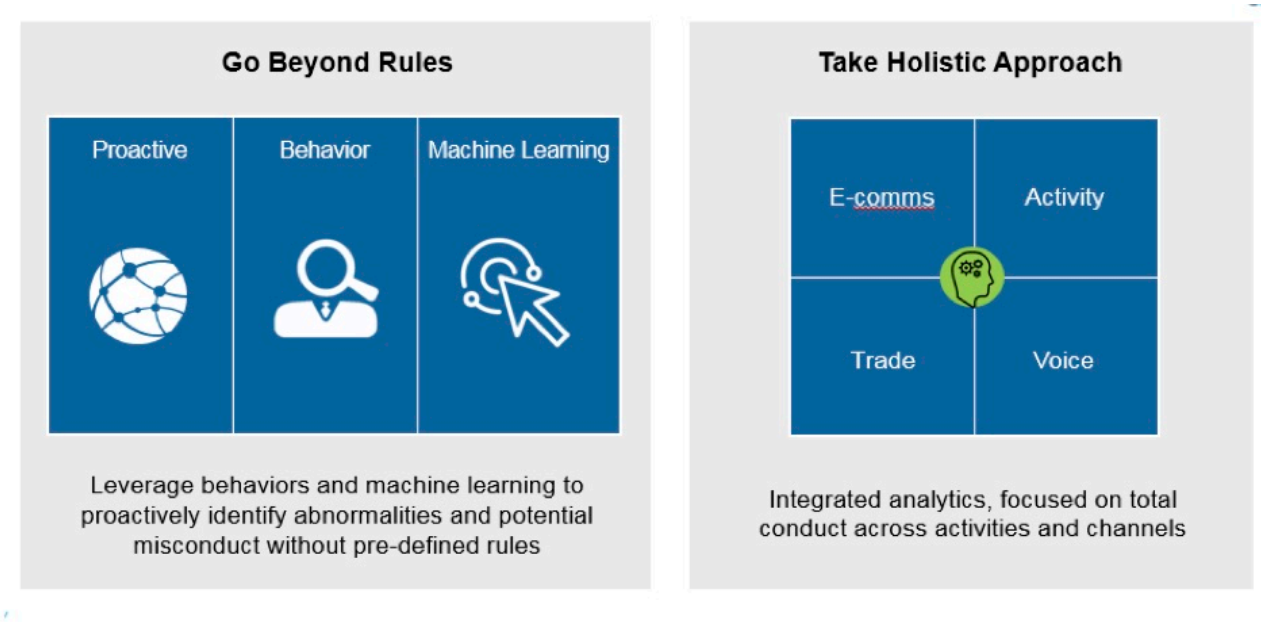


Figure 1: Surveillance Insight overview

Audience

This guide is intended for administrators and users of the IBM Surveillance Insight for Financial Services solution. It provides information on installation and configuration of the solution, and information about using the solution.

Finding information and getting help

To find product documentation on the web, access IBM Knowledge Center (www.ibm.com/support/knowledgecenter/SSWTQQ).

Accessibility features

Accessibility features help users who have a physical disability, such as restricted mobility or limited vision, to use information technology products. Some of the components included in the IBM Surveillance Insight for Financial Services have accessibility features. For more information, see Appendix A, “Accessibility features,” on page 55.

The HTML documentation has accessibility features. PDF documents are supplemental and, as such, include no added accessibility features.

Forward-looking statements

This documentation describes the current functionality of the product. References to items that are not currently available may be included. No implication of any future availability should be inferred. Any such references are not a commitment, promise, or legal obligation to deliver any material, code, or functionality. The development, release, and timing of features or functionality remain at the sole discretion of IBM.

Samples disclaimer

Sample files may contain fictional data manually or machine generated, factual data that is compiled from academic or public sources, or data that is used with permission of the copyright holder, for use as sample data to develop sample applications. Product names that are referenced may be the trademarks of their respective owners. Unauthorized duplication is prohibited.

Chapter 1. IBM Surveillance Insight for Financial Services

IBM Surveillance Insight for Financial Services provides you with the capabilities to meet regulatory obligations by proactively monitoring vast volumes of data for incriminating evidence of rogue trading or other wrong-doing through a cognitive and holistic solution for monitoring all trading-related activities. The solution improves current surveillance process results and delivers greater efficiency and accuracy to bring the power of cognitive analysis to the financial services industry.

The following diagram shows the high-level IBM Surveillance Insight for Financial Services process.

Figure 2: High-level process

1. As a first step in the process, data from electronic communications (such as email and chat), voice data, and structured stock market data are ingested into IBM Surveillance Insight for Financial Services for analysis.
2. The data is analyzed.
3. The results of the analysis are risk indicators with specific scores.
4. The evidences and their scores are used by the inference engine to generate a consolidated score. This score indicates whether an alert needs to be created for the current set of risk evidences. If needed, an alert is generated and associated with the related parties and stock market tickers.
5. The alerts and the related evidences that are collected as part of the analysis can be viewed in the IBM Surveillance Insight for Financial Services Workbench.

After the alerts are created and the evidences are collected, the remaining steps in the process are completed outside of IBM Surveillance Insight for Financial Services. For example, case investigators must work on the alerts and confirm or reject them, and then investigation reports must be sent out to the regulatory bodies as is required by compliance norms.

The solution architecture

IBM Surveillance Insight for Financial Services is a layered architecture is made up of several components.

The following diagram shows the different layers that make up the product:

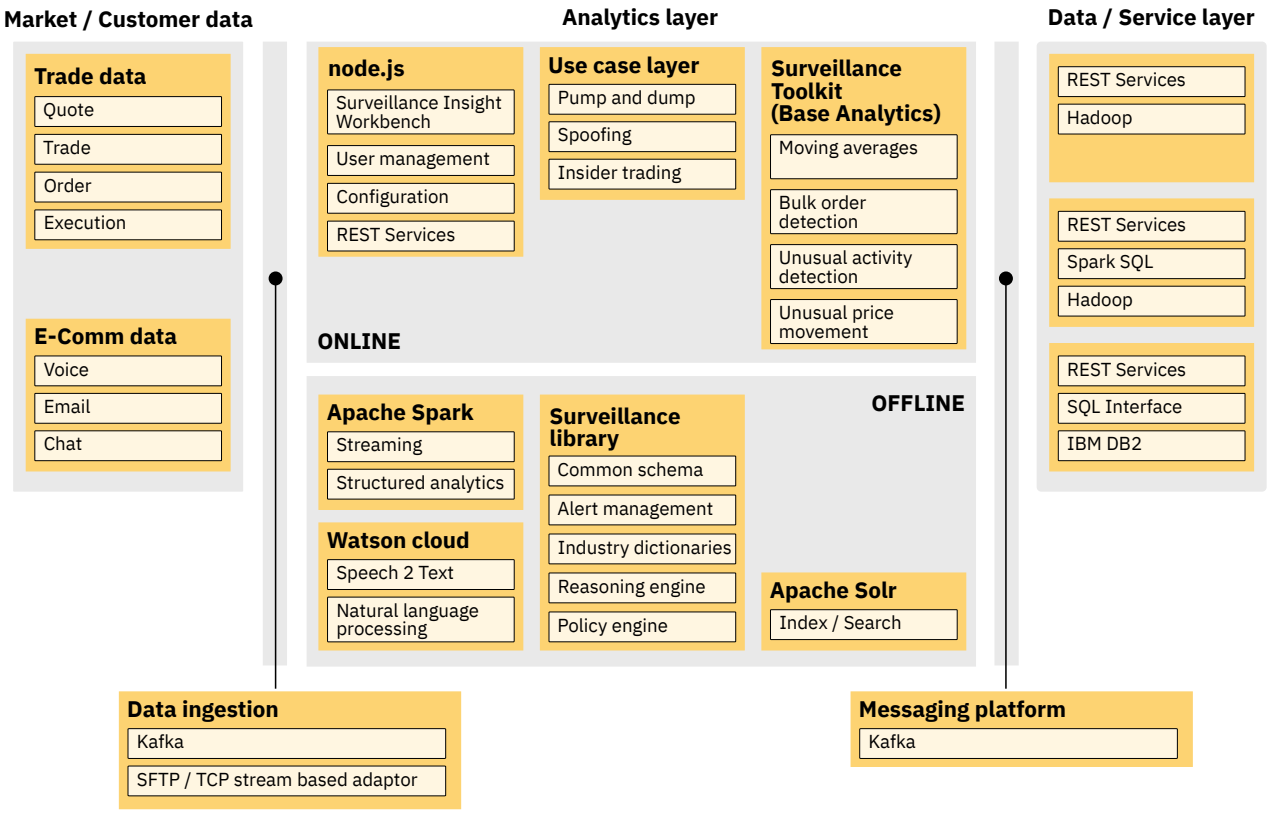


Figure 3: Product layers

- The data layer shows the various types of structured and unstructured data that is consumed by the product.
- The data ingestion layer contains the FTP/TCP-based adaptor that is used to load data into Hadoop. The Kafka messaging system is used for loading e-communications into the system.

  Note: IBM Surveillance Insight for Financial Services does not provide the adaptors with the product.
- The analytics layer contains the following components:
  - The Workbench components and the supporting REST services for the user interfaces.
  - Specific use case implementations that leverage the base toolkit operators.
  - The surveillance library that contains the common components that provide core platform capabilities such as alert management, reasoning, and the policy engine.
  - The Spark Streaming API is used by Spark jobs as part of the use case implementations.
  - Speech 2 Text and the NLP APIs are used in voice surveillance and eComms surveillance.
  - Solr is used to index content to enable search capabilities in the Workbench.
- Kafka is used as an integration component in the use case implementations and to enable asynchronous communication between the Streams jobs and the Spark jobs.
- The data layer primarily consists of data in Hadoop and IBM DB2®. The day-to-day market data is stored in Hadoop. It is accessed by using the spark-sql or spark-graphx APIs. Data in DB2 is accessed by using traditional relational SQL. REST Services are provided for data that needs to be accessed by the user interfaces and for certain operations such as alert management.

- The output, or the risk evidences from the use case implementations (trade, e-comm, and voice), are dropped into the Kafka messaging topics for the use case-specific Spark jobs. The Spark jobs perform the post processing after the evidences are received from the Streams jobs.

Deploy the IBM Surveillance Insight for Financial Services software

IBM Surveillance Insight for Financial Services is deployed on different node computers that host different parts of the solution. Some prerequisite components are required on each of the nodes.

The following diagram provides a high-level overview of the solution architecture.

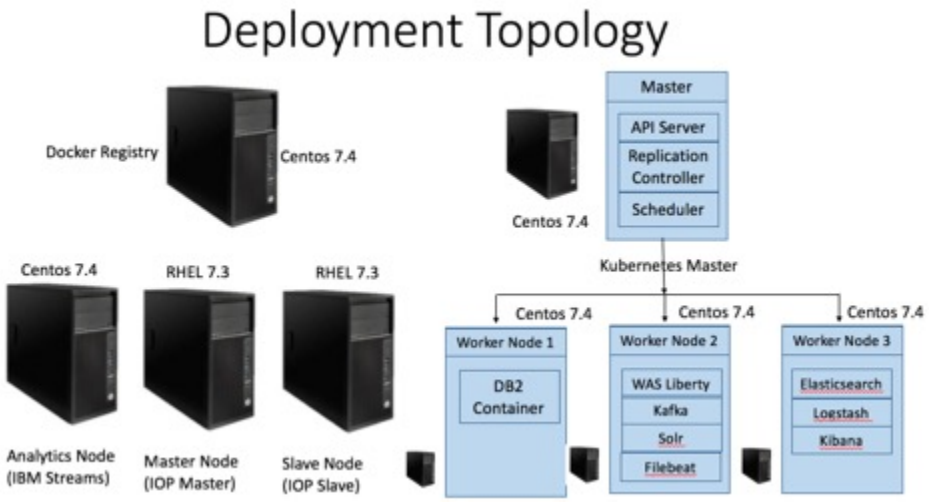


Figure 4: Deployment topology

There is a separate installer for each of the components that comprise IBM Surveillance Insight for Financial Services.

- IBM Surveillance Insight for Financial Services
- IBM Trade Surveillance Analytics
- IBM Electronic Communication Surveillance Analytics
- IBM Voice Surveillance Analytics
- IBM Complaints Analytics

The IBM Surveillance Insight for Financial Services base component also requires the following parts:

- IBM Surveillance Insight for Fin Serv DB2AWSE (1 of 8) 2.0.2 CentOS EN - CNPY7EN
- IBM Surveillance Insight for Fin Serv Liberty (2 of 8) 2.0.2 CentOS EN - CNPY8EN
- IBM Surveillance Insight for Fin Serv Kafka (3 of 8) 2.0.2 CentOS EN - CNPY9EN
- IBM Surveillance Insight for Fin Serv Solr (4 of 8) 2.0.2 CentOS EN - CNPZ0EN
- IBM Surveillance Insight for Fin Serv Kibana (5 of 8) 2.0.2 CentOS EN - CNPZ1EN
- IBM Surveillance Insight for Fin Serv Logstash (6 of 8) 2.0.2 CentOS EN - CNPZ2EN
- IBM Surveillance Insight for Fin Serv Elasticsearch (7 of 8) 2.0.2 CentOS EN - CNPZ3EN
- IBM Surveillance Insight for Fin Serv Filebeat (8 of 8) 2.0.2 CentOS EN - CNPZ4EN

Chapter 2. Supported operating systems and hardware requirements

Review the minimum hardware and operating system requirements before you install IBM Surveillance Insight for Financial Services.

For an up-to-date list of environments that are supported by IBM Surveillance Insight for Financial Services, see the IBM Software Product Compatibility Reports (www.ibm.com/support/docview.wss?uid=swg27047153).

The computer on which you run the solution installer and the computer on which you install IBM Surveillance Insight for Financial Services must be running 64-bit CentOS operating systems. The computers on which you install the IBM Open Platform components must be running 64-bit Red Hat Enterprise Linux Server Edition 7.3 operating systems.

Hardware requirements

The computer on which you install IBM Surveillance Insight for Financial Services must have the following hardware requirements:

Processors
2 sockets with 8 cores per socket

RAM
64 GB

Disk space
A flat partition with 500 GB of disk space

A `/var/lib/docker` directory with 100 GB of disk space that uses a B-tree file system (btrfs)

A `/docker-repo` directory with 100 GB of disk space that uses btrfs

Setting ulimit values

Before you install IBM Surveillance Insight for Financial Services, ensure that you have appropriate ulimit values. You set the ulimit values in two files: `90-nproc.conf` and `91-nproc.conf`.

Procedure

1. Log in to the computer as the root user or as a user with sudo permissions.
2. Go to the `/etc/security/limits.d` directory.
3. Open the `90-nproc.conf` file for editing. If the file does not exist, you must create it.
4. Add the following lines to the file:

```
*     soft  nproc  100000
root  soft  nproc  unlimited
```

5. Save and close the file.
6. Open the `91-nproc.conf` file for editing. If the file does not exist, you must create it.
7. Add the following lines to the file:

```
* - nofile 100000
```

8. Save and close the file.
9. Restart the computer for the changes to take effect.

Verifying host name configuration

Before installing IBM Surveillance Insight for Financial Services, you must verify the short host name, fully qualified host name, and domain name for your servers.

Perform these steps for each of your servers.

Procedure

1. Update the DNS settings or the `/etc/hosts` file on each server.

   Each server must have an entry in DNS or have an entry in the `/etc/hosts` file to allow for the resolution of both the short host name and long host name. To implement the name resolution by using the `/etc/hosts` file, edit the `/etc/hosts` file on the server to add the IP address of the server and the fully qualified host name. For example, add the following line to the file:

   `nnn.nnn.nnn.nnn sifsserver.example.com sifsserver`

   Where,

   - `nnn.nnn.nnn.nnn` is the IP address of the server
   - `sifsserver.example.com` is the fully qualified domain and host name for the server
   - `sifsserver` is the short name of the server

   Note: If you are using an `/etc/hosts` file for name resolution, ensure that `files` is listed first on the `hosts:` entry in the Name Service Switch configuration file (`/etc/nsswitch.conf` file).

   For example,

   ```
   # cat /etc/nsswitch.conf | grep "hosts"
   hosts: files dns myhostname
   ```

2. Log on to the server as `root` user and open a command prompt.
3. Verify and record the defined short host name for the server by typing the following command:

   `hostname -s`

   The command returns the defined short host name for the server, such as `sifsserver`.
4. Verify and record the fully qualified domain and host name for the server by typing the following command:

   `hostname -f`

   The command returns the fully qualified domain and host name for the server, such as `sifsserver.example.com`.
5. Verify and record the domain name for the server by typing the following command:

   `hostname -d`

   The command returns the domain name for the server, such as `example.com`.

Verifying server to server connectivity

Before you install IBM Surveillance Insight for Financial Services, you must verify that each server can ping the other associated servers.

The following procedure verifies the connectivity to each server from the Ambari server. You must perform these steps as the root user.

Procedure

1. Verify that the Ambari server can connect to the other servers by running the following commands on the Ambari server:

   ```
   ping hadoop-master_server.example.com
   ping hadoop-slave_server.example.com
   ```

2. Verify that the Kubernetes master server can connect to the other Kubernetes servers by running the following commands on the Kubernetes master server:

   ```
   ping kubernetes_nfs.example.com
   ping kubernetes_worker_1.example.com
   ping kubernetes_worker_2.example.com
   ```

3. Start SSH on each Hadoop server (hadoop-master, hadoop-slave), if it is not already started, by typing the following command on each server:

   ```
   systemctl start sshd
   ```

   If disabled, enable remote root login and password authentication by completing the following steps:

   a. Open the `/etc/ssh/sshd_config` file in a text editor.

   b. Find the lines that contain `PermitRootLogin` and `PasswordAuthentication` and ensure that both values are set to `yes`.

   c. Save the file.

   d. Run the following command to restart the SSH service:

      ```
      systemctl restart sshd
      ```

   Important: You must repeat this step on each of the Hadoop nodes: hadoop-master and hadoop-slave.

4. Verify that the `root` user can log in to each server via SSH from the Ambari server. While you are logged in to a Terminal session to the Surveillance Insight node server, verify that the `root` user can log in to the hadoop servers via SSH by typing the following commands:

   ```
   ssh root@hadoop-master_server.example.com
   ssh root@hadoop-slave_server.example.com
   ```

Chapter 3. Install prerequisite software

Some prerequisite software is required before you can install IBM Surveillance Insight for Financial Services.

Installing Python on the BigData master node

You must install Python on the BigData master node. You must also install the Python Package Index (PIP) and the pgmpy Bayesian network library.

Procedure

1. Download Python from https://www.python.org/downloads/release/python-352/.
2. Extract `Python-3.5.2.tar.xz`.

   For example,

   ```
   tar -xJf Python-3.5.2.tar.xz
   ```

3. Go to the extracted folder, and run the following commands:

   ```
   ./configure
   ```

   ```
   make
   ```

   ```
   make altinstall
   ```

   The `make install` command makes 3.5.2 the default Python. Because Spark has some dependencies on the 2.7.2 version, use he `make altinstall` command.
4. Verify the Python version:

   ```
   /usr/local/bin/python3.5 -V
   ```

   The result should display `Python 3.5.2`.
5. Use the following commands to install the Python modules:

   ```
   /usr/local/bin/pip3.5 install numpy
   /usr/local/bin/pip3.5 install pandas
   /usr/local/bin/pip3.5 install scipy
   /usr/local/bin/pip3.5 install pyparsing
   /usr/local/bin/pip3.5 install flask
   /usr/local/bin/pip3.5 install wrapt
   /usr/local/bin/pip3.5 install flask_cors
   /usr/local/bin/pip3.5 install keras
   /usr/local/bin/pip3.5 install sklearn
   /usr/local/bin/pip3.5 install pickle
   /usr/local/bin/pip3.5 install flask_restful
   /usr/local/bin/pip3.5 install ssl
   /usr/local/bin/pip3.5 install spacy
   /usr/local/bin/pip3.5 install configparser
   /usr/local/bin/pip3.5 install email_reply_parser
   /usr/local/bin/pip3.5 install textacy
   /usr/local/bin/pip3.5 install tensorflow
   /usr/local/bin/pip3.5 install flask_cors
   ```

6. Install the pgmpy Bayesian network library:

Note: Do not use PIP to install pgmpy.

a) Download the source (zip file) from https://github.com/pgmpy/pgmpy or clone the pgmpy repository. If you cloned the repository, use the following steps:

```
git clone https://github.com/pgmpy/pgmpy
git checkout dev
```

b) If you downloaded the source, decompress the file.

c) Run the `setup.py install` command.

For example,

```
/usr/local/bin/python3.5 setup.py install
```

7. Install spaCy.

a) As the root user, run the following command:

```
python -m spacy download en
```

You must install the following version: `en_core_web_sm-1.2.0/en_core_web_sm-1.2.0.tar.gz`

For more information about spaCy, see the models and language quck start (https://spacy.io/docs/usage/models).

Creating a group for IBM Streams on the Analytics node

You must create a group for Streams. For example, create a `streamsadmin` group. The group must exist before you install Apache Solr and IBM Streams.

Log in as the root user and use the following commands to add the `streamsadmin` user:

- Use the `groupadd` command to create the `streamsadmin` group. For example, in a terminal window, enter `groupadd streamsadmin`.

For more information about IBM Streams, see the product documentation (www.ibm.com/support/knowledgecenter/SSCRJU_4.2.0).

Installing IBM Streams on the Analytics node

You must install IBM Streams on the Analytics node computer. IBM Streams is a software platform that enables the development and execution of applications that process information in data streams. Incoming data is processed through IBM Streams and then output to the IBM Surveillance Insight for Financial Services data stores.

You must create a user for Streams. For example, create a `streamsadmin` user that belongs to the `streamsadmin` group. The user must exist before you can install the product.

Log in as the root user and use the following commands to add the `streamsadmin` user:

- Use the `useradd` command to create the `streamsadmin` user and include the `-g` option to add the user to the group. For example, in a terminal window, enter `useradd streamsadmin -g streamsadmin`.
- Use the `passwd` command to set the `streamsadmin` user's password. For example, enter `passwd streamsadmin`, and follow the prompts to set the password.

For more information about IBM Streams, see the product documentation (www.ibm.com/support/knowledgecenter/SSCRJU_4.2.0).

Procedure

1. Download IBM Streams 4.2 Fix Pack 2. For more information, see the IBM Streams Version 4.2 Fix Pack 2 page (http://www.ibm.com/support/docview.wss?uid=swg24043181). The Fix Pack contains a full version of the product.

   For example, if the following packages are missing, use the following commands to install them:

   ```
   sudo yum install gcc gcc-c++
   ```

   ```
   sudo yum install perl-XML-Simple
   ```

2. Go to the directory where you downloaded the installation files, and decompress `4.2.0.2-IM-Streams-el6-x86_64-fp0002.tar.gz`.
3. In the `4.2.0.2-IM-Streams-el6-x86_64-fp0002` directory, decompress `Streams-4.2.0.2-x86_64-el6.tar.gz`.
4. Go to the `StreamsInstallFiles` directory.
5. Copy `ibmjgssprovider.jar` packaged with the installer to `/opt/ibm/InfoSphere_Streams/4.2.0.2/java/jre/lib/ibmjgssprovider.jar`
6. In the `StreamsInstallFiles` directory, start the installer by using the following command: `./IBMStreamsSetup.bin`.
7. In the installer, on the **Select the edition to install** page, select **IBM Streams**, and click **Next**.
8. Ensure that you install all of the missing software packages that are identified on the **Software Dependencies** page.
9. Enter a Streams user and group. This user runs the Streams services. If the user does not exist, it is created by the installer.
   For example, enter `streamsadmin` as the **User** and `streamsadmin` as the **Group**.
10. To download the latest IBM Watson Speech to Text, contact Streams Support.
11. Copy the downloaded file to the `toolkit.speech2text` directory, and decompress the installation files for RHEL7 x86 64:

    ```
    tar xvf toolkit.speech2text-rapid3.1.0-v2.4.0-RHEL7.tar
    ```

    Important: You must decompress the file to the `toolkit.speech2text` directory.
12. In the `IBMWatson-speech2test` directory, review the `README.txt`.
13. Download `streamsx.inet` from Github (https://github.com/IBMStreams/streamsx.inet/releases/download/v2.7.4/streamsx.inet.toolkits-2.7.4-20160502-0727.tgz).
14. Ensure that the `streamsadmin` user has access to the decompressed files.

Securing communications for IBM Streams

You must import the public certificate that you created into the IBM Streams keystore.

Procedure

1. Enter the following command to import the public certificate:

   ```
   keytool -keystore /home/streamsadmin/security/SIDB2StreamsClient.jks -
   alias DB2Streams -import -file /home/db2inst1/SIDB2.arm
   ```

2. When prompted, enter the password that you used.

Installing Apache Kafka on the Services node

Apache Kafka is used with IBM Streams. You must download and install Apache Kafka on the services node.

For information about using Apache Kafka, see the Kafka documentation (https://kafka.apache.org/quickstart).

Procedure

1. Go to the Apache Kafka download page (https://www.apache.org/dyn/closer.cgi?path=/kafka/0.10.0.1/kafka_2.11-0.10.0.1.tgz).
2. Click one of the links to download the software.
3. In the download directory, decompress the file to the `/opt` directory.
   For example, enter `tar xvf kafka_2.11-0.10.0.1.tgz -C /opt`
4. Set the JAVA_HOME environment variable to point to OpenJDK Java™.

   ```
   export JAVA_HOME=/usr/jdk64/java-1.8.0-
   openjdk-1.8.0.77-0.b03.el7_2.x86_64/jre
   ```
5. Set the PATH environment variable to include the OpenJDK Java `bin` directory.

   ```
   export PATH=/usr/jdk64/java-1.8.0-openjdk-1.8.0.77-0.b03.el7_2.x86_64/jre/
   bin:$PATH
   ```

Securing data at rest for Apache Kafka

You must create a secure key and keystore, and configure IBM Streams and WebSphere® Application Server to be able to encrypt and decrypt messages with Apache Kafka.

Procedure

1. On the computer where Apache Kafka is installed, log on as the root user.
2. Enter the following command to create a secure key for decrypting data:

   ```
   keytool -genkey -alias SIKafkaSecurityKey -validity 365 -keyalg RSA -
   keysize 1024 -keystore SIKafkaDecrypt.jks -dname
   "CN=si.ibm.com,O=IBM,OU=IBMAnalytics,L=IN,ST=ON,C=CA" -keypass
   YourKeyPassword
   ```
3. When prompted, enter a password for the key.
4. Extract the certificate that you created to a public certificate file.

   ```
   keytool -export -alias SIKafkaSecurityKey -file SIKafka.arm -keystore
   SIKafkaDecrypt.jks
   ```
5. When prompted, enter the password that you used.

   Note: You must copy the extracted public certificate file to each computer that is running a component that encrypts messages to be sent to the Apache Kafka queue.
6. Create a keystore and import the public certificate file that you extracted in step 4.

   ```
   keytool -import -file SIKafka.arm -keystore SIKafkaEncrypt.jks -alias
   SIKafkaSecurityKey
   ```
7. When prompted, enter the password that you used.
8. Copy `SIKafkaDecrypt.jks` and `SIKafkaEncrypt.jks` files to the `/home/streamsadmin/security` directory.
9. Create a file that is named `encrypt.properties` in the `/home/streamsadmin/config/properties` directory.

10. Enter the following text into the `encrypt.properties` file.

```
algorithm=3DES
keylength=168
encryptionkeypath=/home/streamsadmin/security/SIKafkaEncrypt.jks
keystorepassword=YourPassword
aliasname=SIKafkaSecurityKey
```

11. Save and close the file.

    Note: Ensure that the streamsadmin user has access to this file.

12. Create a file that is named `decrypt.properties` in the `/home/streamsadmin/config/properties` directory.
13. Enter the following text into the `decrypt.properties` file.

```
encryptionkeypath=/home/streamsadmin/security/SIKafkaDecrypt.jks
keystorepassword=YourPassword
keypassword=YourKeyPassword
aliasname=SIKafkaSecurityKey
```

14. Save and close the file.

    Note: Ensure that the streamsadmin user has access to this file.

Securing data in motion for Apache Kafka

You must create a key and a certificate for the Apache Kafka broker.

Procedure

1. On the computer where Apache Kafka is installed, log on as the root user.
2. Create a key and a keystore for each Kafka broker.

```
keytool -genkey -alias SIKafkaServerSSL -validity 365 -keystore
SIKafkaServerSSLKeystore.jks -dname
"CN=si.ibm.com,O=IBM,OU=IBMAnalytics,L=IN,ST=ON,C=CA" -keypass
YourKeyPassword
```

3. When prompted, enter a password for the key.
4. Export the certificate from the keystore.

```
keytool -certreq -file SIKafkaCert -alias SIKafkaServerSSL -keystore
SIKafkaServerSSLKeystore.jks
```

5. When prompted, enter the password that you used.

   Note: The certificate must be signed by a certificate authority.

6. Generate the certificate authority key.

```
openssl req -new -x509 -keyout ca-key -out ca-cert -days 365
```

   Follow the prompts to generate the key.

7. Add the key to the server truststore.

```
keytool -import -file ca-cert -keystore SIKafkaServerSSLTruststore.jks -
alias CARoot
```

   The truststore is automatically created.

8. Add the key to the server keystore.

```
keytool -import -file ca-cert -keystore SIKafkaServerSSLKeystore.jks -
alias CARoot
```

9. Sign the certificate:

```
openssl x509 -req -CA ca-cert -CAkey ca-key -in SIKafkaCert -out
SIKafkaCertSigned -days 365 -CAcreateserial -passin pass:YourPassword
```

10. Import the signed certificate into the server keystore:

```
keytool -import -file SIKafkaCertSigned -keystore
SIKafkaServerSSLKeystore.jks -alias SIKafkaServerSSL
```

11. Update the *KafkaInstallLocation*/config/server.properties file to include the following text:

```
listeners=SSL://<IP>:<Port>
advertised.listeners=SSL://<IP>:<Port>
ssl.keystore.location=/home/SIUser/SIKafkaServerSSLKeystore.jks
ssl.keystore.password=YourPassword
ssl.key.password= YourKeyPassword
ssl.truststore.location=/home/SIUser/SIKafkaServerSSLTruststore.jks
ssl.truststore.password=YourPassword
ssl.client.auth=required
security.inter.broker.protocol=SSL
```

Where *<IP>* is the IP address where Kafka is running and *<Port>* can be any open port number, such as 2182.

12. Copy the SIKafkaServerSSLKeystore.jks and SIKafkaServerSSLTruststore.jks files to the /home/streamsadmin/security directory.

Note: Ensure that the streamsadmin user has access to this file.

Configuring SSL for Apache Kafka

Follow these steps to configure SSL for Apache Kafka. These steps must be performed on the computer where IBM Streams and WebSphere Application Server are installed.

Procedure

1. On the computer where Apache Kafka is installed, log on as the root user.
2. Add the signed certificate that you created in "Securing data in motion for Apache Kafka" on page 13 to the truststore.

```
keytool -import -file ca-cert -keystore SIKafkaClientSSLTruststore.jks -
alias CARoot
```

The truststore is automatically created.

3. When prompted, enter the password that you used.
4. Create a key and a keystore for each Kafka producer or consumer client.

```
keytool -genkey -alias SIKafkaClientSSL -validity 365 -keystore
SIKafkaClientSSLKeystore.jks -dname
"CN=si.ibm.com,O=IBM,OU=IBMAnalytics,L=IN,ST=ON,C=CA" -keypass
YourKeyPassword
```

5. When prompted, enter a password for the key.

6. Export the client certificate from the keystore. The certificate must be imported into the Apache Kafka server. This certificate can be self-signed.

```
keytool -export -file SIKafkaClientCert.arm -alias SIKafkaClientSSL -
keystore SIKafkaClientSSLKeystore.jks
```

7. When prompted, enter the password that you used.
8. Import the client certificate to the truststore for the Apache Kafka broker (server).

```
keytool -import -keystore SIKafkaServerSSLTruststore.jks -alias
SIKafkaClientCert1 -file SIKafkaClientCert.arm
```

9. Create a file that is named `producer.properties` in the `/home/streamsadmin/config/properties` directory.

   If the `config/properties` path does not exist in `/home/streamsadmin`, you must create it.

   The `producer.properties` file must contain the following information:

```
bootstrap.servers=<kafka_host_IP:port>
serializer.class=kafka.serializer.StringEncoder
request.required.acks=1
value.serializer=org.apache.kafka.common.serialization.ByteArraySerialize
r
key.serializer=org.apache.kafka.common.serialization.StringSerializer
security.protocol=SSL
ssl.protocol=TLSv1.1
ssl.enabled.protocols=TLSv1.1
ssl.truststore.type=JKS
ssl.truststore.location=/home/streamsadmin/security/
SIKafkaServerSSLTruststore.jks
ssl.truststore.password=YourTruststorePassword
ssl.keystore.location=/home/streamsadmin/security/
SIKafkaServerSSLKeystore.jks
ssl.keystore.password=YourKeystorePassword
ssl.key.password=YourKeyPassword
```

10. Create a file that is named `consumer.properties` in the `/home/streamsadmin/config/properties` directory.

    The `consumer.properties` file must contain the following information:

```
bootstrap.servers=<kafka_host_IP:port>
serializer.class=kafka.serializer.StringEncoder
request.required.acks=1
value.serializer=org.apache.kafka.common.serialization.ByteArraySerialize
r
key.serializer=org.apache.kafka.common.serialization.StringSerializer
security.protocol=SSL
ssl.protocol=TLSv1.1
ssl.enabled.protocols=TLSv1.1
ssl.truststore.type=JKS
ssl.truststore.location=/home/streamsadmin/security/
SIKafkaServerSSLTruststore.jks
ssl.truststore.password=YourTruststorePassword
ssl.keystore.location=/home/streamsadmin/security/
SIKafkaServerSSLKeystore.jks
ssl.keystore.password=YourKeystorePassword
ssl.key.password=YourKeyPassword
```

Installing IBM BigInsights on the IOP master node

You must install IBM Open Platform before you can install IBM BigInsights®.

For more information about IBM BigInsights, see the product documentation (https://www.ibm.com/support/knowledgecenter/SSPT3X_4.2.0).

Procedure

1. Click the following link to download the IBM repository for IBM Open Platform: http://ibm.biz/downloadi-file-el7-x86-64-native.

   For more information, see Downloading the IBM repository definition for the IBM Open Platform with Apache Spark and Apache Hadoop (https://www.ibm.com/support/knowledgecenter/SSPT3X_4.2.0/com.ibm.swg.im.infosphere.biginsights.install.doc/doc/bi_install_download_software.html) in the IBM BigInsights documentation.
2. Follow the steps in Running the installation package (https://www.ibm.com/support/knowledgecenter/SSPT3X_4.2.0/com.ibm.swg.im.infosphere.biginsights.install.doc/doc/bi_install_iop_biginsights.html#bi_install_IOP_BigInsights) to complete the installation.
   a) In step 17 of Running the installation package, select HDFS, YARN, and Ambari-Metrics.

      MapReduce2 and ZooKeeper should be automatically selected.
   b) When you are enabling the YARN service, select at least 2 NodeManager nodes.
   c) After the installation is complete, you can log in from the Ambari console to verify that all of the services are running.
3. Create a KDC instance, as described in Setting up a KDC manually (https://www.ibm.com/support/knowledgecenter/SSPT3X_4.2.0/com.ibm.swg.im.infosphere.biginsights.admin.doc/doc/admin_kerb_mankdc2.html).

   Note: You must disable 256-bit encryption. To do this, remove aes256-cts:normal from the supported_enctypes field of the `/var/kerberos/krb5kdc/kdc.conf` file.
4. Enable Kerberos in Ambari as described in Setting up Kerberos for IBM Open Platform with Apache Spark and Apache Hadoopclusters (https://www.ibm.com/support/knowledgecenter/SSPT3X_4.2.0/com.ibm.swg.im.infosphere.biginsights.admin.doc/doc/admin_iop_kerberos.html).
   a) You must create a separate non-root user on all YARN Node Managers that will be used to run Spark. Name the user `sifsuser`.

      For example, use the following commands to create the user:

      ```
      sudo groupadd sifsuser
      ```

      ```
      sudo useradd -g streamsadmin sifsuser
      ```
   b) As the Hadoop Distributed File System (hdfs) user, create a home directory on hdfs for `sifsuser`.

      For example, `hdfs dfs -mkdir /user/sifsuser`
   c) Add a principal by running `kadmin.local` as the root user, and entering the following in the prompt:

      ```
      addprinc -randkey sifsuser@IBM.COM
      ktadd -norandkey -k /etc/security/keytabs/sifsuser.keytab
      sifsuser@IBM.COM
      ```

      Then set the ownership on the new keytab file by entering the following in the prompt:

      ```
      chown sifsuser:hadoop /etc/security/keytabs/sifsuser.keytab
      chmod a+r /etc/security/keytabs/sifsuser.keytab
      ```

d) Copy `sifsuser.keytab` to all of the YARN Node Manager nodes.

e) Log in as the sifsuser and run the following command to initialize the Kerberos ticket:

```
kinit -kt /etc/security/keytabs/sifsuser.keytab sifsuser@IBM.COM
```

Enabling Hadoop encryption

You must enable encryption in Hadoop.

Procedure

1. Create a service user. For example, `useradd kms`.
2. Copy the Hadoop-KMS package to the home directory.
   For example, enter the following command:

   ```
   cp /usr/iop/current/hadoop-client/mapreduce.tar.gz /home/kms/
   mapreduce.tar.gz
   ```

3. Extract the archive.
   For example, enter the following command:

   ```
   export KMS_ROOT=/home/kms/
   cd $KMS_ROOT
   tar -xvf mapreduce.tar.gz
   ```

4. Start the KMS server.
   a) If you do not have the JAVA_HOME variable set, run the following command:

   ```
   export JAVA_HOME=/usr/jdk64/java-1.8.0-
   openjdk-1.8.0.77-0.b03.el7_2.x86_64/jre
   ```

   Ensure that you use the appropriate path for your environment.

   b) Go to the `$KMS_ROOT/hadoop/sbin/` directory.

   c) Enter the following command: `./kms.sh run`

   Wait until you see that the server started.
5. From the Ambari console, update the KMS server.
   a) In the Ambari console, click the HDFS service.

   b) Click **Configs** > **Advanced**.

   c) Add the following values:

Configuration section	Key	Value1
Advanced core-site	hadoop.security.key.provider.path	kms://http@<KMS Server IP>:16000/kms
Advanced hdfs-site	dfs.encryption.key.provider.uri	kms://http@<KMS Server IP>:16000/kms

6. Generate a key as a regular user.
   a) Log on as a regular user, such as ambari-qa.

   b) Create the key by entering the following command: `hadoop key create ambariqa-key`
7. Create an encryption zone for the `/user/sifsuser` directory.
   a) Log in as the hdfs user.

b) Run the following commands:

```
hdfs crypto -createZone -keyName ambariqa-key -path /user/sifsuser
hdfs dfs -chown sifsuser:hadoop /user/sifsuser
```

Tip: If you encounter any errors, you can check the following log directories:

- `/var/log/hadoop/hdfs`
- `/var/log/ambari-server`
- `/var/log/ambari-agent`
- `/var/lib/ambari-agent/data`

8. Verify that the contents are encrypted.
   a) Log in as the sifsuser.
   b) Copy a test data file to the `/user/sifsuser` directory.
   c) Run the following commands:

```
hdfs dfs -put testdata.txt /user/sifsuser/
hdfs dfs -cat /user/sifsuser/testdata.txt
hdfs dfs -cat /.reserved/raw/user/sifsuser/testdata.txt
```

This should show decrypted, clear text data.

Run the following command:

```
hdfs dfs -cat /.reserved/raw/user/sifsuser/testdata.txt
```

This should show encrypted data.

Note: If the Kerberos session has expired, you can run the `kinit` command.

Installing Apache Ant libraries on all nodes

You must install Apache Ant and Ant Contrib on all of the computers on which you will install a IBM Surveillance Insight for Financial Services component.

Procedure

1. Download Apache Ant from the Apache Ant website (ant.apache.org/srcdownload.cgi).
2. Decompress the downloaded file to any location.
3. Edit the `$HOME/.bash_profile` file to include the following:

   ```
   $ANT_HOME=/path_to_ant
   ```

4. Download the `ant-contrib-1.0b3.jar` file.
   For example, go to https://sourceforge.net/projects/ant-contrib/files/ant-contrib/1.0b3/, and download `ant-contrib-1.0b3-bin.zip`.
5. Copy `ant-contrib-1.0b3.jar` to the *ANT_HOME*`/lib` directory.

Chapter 4. Install the Surveillance Insights artifacts on the Analytics and IOP nodes

There is a separate installer for each of the components that comprise IBM Surveillance Insight for Financial Services.

There are separate installers for the following components:

- IBM Surveillance Insight for Financial Services
- IBM Trade Surveillance Analytics
- IBM Electronic Communication Surveillance Analytics
- IBM Voice Surveillance Analytics
- IBM Complaints Surveillance Analytics

Creating directories for the solution installer

You must create a directory on each computer on which you deploy an IBM Surveillance Insight for Financial Services component.

The solution installer uses the `/opt/IBM` directory to copy license files and other files. This directory must exist on each node computer and on the computer on which you run the solution installer before you run the installation.

Procedure

1. Create an `/opt/IBM` directory on each computer on which you are going to deploy an IBM Surveillance Insight for Financial Services component.
2. Create an `/opt/IBM` directory on the computer on which you are going to run the solution installer.

Adding each node computer to the hosts file on all computers

You must add all computers on which you deploy an IBM Surveillance Insight for Financial Services component to the `hosts` file on each other computer.

Procedure

1. On each computer, open the `/etc/hosts` file.
2. Ensure that each node computer is listed in the file.
   For example, ensure that your `hosts` files contain entries in the following pattern:

```
127.0.0.1 localhost.localdomain localhost
IP_Address computer1.domain.com computer1
IP_Address computer2.domain.com computer2
```

3. Save and close the file.

Modifying the sudoers file for the user who runs the installation

To deploy IBM Surveillance Insight for Financial Services components as a non-root user, you must add that user to the `sudoers` file on each computer.

Procedure

1. Log in as root user.
2. Go to the `etc` directory, and open the `sudoers` file in a text editor.
3. Add the following line for your user:

```
username ALL=(ALL) ALL
```

4. Save and close the file.
5. Repeat these steps on each computer on which you deploy an IBM Surveillance Insight for Financial Services component.

Downloading and decompressing the installation files

You download the IBM Surveillance Insight for Financial Services solution from IBM Passport Advantage®, and then decompress the files to run the solution installer.

For more information about the files that you must download, see Downloading IBM Surveillance Insights for Financial Services (www.ibm.com/support/docview.wss?uid=swg24042930).

Procedure

1. Access the IBM Passport Advantage web site.

   Tip: If you receive an error, use a different web browser to access Passport Advantage.
2. Sign in and navigate to the software downloads page.
3. Find the eImages for IBM Surveillance Insight for Financial Services.
4. Download an eImage by selecting the check box beside the name.

   After the download is complete, a **Download Complete** message is displayed. The location of the downloaded files is displayed in the message window.
5. Decompress the installation files.

Preparing the downloaded files

You must copy some of the downloaded files to a common directory. You must also modify one of the compressed files.

For more information about the files, see "Deploy the IBM Surveillance Insight for Financial Services software" on page 3.

Procedure

1. Create a temporary directory for the installation files.
   For example, `mkdir sifsmedia`

   The directory can be any directory on the file system, but it should have at least 10 GB of free disk space.
2. Copy the downloaded files to the `/sifsmedia` directory.

   Copy the following downloaded files:

   - IBM Surveillance Insight for Fin Serv DB2AWSE (1 of 8) 2.0.2 CentOS EN - CNPY7EN
   - IBM Surveillance Insight for Fin Serv Liberty (2 of 8) 2.0.2 CentOS EN - CNPY8EN
   - IBM Surveillance Insight for Fin Serv Kafka (3 of 8) 2.0.2 CentOS EN - CNPY9EN
   - IBM Surveillance Insight for Fin Serv Solr (4 of 8) 2.0.2 CentOS EN - CNPZ0EN
   - IBM Surveillance Insight for Fin Serv Kibana (5 of 8) 2.0.2 CentOS EN - CNPZ1EN
   - IBM Surveillance Insight for Fin Serv Logstash (6 of 8) 2.0.2 CentOS EN - CNPZ2EN

- IBM Surveillance Insight for Fin Serv Elasticsearch (7 of 8) 2.0.2 CentOS EN - CNPZ3EN
- IBM Surveillance Insight for Fin Serv Filebeat (8 of 8) 2.0.2 CentOS EN - CNPZ4EN

3. Decompress the IBM Surveillance Insight for Fin Serv DB2AWSE (1 of 8) 2.0.2 CentOS EN downloaded file (`CNPY7EN.tar`).
4. Copy `sifs-2.0.2.tar` and `Base` to another location, and remove them from the `CNPY7EN` directory. For example, copy them to a `/sifs` directory.
5. Recompress `CNPY7EN`.
   For example, `tar -cvf CNPY7EN`.

Opening firewall ports for the solution installer

You can run the `firewall.sh` script to open the ports that are required on the computer on which you are running the IBM Surveillance Insight for Financial Services solution installer.

You must also open ports on the target, or client, computer on which you are installing the IBM Surveillance Insight for Financial Services components. You can use the `client_firewall.sh` script to open the required ports.

The `firewall.sh` script opens the following ports on the solution installer computer:

- 8080 incoming
- 445 incoming
- 9683 incoming
- 22 outgoing

On the target, or client computers, the `client_firewall.sh` script opens the following ports:

- 8080 outgoing
- 445 incoming
- 9683 outgoing
- 22 incoming

Procedure

1. Log on to the computer that contains the solution installer node as the root user or as a user with sudo permissions.
2. Back up your existing firewall settings by typing the following command: `/etc/init.d/iptables save`.
3. Go to the `SolutionInstaller` directory where you decompressed the solution installer files, and run the firewall script by typing the following command: `sh firewall.sh`.
4. Copy the `client_firewall.sh` file onto the computer on which you are going to install IBM Surveillance Insight for Financial Services.
5. On the client computer, back up your existing firewall settings by typing the following command: `/etc/init.d/iptables save`.
6. Go to the directory where you copied the `client_firewall.sh` file, and run the script by typing the following command: `sh client_firewall.sh`.

Starting the solution installer

You use the solution installer to deploy the components. After the solution installer is running, you can access the installer interface from a web browser.

Note: Ensure that you copy the solution installer files to a directory in which you have permissions to execute files.

Procedure

1. Log on to the computer where you decompressed the installation files as the root user or as a user with sudo permissions.
2. Go to the `CNPY6EN/SolutionInstaller` directory where you decompressed the solution installer files.
3. Enter the following command: `sh setup.sh` *`username first_name last_name email password`*.

   You must enter each of the values after `setup.sh`. If you do not enter a password, you will be prompted to enter one. The password must have at least 6 characters.

   After the solution installer starts, open a web browser, and go to the solution installer URL: `https://`*`servername`*`:8080/UI/index.html`.

   The computer on which you are using the browser to access the solution installer must have a screen resolution that is greater than 1024 by 760.

   The solution installer interface can be accessed from a Google Chrome 44, or later, or Mozilla Firefox 38 or later, web browser. It does not run in an Internet Explorer web browser.

Using the solution installer to deploy the base component artifacts

Use the solution installer to copy the IBM Surveillance Insight for Financial Services base component files to the computer or computers on which you want to install the solution components.

Procedure

1. Open the solution installer in a web browser.

   After the solution installer is running, you can access the URL from any computer from a Firefox or Chrome web browser.

   The URL is `https://`*`servername`*`:8080/UI/index.html`, where *`servername`* is the name of the computer where you ran the solution installer.
2. Click **New Configuration**

   If you have a configuration that was previously saved, you can start from that saved configuration.
3. From the **Mandatory Content List** pane, select **Node** and drag it to the **Configuration Editor** pane.

   The **Node** represents the computer where the IBM Surveillance Insight for Financial Services files are to be placed.

   For example, you define a node for the computer where IBM Streams is installed. The **Streams Content** component must be deployed to that computer.

   You can deploy the content on the same computer or on different computers.
4. Select a node object, and enter the following information in the **Property Editor** pane:
   a) Enter a name for the node in **Name**, and press Enter.
   b) Enter the server name in **Host Name**, and press Enter.
   c) Enter the user who has access to install the components in **User Name**, and press Enter.
   For example, enter `root` or a user with sudo permissions.
   d) Enter the user's password in **User Password**, and press Enter.
5. Repeat steps 3 - 4 for each node that you want to install content on.
6. From the **Mandatory Content List** pane, drag the components to the appropriate node that you defined.

   Analytics Content
   Contains the Streams Jobs and Configuration files.

   You must deploy this component to a computer where you installed the IBM Streams.

BigData Master Content
Contains the deployment files for the Surveillance BigData component.

You must deploy this component to a computer where you installed the IBM Open Hadoop Master.

BigData Slave Content
Contains the deployment files for the Surveillance BigData component.

You must deploy this component to a computer where you installed the IBM Open Hadoop Slave.

The files content files are copied to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2` directory. You can change that value in the **Property Editor**.

Tip: If you do not want the components to be decompressed automatically, select each of the content components, and in the **Property Editor**, clear **Uncompress File**.

7. Click **Validate** to ensure that the configuration is complete.

   Any errors or missing information is displayed. You must correct the error or provide the information before you can run the deployment.
8. Click **Run** to start the deployment.
9. When the deployment is complete, click **Close**, and then exit the solution installer.

Using the solution installer to deploy the remaining solution components

There are five components that you must install for IBM Surveillance Insight for Financial Services.

After you install a component, you must uninstall the solution installer and then re-install the solution installer for the next component. You must repeat this task for each component that you install.

Important: You must run a script on each node computer where you installed an IBM Surveillance Insight for Financial Services component to remove the solution installer processes on the node computer. Ensure that you copy the `cleanupClient.sh` file from the `SolutionInstaller` directory before you uninstall the server solution installer.

Procedure

1. To remove the solution installer processes on the client node computers, do the following steps on each client node computer:
   a) From the installation computer, copy the `SolutionInstaller/cleanupClient.sh` file onto each computer on which you installed an IBM Surveillance Insight for Financial Services component.
   b) On the client node computer, go to the directory where you copied the `cleanupClient.sh` file.
   c) Enter the following command: `sh cleanupClient.sh`.
2. To remove the solution installer from the installation computer, do the following steps:
   a) Go to the `SolutionInstaller` directory.
   b) Enter the following command: `sh cleanup.sh`.
   c) Restart the installation computer.

Extracting the Kubernetes artifacts and Docker images

Some IBM Surveillance Insight for Financial Services components are provided as Kubernetes artifacts and Docker images.

Procedure

1. Log in to the Kubernetes master node computer as the root user.

2. Create a temporary directory for the installation files.
   For example, `mkdir sifs`

   The directory can be any directory on the file system, but it should have at least 10 GB of free disk space.
3. Copy `sifs-2.0.2.tar` to the temporary directory.
4. Extract the `sifs-2.0.2.tar` file.

```
cd /sifs && tar -zxvf sifs-2.0.2.tar.gz
```

Preparing the installation media

After you have extracted the Kubernetes artifacts and Docker images, you must prepare the files.

Before you begin

Ensure that you:

- Copy all of the IBM Surveillance Insight for Financial Services installation media files to the Kubernetes master node.
- Run the install as the root user on the Kubernetes master node.

Procedure

1. Download and copy the installation TAR files to the installation media directory.

   The installation media directory value is set by the media.dir property in the `install.properties` file. The default directory is `/sifsmedia`.
2. Create a SHA1 checksum file that is named `SIFS.sha1` in the installation media directory.

   For example, use the following command to create the checksum file:

```
cat > /sifsmedia/SIFS.sha1 << EOF
22f415ae2a2a279b456f273d8e22eab0a6f32060  CNPY7EN.tar
1eae1d481f9dd1ddd4e172f9f4efcae3c571cd00  CNPY8EN.tar
f2ce2e576b5b64e81aa46857dec78a741952cec4  CNPY9EN.tar
f0a0ac811779578995bd1094228bedc75de2846c  CNPZ0EN.tar
ac495aa105d06c4fd7f5b0907d84f07dc5552648  CNPZ1EN.tar
5c2b46841390758677a3d0217d67ebd2ec0c405f  CNPZ2EN.tar
af2c8228b1a523de291de5776c18ee02c564ca05  CNPZ3EN.tar
334fb17a3d09b57c2280d3bf8df7b4b83e5ec0f2  CNPZ4EN.tar
EOF
```

3. Change the SHA1 checksum file permissions and run the `dos2unix` command on the checksum file:

```
chmod 755 SIFS.sha1
```

```
dos2unix SIFS.sha1
```

   If you do not have dos2unix installed, enter the following command to install it:

```
yum install -y dos2unix
```

4. Extract the installation scripts to the root user's home directory:

```
mkdir ${HOME}/sifs-2.0.2
```

```
tar xOf /sifsmedia/sifs-2.0.2.tar sifs-2.0.2.tar | tar xzf - -C ${HOME}/
sifs-2.0.2
```

```
cd ${HOME}/sifs-2.0.2/kubernetes/sifs-install
```

5. Open the `${HOME}/sifs-2.0.2/kubernetes/sifs-install/prepare/install_kubernetes.txt` file in a text editor.
6. Add `kubernetes-cni-0.5.1` to the end of the `yum install` and `yum versionlock` lines.

   For example:

```
yum install -y kubelet-1.7.5 kubeadm-1.7.5 kubectl-1.7.5 bridge-utils nfs-
utils kubernetes-cni-0.5.1
yum versionlock kubelet-1.7.5 kubeadm-1.7.5 kubectl-1.7.5 kubernetes-
cni-0.5.1
```

   You can also use sed commands to modify the file. For example:

```
sed -i '/yum install/c\yum install -y kubelet-1.7.5 kubeadm-1.7.5
kubectl-1.7.5 bridge-utils nfs-utils kubernetes-cni-0.5.1' ${HOME}/
sifs-2.0.2/kubernetes/sifs-install/prepare/install_kubernetes.txt
```

```
sed -i '/yum versionlock/c\yum versionlock kubelet-1.7.5 kubeadm-1.7.5
kubectl-1.7.5 kubernetes-cni-0.5.1' ${HOME}/ sifs-2.0.2/kubernetes/sifs-
install/prepare/install_kubernetes.txt
```

7. Save and close the file.

Customizing the `install.hosts.properties` file for the deployment

You must modify the `install.hosts.properties` file before you can run the IBM Surveillance Insight for Financial Services installation.

The `install.hosts.properties` file lists the servers on which you will install IBM Surveillance Insight for Financial Services components. The file contains five fields separated by tabs. For example,

`<server_purpose> <ip_address> <fqdn> <short_name> <root_password>`

Where:

- `<server_purpose>` is the type of node. The values can be:
  - `manager` for the Kubernetes cluster manager
  - `docker` for the Docker registry node
  - `nfs` for the NFS server
  - `worker.##` for the worker node in the Kubernetes cluster
- `<ip_address>` is the IP address for the server
- `<fqdn>` is the fully qualified domain name of the server. All machines in the configuration must be able to reach the other machines in the configuration by using this name.
- `<short_name>` is the short name for the server
- `<root_password>` is the root password for the server

For example:

```
manager    192.168.244.167    kube-mgr.yourdomain.local    kube-mgr
root_password
docker    192.168.244.168    kube-docker.yourdomain.local    kube-docker
root_password
worker.1    192.168.244.169    kube-worker1.yourdomain.local    kube-
worker1    root_password
worker.2    192.168.244.170    kube-worker2.yourdomain.local    kube-
worker2    root_password
worker.3    192.168.244.171    kube-worker3.yourdomain.local    kube-
worker3    root_password
```

Procedure

1. Log in to the Kubernetes master server computer as the root user.
2. Go to the `/sifs` directory, or the directory where you extracted `sifs-2.0.2.tar`.
3. Open the `install.hosts.properties` file in a text editor.
4. Update the properties to match your environment.
5. Save and close the file.

Customizing the `install.properties` file for the deployment

You must modify the `install.properties` file before you can run the IBM Surveillance Insight for Financial Services installation.

Procedure

1. Log in to the Kubernetes master server computer as the root user.
2. Go to the `/sifs` directory, or the directory where you extracted `sifs-2.0.2.tar`.
3. Open the `install.properties` file in a text editor.
4. Edit the properties for your environment:
   - `docker.type` is the type of Docker installation
     - Enter `ce` to use Docker Community Edition
     - Enter `ee` to use Docker Enterprise Edition
   - `docker.ee.url` is the URL for the yum repository for Docker Enterprise Edition. This value is used only if you enter `ee` for the `docker.type` value. To obtain the `docker.ee.url` value:
     a. Go to https://store.docker.com/my-content.
     b. Click **Setup** for Docker Enterprise Edition for CentOS.
     c. Copy the URL to the `install.properties` file.
   - `external.docker.registry.url` to use an external Docker registry. The format of this value is *<docker_registry_hostname>*:*<registry_portnumber>*/*<registry_prefix_if_required>*. For example, `external-registry.yourcompany.com:5000/`

     If you do not enter a value, the Docker registry is installed on the `docker` value that you specified in the `install.hosts.properties` file.
   - `external.docker.registry.isSecure` if the external Docker registry is secure.
     - Enter `true` if the Docker registry is secure.
     - Enter `false` if the registry is not secured. If you enter `false`, you must also update the `/etc/docker/daemon.json` to include the registry in the insecure-registries value for all nodes in the Kubernetes cluster.

- `external.nfsserver` if you are using an external NFS server. The NFS server must be NFS4 or later and you must be able to mount to the / directory in read/write mode to create mount points.
- `media.dir` is the location where the installation media files reside.
- `offline.install` for whether the servers do not have internet access during the installation. Enter `true` if the installation is running on servers that do not have internet access. Enter `false` if the installation is running on servers that have internet access.
- `offline.rpm.dir` for offline installations for the location of the required RPMs.
- `offline.image.dir` for offline installations for the location of the Docker image archives.
- `installation.properties` is the detailed installation configuration file for the type of installation you are doing. For example, `sifs-core-`*`[version]`*`.properties` is the base version.

At a minimum, the `install.properties` file should contain:

```
docker.type = ce
media.dir = /sifsmedia
installation.properties = sifs-core-2.0.2.properties
```

Running the IBM Surveillance Insight for Financial Services installation

After you configure the `install.hosts.properties` and `install.properties` files, you can run the IBM Surveillance Insight for Financial Services installation.

Procedure

1. Go to the `cd ${HOME}/sifs-2.0.1/sifs-install1/fci-install` directory.
2. Run the following command: `./sifsinstall.sh`

   The installation takes up to 2.5 hours to complete.

Kubernetes and Docker commands

The following are helpful commands to use with Kubernetes and Docker for IBM Surveillance Insight for Financial Services.

List the services

To list all of the services in the namespace:

```
kubectl get services
```

List the pods

To list all of the pods in the namespace:

```
kubectl get pods
```

To list the pods with more details:

```
kubectl get pods -o wide
```

Pod logs

To dump the pod log files in the namespace:

```
kubectl logs my-pod
```

To dump the pod log files in the namespace if you have multiple containers:

```
kubectl logs my-pod -c my-container
```

Run a command in a pod

To run a command in a single container pod:

```
kubectl exec my-pod -- ls /
```

To run a command in a multi-container pod:

```
kubectl exec my-pod -c my-container -- ls /
```

IBM Surveillance Insight for Financial Services uses the following containers:

- `sifs-db2instance`
- `sifs-liberty-instance`
- `sifs-solr-instance`
- `sifs-elasticsearch-instance`
- `sifs-logstash-instance`
- `sifs-kibanna-instance`
- `sifs-filebeat-instance`

Describe commands

```
kubectl describe nodes my-node
kubectl describe pods my-pod
```

List Docker images

```
docker images
```

List Docker containers

```
docker ps -a
```

Copying files from and to Docker

```
docker cp CONTAINER:SRC_PATH DEST_PATH|-
docker cp SRC_PATH|- CONTAINER:DEST_PATH
```

Manual steps

Solr Docker

1. Log in to the Solr Docker container through the services pod, and run the following commands:

```
cd /home/solr-anchor/etc
keytool -genkeypair -alias solr-ssl -keyalg RSA -keysize 2048 -keypass
```

```
$password -storepass $password -validity 365 -keystore ${dockerDir}/etc/
solrssl.keystore.jks -dname "CN=<Kube Master IP>, OU=IBM, O=IBM, C=IN"
```

```
keytool -export -keystore solr-ssl.keystore.jks -alias solr-ssl -file
solr.cer
```

2. Enter the password to be set for the keystore.

Liberty Docker

Log in to the Liberty Docker container through the services pod, and run the following steps:

1. Go to the `/opt/ibm/wlp/usr/servers/SIFSServer` directory.
2. Open `server.env` in a text editor.
3. Update the DB2, NLC, NLU, and keystore-related entries.
4. Save and close the file.
5. Go to the `/opt/ibm/wlp` directory.
6. Run the following commands:

```
./server stop SIFSServer
./server start SIFSServer
```

7. Copy all of the generated keys for Kafka from the Kafka VM (`/home/sifsuser/security`) to any location in the Liberty container.

Configuration between IOP and the Liberty Docker container

1. Copy the following files from the IOP master node to the NFS mount point of the Liberty persistent volume:
   - `/etc/security/keytabs/sifsuser.keytab`
   - `/usr/iop/4.2.0.0/hadoop/conf` —You must copy the entire directory.
2. Update the location in the `server.env` file for the Liberty Docker container.
3. Restart the SIFSServer service.

Install YASM, NASM, and ffmpeg on the Liberty Docker container

For the voice component, you must install YASM, NASM, and ffmpeg on the Liberty Docker container.

Log into the Liberty Docker container, and do the following:

1. Download ffmpeg from http://ffmpeg.org/releases/ffmpeg-3.3.2.tar.bz2.
2. Extract `ffmpeg-3.3.1.tar.bz2` into a directory.
3. Follow the instructions in `INSTALL.md` to install the utility.
4. Install YASM by following the instruction at http://www.linuxfromscratch.org/blfs/view/cvs/general/yasm.html.
5. Install NASM by following the instructions at http://www.linuxfromscratch.org/blfs/view/svn/general/nasm.html.

Configure a case manager for IBM Surveillance Insight for Financial Services

You can configure IBM Surveillance Insight for Financial Services to work with IBM Financial Crimes Insight Case Manager or Actiance Case Manager.

Integrating with IBM FCI Case Manager

You can configure IBM Surveillance Insight for Financial Services to use IBM Financial Crimes Insight Case Manager.

Procedure

1. On a computer from where the IBM FCI Case Manager is accessible, create a directory. For example, create a `/home/sifsuser/casemanager` directory.
2. Copy the following files to the new directory:
   - `commons-codec-1.9.jar`
   - `commons-logging-1.2.jar`
   - `db2jcc4.jar`
   - `httpclient-4.5.2.jar`
   - `httpcore-4.4.4.jar`
   - `json-20160212.jar`
   - `services.sifscmregistration-2.0.2-SNAPSHOT.jar`
   - `CF_Individual.json`
   - `fci_sifs.properties`

   The files are available in the Liberty Docker container, in the `/opt/ibm/sifs/liberty/casemanager/` directory.

   For example, run the following command:

   ```
   java -cp "/home/sifsuser/casemanager/*"
   com.ibm.sifs.services.RegisterSIFS /home/sifsuser/casemanager/
   fci_sifs.properties
   ```
3. Update the JNDI variable to point to the IBM FCI Case Manager.

   ```
   REGISTER_CASE=true
   SIFS_CASE_URL=https://<kube master IP>:9443/casemanagerservices/case
   CASEMANAGER_PROP_LOC=/home/sifsuser/casemanager/fci_sifs.properties
   CASEMANAGER_FULL_CLASSNAME=com.ibm.sifs.cm.fci.service.FCICaseManagerImpl
   ```

   Ensure that you change the SIFS_CASE_URL and the CASEMANAGER_PROP_LOC for your environment.
4. Modify the `fci_sifs.properties` file to match you IBM FCI Case Manager properties.
5. Restart the Liberty server.

Integrating with Actiance Case Manager

You can configure IBM Surveillance Insight for Financial Services to use Actiance Case Manager.

Procedure

1. Update the JNDI variable to point to Actiance Case Manager.

   ```
   REGISTER_CASE=true
   SIFS_CASE_URL=https://<kube master IP>:9443/casemanagerservices/case
   CASEMANAGER_PROP_LOC=/home/sifsuser/casemanager/actiance_sifs.properties
   CASEMANAGER_FULL_CLASSNAME=com.ibm.sifs.cm.fci.service.ActianceCaseManager
   Impl
   ```

   Ensure that you change the SIFS_CASE_URL and the CASEMANAGER_PROP_LOC for your environment.
2. Modify the `actiance_sifs.properties` file to match you ActianceCase Manager properties.

3. Restart the Liberty server.

Installing the base component artifacts

To install the IBM Surveillance Insight for Financial Services base component artifacts, you must run scripts on each node computer.

Procedure

1. Create the IBM Streams domain and instance, copy jar files, and copy Streams projects to the home directory on the analytics node computer.
   a) Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/Analytics/FinancialMkts_SurveillanceInsight_AnalyticsContent/bin` directory.
   b) Open the `build.properties` file and update the file with the appropriate values for your environment.
   c) Run the following command: `sh Install_Analytics.sh`
2. Copy jar files and properties files on the BigData node computer.
   a) Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/BigData/FinancialMkts_SurveillanceInsight_BigDataContent/bin` directory.
   b) Open the `build.properties` file and update the file with the appropriate values for your environment.
   c) Run the following command: `sh Install_BigData.sh`

Replacing the IBM Streams Java file

After you install IBM Streams, you must replace a JAR file with on that is provided by the IBM Surveillance Insight for Financial Services installer.

Procedure

1. On the computer where you installed the IBM Surveillance Insight for Financial Services base components, go to the `/opt/IBM/ IS_FinancialMkts_SurveillanceInsight_2.0.2/Services/FinancialMkts_SurveillanceInsight_ServicesContent` directory.
2. Copy `ibmjgssprovider.jar` to the `/opt/ibm/InfoSphere_Streams/4.2.0.2/java/jre/lib` directory.

   Replace the existing `ibmjgssprovider.jar` file.

Run the Spark jobs on the IOP master node

PersistComm job

If Kerberos is enabled and Db2 security is enabled, run the following command:

```
processCommunication.sh
```

If Kerberos is not enabled and Db2 security is not enabled, run the following command:

```
/home/sifsuser/spark-2.1.1-hadoop2.7/bin/spark-submit --class
com.ibm.sifs.ecomm.PersistComm --master yarn --deploy-mode cluster --
executor-cores 3 --num-executors 10 --driver-memory 1g --executor-memory 2g
--jars /home/sifsuser/spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar --
conf="spark.driver.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
```

```
spark-2.1.1-hadoop2.7/jars/*"  --conf="spark.yarn.jars=/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar" --
conf="spark.executor.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/*"  /home/sifsuser/lib/ECommProcessing-2.0.1-
SNAPSHOT.jar /home/sifsuser/lib/sifs.spark.properties
```

PersistEmail job

If Kerberos is enabled and Db2 security is enabled, run the following command:

```
processEmail.sh
```

If Kerberos is not enabled and Db2 security is not enabled, run the following command:

```
/home/sifsuser/spark-2.1.1-hadoop2.7/bin/spark-submit --class
com.ibm.sifs.ecomm.PersistEmail --master yarn --deploy-mode cluster --
executor-cores 3 --num-executors 10 --driver-memory 1g --executor-memory 2g
--jars /home/sifsuser/spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar --
conf="spark.driver.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/*"  --conf="spark.yarn.jars=/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar" --
conf="spark.executor.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/*"  /home/sifsuser/lib/ECommProcessing-2.0.1-
SNAPSHOT.jar /home/sifsuser/lib/sifs.spark.properties
```

PersistChat job

If Kerberos is enabled and Db2 security is enabled, run the following command:

```
processChat.sh
```

If Kerberos is not enabled and Db2 security is not enabled, run the following command:

```
/home/sifsuser/spark-2.1.1-hadoop2.7/bin/spark-submit --class
com.ibm.sifs.ecomm.PersistChat --master yarn --deploy-mode cluster --
executor-cores 3 --num-executors 10 --driver-memory 1g --executor-memory 2g
--jars /home/sifsuser/spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar --
conf="spark.driver.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/*"  --conf="spark.yarn.jars=/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar" --
conf="spark.executor.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/*"  /home/sifsuser/lib/ECommProcessing-2.0.1-
SNAPSHOT.jar /home/sifsuser/lib/sifs.spark.properties
```

AnalyzeComm job

If Kerberos is enabled and Db2 security is enabled, run the following command:

```
analyzeComm.sh <date>
```

If Kerberos is not enabled and Db2 security is not enabled, run the following command:

```
/home/sifsuser/spark-2.1.1-hadoop2.7/bin/spark-submit --class
com.ibm.sifs.ecomm.AnalyzeComm --master yarn --deploy-mode cluster --
executor-cores 3 --num-executors 10 --driver-memory 2g --executor-memory 3g
--jars /home/sifsuser/spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar --
conf="spark.driver.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/*"  --conf="spark.yarn.jars=/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar" --
conf="spark.executor.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
```

```
spark-2.1.1-hadoop2.7/jars/*"  /home/sifsuser/lib/ECommProcessing-2.0.1-
SNAPSHOT.jar /home/sifsuser/lib/sifs.spark.properties <date>
```

ProfileAggregator job

Ensure that the `sifs.spark.properties` property is updated to the date for which the reference profile must be computed: `window=30`

If Kerberos is enabled and Db2 security is enabled, run the following command:

```
computeProfile.sh <date>
```

If Kerberos is not enabled and Db2 security is not enabled, run the following command:

```
/home/sifsuser/spark-2.1.1-hadoop2.7/bin/spark-submit --class
com.ibm.sifs.ecomm.ProfileAggregator --master yarn --deploy-mode cluster --
executor-cores 3 --num-executors 6 --driver-memory 4g --executor-memory 3g --
jars /home/sifsuser/spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar --
conf="spark.driver.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/*"  --conf="spark.yarn.jars=/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar" --
conf="spark.executor.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/*"  /home/sifsuser/lib/ECommProcessing-2.0.1-
SNAPSHOT.jar /home/sifsuser/lib/sifs.spark.properties <date>
```

PartyRiskScoring job

Ensure that the `sifs.spark.properties` property is updated to the date for which the reference profile must be computed:

```
PartyRiskDateWindow=90
SolrProxyURL=https://localhost:9443/SIFSServices/surveillanceui/v1/index/
update
```

If Kerberos is enabled and Db2 security is enabled, run the following command:

```
partyRiskScoring.sh
```

If Kerberos is not enabled and Db2 security is not enabled, run the following command:

```
/home/sifsuser/spark-2.1.1-hadoop2.7/bin/spark-submit --class
com.ibm.sifs.scoring.PartyRiskScoring --master yarn --deploy-mode cluster --
executor-cores 4 --num-executors 4 --driver-memory 2g --executor-memory 3g --
jars /home/sifsuser/spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar --
conf="spark.driver.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/*" --conf="spark.yarn.jars=/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/spark-yarn_2.11-2.1.1.jar" --
conf="spark.executor.extraClassPath=/home/sifsuser/lib/*:/home/sifsuser/
spark-2.1.1-hadoop2.7/jars/*"  /home/sifsuser/lib/PartyRiskScoring-2.0.1-
SNAPSHOT.jar /home/sifsuser/lib/sifs.spark.properties
```

Installing the e-comms component artifacts

To install the IBM Electronic Communication Surveillance Analytics component artifacts, you must run scripts on the BigData node computer.

Procedure

1. Log on to the BigData node computer.

2. Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_EComm_2.0.2/BigData/FinancialMkts_SurveillanceInsight_BigDataContent/ecomm/bin` directory.
3. Open the `build.properties` file and update the file with the appropriate values for your environment.
4. Run the following command: `sh Install_BigData.sh`

Installing the voice component artifacts

To install the IBM Voice Surveillance Analytics component artifacts, you must run scripts on the analytics node and BigData node computers.

Before you begin

Before you install the voice component, ensure that you do the following tasks:

- Ensure that the ffmpeg utility is installed on the computer where WebSphere Application Server is running.

  If you do not have ffmpeg installed:

  1. Download ffmpeg from http://ffmpeg.org/releases/ffmpeg-3.3.2.tar.bz2.
  2. Extract `ffmpeg-3.3.1.tar.bz2` into a directory.
  3. Follow the instructions in `INSTALL.md` to install the utility.
  4. Install YASM by following the instruction at http://www.linuxfromscratch.org/blfs/view/cvs/general/yasm.html.
  5. Install NASM by following the instructions at http://www.linuxfromscratch.org/blfs/view/svn/general/nasm.html.
- Install the following RPMs on the analytics node computer.
  - libxblas-1.0.248-1mamba.x86_64.rpm
  - atlas-3.10.1-12.el7.x86_64.rpm
  - atlas-devel-3.10.1-12.el7.x86_64.rpm
  - libpcap-devel-1.5.3-8.el7.x86_64.rpm

Procedure

1. Log on to the analytics node computer.
   a) Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Voice_2.0.2/Analytics/FinancialMkts_SurveillanceInsight_AnalyticsContent/Voice/bin` directory.
   b) Open the `build.properties` file and update the file with the appropriate values for your environment.
   c) Run the following command: `sh Install_Analytics.sh`
2. Log on to the BigData node computer.
   a) Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Voice_2.0.2/BigData/FinancialMkts_SurveillanceInsight_BigDataContent/Voice/bin` directory.
   b) Open the `build.properties` file and update the file with the appropriate values for your environment.
   c) Run the following command: `sh Install_BigData.sh`

Installing the Trade Surveillance component artifacts

To install the IBM Trade Surveillance Analytics component artifacts, you must run scripts on the analytics node and BigData node computers.

Procedure

1. Log on to the analytics node computer.
   a) Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Trade_2.0.2/Analytics/FinancialMkts_SurveillanceInsight_AnalyticsContent/Trade/bin` directory.
   b) Open the `build.properties` file and update the file with the appropriate values for your environment.
   c) Run the following command: `sh Install_Analytics.sh`
2. Log on to the BigData node computer.
   a) Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Trade_2.0.2/BigData/FinancialMkts_SurveillanceInsight_BigDataContent/Trade/bin` directory.
   b) Open the `build.properties` file and update the file with the appropriate values for your environment.
   c) Run the following command: `sh Install_BigData.sh`

Run streams on multiple hosts

You can add more resources for your existing InfoSphere Streams server. If you add resources, you must create a streams instance to use the multiple resources, and then run the voice streams job using the instance with multiple resources.

Note:

You must have a complete installation of InfoSphere Streams available. In these tasks, the host is referred to as streams-server. In InfoSphere Streams, a resource is a physical host.

You must ensure that the streams-server host and all of the resources use the same operating system level.

Adding new physical hosts to InfoSphere Streams

You must add the new physical hosts to Streams by using the Streams console.

Procedure

1. On each host that you want to add, run the following commands to create a streamsadmin user and streamsadmin group.

   ```
   groupadd streamsadmin
   ```

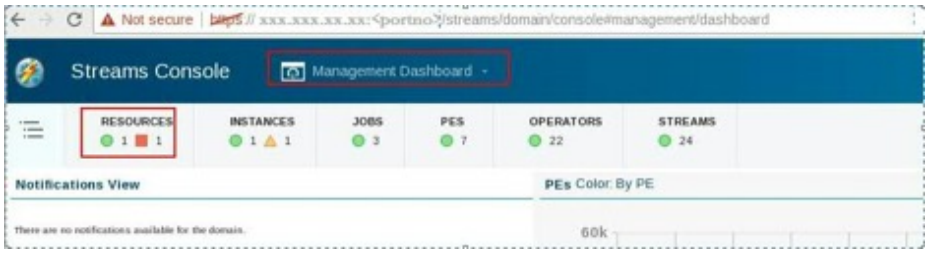
   ```
   useradd streamsadmin  -g streamsadmin
   ```

   You must also set a password for the streamsadmin user.
2. On each host that you want to add, install the following RPMs:
   - atlas-3.10.1-12.el7.x86_64.rpm
   - atlas-devel-3.10.1-12.el7.x86_64.rpm

3. On the streams-server (your existing InfoSphere Streams server), change to the streamsadmin user, and run the following command:

```
streamtool geturl https://<streams-server host IP>:<port>/streams/domain/
console
```

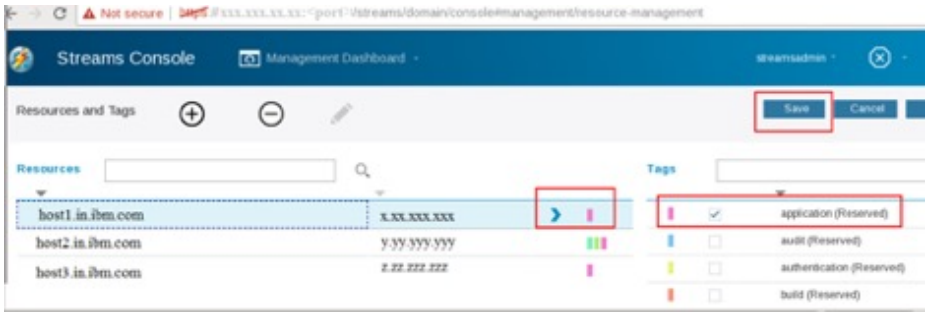
4. Go to the URL in a web browser, and click **RESOURCES**.



5. Hover over **RESOURCES**, and click **Manage Resources and Tags**.
6. Click the plus sign (**+**), and select **Add Resource**.
7. Follow the instructions that are provided under **If IBM Streams is not installed on the host** to download a `tar` file.
8. Copy the `tar` file to each new host that you are adding.
9. As the streamsadmin user, run **dependency_checker.sh** under **StreamsDomainHost**.

   You must run the script on each new resource. If the dependency script fails with errors, you must correct the errors and rerun the `dependency_checker.sh` script.

   Some common errors are described in Appendix B, “Troubleshooting,” on page 57.
10. As the root user, run the `streamsdomainhostsetup.sh` script on each new resource.
11. In the Streams Console, tag each resource as **application**.



12. Replace the `/opt/ibm/InfoSphere_Streams/4.2.0.2/java/jre/lib/ibmjgssprovider.jar` on each new resource with the `ibmjgssprovider.jar` that is provided with the installer.

Creating a Streams instance with multiple resources

You create a Streams instance by using the command line as the streamsadmin user.

For more information, see Resource specification options.

Procedure

1. Change to the streamsadmin user.
2. Run the following command to create the instance:

```
streamtool mkinstance -i SIInstance --hosts
host1.in.ibm.com,host2.in.ibm.com,host3.in.ibm.com
```

The resource list should contain resource names for both the streams-server and the new resources.

Ensure that you use the FQDN for the resources.

3. Run the following command to verify that the instance was created with multiple resources:

```
instance -i SIInstance
```

```
lsavailablehosts
```

4. Restart the Streams domain and instance.

Running the WAVAdaptor Streams job

Procedure

1. As the streamsadmin user, copy the artifacts that are mentioned against the CONFIGFILE, MODELFILE, and CONFIGPATH variables in `submitjob.sh` across all of the resources and the streams-server.
2. As the streamsadmin user, create the folder that is used in the DATAPATH value in the `submitjob.sh` across all of the resources and the streams-server.
3. Ensure that the `conf` directory on the HADOOP server is available at $HADOOP_HOME. $HADOOP_HOME can be any valid folder across all resources and the streams-server.

   You set the $HADOOP_HOME value in the streamsadmin user's `~/.bashrc` file:

```
export HADOOP_HOME=<path to the directory that contains the conf folder>
```

4. Copy the `/etc/krb5.conf` across all of the resources and the streams-server.

   Ensure that the `krb5.conf` file points to the correct HADOOP server.
5. As the streamsadmin user, copy the artifacts that are mentioned against the HDFSAUTHKEYTAB variable in `submitjob.sh` across all of the resources and the streams-server.

Running the PCAP Streams job

Procedure

1. As the streamsadmin user, copy the artifacts that are mentioned against the PARAMSET_FILE, PACKAGE_FILE, and CONFIGPATH variables in `submitPCAPSpeech.sh` and `submitSpeechRoute.sh` across all of the resources and the streams-server.
2. As the streamsadmin user, create the directories that are used in the DIAGNOSTICS_PATH and SPEECHDIRROOTvalues in `submitPCAPSpeech.sh` and `submitSpeechRoute.sh` across all of the resources and the streams-server.
3. As the streamsadmin user, create the directories that are mentioned against the ADDLSPEECHDIRS variable in `submitSpeechRoute.sh` across all of the resources and the streams-server
4. Ensure that the directory that is mentioned against the SPEECHDIRROOT variable in `submitSpeechRoute.sh` is shared over the network with read and write access.

   The voice files that are captured from communication networks are written to SPEECHDIRROOT as PCAP files by the RouteSpeech.spl job. The PCAP files are the read by `PCAPSpeech.spl` from SPEECHDIRROOT for further processing.
5. Ensure that the `conf` directory on the HADOOP server is available at $HADOOP_HOME. $HADOOP_HOME can be any valid folder across all resources and the streams-server.

   You set the $HADOOP_HOME value in the streamsadmin user's `~/.bashrc` file:

```
export HADOOP_HOME=<path to the directory that contains the conf folder>
```

6. Copy the `/etc/krb5.conf` across all of the resources and the streams-server.

Ensure that the `krb5.conf` file points to the correct HADOOP server.

7. As the streamsadmin user, copy the artifacts that are mentioned against the HDFSAUTHKEYTAB variable in `submitjob.sh` across all of the resources and the streams-server.

Configuring the voice language model in Surveillance Insight Design Studio

You must configure the voice language model in the Surveillance Insight Design Studio.

Procedure

1. On IBM Bluemix®, create an instance of the Speech 2 Text service, and note the service credentials, the user name and password, that are needed to access the S2T service.
2. In the directory where the voice components are extracted, build the SpeechTraining job by running the `build.sh` script. Note the JMX URL for the Streams instance and the `.sab` file location of SpeechTraining in the output directory.
3. Create a directory that is shared between IBM Streams and the WebSphere Liberty host. The Liberty Service and the Streams SpeechTraining job must have read and write access to the shared directory for the model training.
   a) Create a directory in `/home/streamsadmin` for the training artifacts.
   For example, `/home/streamsadmin/training`
   b) Make this directory accessible on the Service node for WebSphere Liberty with read and write access.
4. Extract `WER.tar` from `Analytics/Voice/word.error.rate` to the shared directory that you created.
   For example, `/home/streamsadmin/training`
5. Update the Watson™ service credentials and the Streams instance details in the WebSphere Liberty server configuration file: `/opt/ibm/wlp/usr/servers/SIFSServer/server.env`

Installing the Complaints Surveillance component artifacts

To install the Complaints Surveillance component artifacts, you must run scripts on the BigData node computer.

Procedure

Log on to the BigData node computer.

a) Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Trade_2.0.2/BigData/FinancialMkts_SurveillanceInsight_BigDataContent/Trade/bin` directory.
b) Open the `build.properties` file and update the file with the appropriate values for your environment.
c) Run the following command: `sh Install_BigData.sh`

Install IBM HTTP Server and the WebSphere plug-in

You must install IBM HTTP Server and the WebSphere plug-in component.

Installing IBM Installation Manager

You must install IBM Installation Manager so that you can install WebSphere Application Server. Install IBM Installation Manager and WebSphere Application Server to provide a front end for users to access IBM Surveillance Insight for Financial Services.

Procedure

1. Go to the directory where you decompressed the IBM Installation Manager installation files.
2. Enter the following command to start the installer:

   ```
   ./userinstc -acceptLicense
   ```
3. Follow the steps to install IBM Installation Manager.

Installing IBM HTTP Server and the WebSphere Plug-in

You must install IBM HTTP Server and the WebSphere plug-in component.

IBM Surveillance Insight for Financial Services uses:

- IBM WebSphere Application Server V9.0 Supplements - IBM HTTP Server
- IBM WebSphere Application Server V9.0 Supplements - Web Server Plugins

Procedure

1. Log into the server as root.
2. Start IBM Installation Manager from the **Application Browser** or go to the `/opt/IBM/InstallationManager/eclipse` directory, and run `./IBMIM`.
3. Click **File** > **Preferences**.
4. Click **Add Repository**.
5. Browse to the location where you decompressed the IBM HTTP Server, IBM Web Server Plug-ins, and the IBM SDK installation files.
6. In the directory where you decompressed the installation files, select `repository.config`, click **OK** to close the dialog.
7. Click **Add Repository**.
8. Browse to the location where you decompressed the WebSphere Customization Tool installation files.
9. In the directory where you decompressed the installation files, select `repository.config`, click **OK** to close the dialog.
10. In IBM Installation Manager, click **Install**.
11. Select **Web Server Plug-ins for IBM WebSphere Application Server** and **WebSphere Customization Toolbox**, and ensure that the underlying components are selected as well.
12. Click **Next**, and follow the steps in IBM Installation Manager to install the product.
13. When prompted for **Which program do you want to start?**, select **None**, and click **Finish**.

Configuring IBM HTTP Server and IBM WebSphere Plug-In

You must configure IBM HTTP Server and the IBM WebSphere Plug-in for IBM Surveillance Insight for Financial Services.

Procedure

1. Log in to the server as root.
2. Create an ihsmgr user and group.

   ```
   groupadd -g 1511 ihsmgr
   ```

   ```
   useradd -u 1511 -g 1511 ihsmgr
   ```

3. Run the following command to configure the admin server.

```
"/opt/IBM/HTTPServer/bin/setupadm" -usr ihsmgr -grp ihsmgr -cfg
"/opt/IBM/HTTPServer/conf/httpd.conf" -adm "/opt/IBM/HTTPServer/conf/
admin.conf"
```

   If you used a directory other than `/opt/IBM/HTTPServer` for the HTTP Server installation, ensure that you change the paths in the command to the correct paths for your environment.

4. Set the admin password:

```
"/opt/IBM/HTTPServer/bin/htpasswd" -b "/opt/IBM/HTTPServer/conf/
admin.passwd" "ihsmgr" "password"
```

5. Set the admin port in the `admin.conf` file:

```
sed -i s#@@AdminPort@@#"8008"#g "/opt/IBM/HTTPServer/conf/admin.conf"
```

6. Create an SSL certificate for IBM HTTP Server.
   a) Create a directory for the certificate:

```
rm -rf "/opt/IBM/HTTPServer/cert"
```

```
mkdir "/opt/IBM/HTTPServer/cert"
```

   b) Create a keystore:

```
"/opt/IBM/HTTPServer/bin/gskcmd" -keydb -create -db "/opt/IBM/
HTTPServer/cert/sifs_ihs_ks.kdb" -pw "password" -type cms -expire
"3650" -stash
```

   c) Create a self-signed certificate:

```
"/opt/IBM/HTTPServer/bin/gskcmd" -cert -create -db "/opt/IBM/
HTTPServer/cert/sifs_ihs_ks.kdb" -pw "password" -size "2048" -dn
"CN=localhost,OU=SIFS,O=IBM,C=US" -label "sifs_ihs_cert" -default_cert
yes -expire 3650
```

   Ensure that you change the dn value to suit your environment.

7. Back up the original `httpd.conf` file.

```
cp "/opt/IBM/HTTPServer/conf/httpd.conf" "/opt/IBM/HTTPServer/conf/
httpd.conf.original"
```

8. Open the `httpd.conf` file in a text editor, and change the following values:
   a) Search for `Listen 80`, and comment out the line.
   b) Search for `ServerName`, and change the port number to 443, the default secure port for the web server.
   c) Add the following lines to the end of the file:

```
LoadModule rewrite_module modules/mod_rewrite.so
LoadModule ibm_ssl_module modules/mod_ibm_ssl.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so
Listen 443
SSLCheckCertificateExpiration 30
<VirtualHost *:443>
 SSLEnable

 SSLProtocolEnable TLSv12
 SSLCipherSpec ALL -SSL_RSA_WITH_RC4_128_SHA  -
```

```
SSL_RSA_WITH_3DES_EDE_CBC_SHA
-TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA -
TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA

 #For POODLE TLS attack (CVE-2014-8730)
SSLAttributeSet 471 1

 FileETag None

 Header always set Strict-Transport-Security "max-age=31536000;
includeSubDomains; preload"
 Header always set Content-Security-Policy "default-src https: data:
'unsafe-inline' 'unsafe-eval'"
 Header always append X-Frame-Options SAMEORIGIN
 Header set X-XSS-Protection "1; mode=block"
 Header set X-Content-Type-Options nosniff
 Header edit Set-Cookie ^(.*)$ $1;Secure
 Header set Cache-Control "no-cache, no-store, must-revalidate"
 Header set Pragma "no-cache"
 Header set Expires 0
</VirtualHost>
KeyFile /opt/IBM/HTTPServer/cert/sifs_ihs_ks.kdb
SSLStashfile /opt/IBM/HTTPServer/cert/sifs_ihs_ks.sth
SSLDisable

#Removing Server Version Banner
AddServerHeader Off
ServerTokens Prod
ServerSignature Off

#Code to rewrite/redirect http traffic to https
RewriteEngine On
RewriteCond %{SERVER_PORT} =80
RewriteRule ^(.*) https://%{SERVER_NAME}%{REQUEST_URI} [L,R]

#Deny certain operations
RewriteCond %{REQUEST_METHOD} ^(TRACE|TRACK) [NC]
RewriteRule .? - [F]

LoadModule was_ap24_module /opt/IBM/WebSphere/Plugins/bin/64bits/
mod_was_ap24_http.so
WebSpherePluginConfig /opt/IBM/HTTPServer/conf/plugin-cfg.xml
```

d) Ensure that only one line says `Listen 443`. If there are multiple lines, delete the other lines.

e) To stop the directory traversal threat, under `Directory`, set `Options -Indexes` wherever appropriate.

f) Save and close the `httpd.conf` file.

9. Grant the ihsmgr user permission for the configuration files:

```
chown ihsmgr:ihsmgr "/opt/IBM/HTTPServer/conf/httpd.conf"
```

```
chown ihsmgr:ihsmgr "/opt/IBM/HTTPServer/conf/admin.conf"
```

10. Start the IBM HTTP Server and the admin processes:

```
/opt/IBM/HTTPServer/bin/adminctl start
```

```
/opt/IBM/HTTPServer/bin/apachectl start
```

11. Verify that the server is running by accessing the URL: `https://`*<hostname>*

Integrating IBM WebSphere Liberty ND with IBM HTTP Server

You must create a configuration file to integrate IBM WebSphere Liberty ND with IBM HTTP Server.

Procedure

1. Create the following file: `/opt/IBM/HTTPServer/conf/plugin-cfg.xml`

   If you used a different path for HTTP Server, ensure that you use the appropriate path.

2. Add the following content to the file:

```
<?xml version="1.0" encoding="UTF-8"?>
<Config ASDisableNagle="false" AcceptAllContent="false"
AppServerPortPreference="HostHeader" ChunkedResponse="false"
    FIPSEnable="false" IISDisableNagle="false" IISPluginPriority="High"
IgnoreDNSFailures="false" RefreshInterval="60"
    ResponseChunkSize="64" SSLConsolidate="false"
TrustedProxyEnable="false" VHostMatchingCompat="false">
    <Log LogLevel="Error" Name="/opt/IBM/HTTPServer/logs/
http_plugin.log" />
    <Property Name="ESIEnable" Value="true" />
    <Property Name="ESIMaxCacheSize" Value="1024" />
    <Property Name="ESIInvalidationMonitor" Value="false" />
    <Property Name="ESIEnableToPassCookies" Value="false" />
    <Property Name="PluginInstallRoot" Value="/opt/IBM/WebSphere/
Plugins" />
    <!-- Configuration generated using
httpEndpointRef=defaultHttpEndpoint -->
    <!-- The default_host contained only aliases for endpoint
defaultHttpEndpoint. The generated VirtualHostGroup
        will contain only configured web server ports: webserverPort=80
webserverSecurePort=443 -->
    <VirtualHostGroup Name="default_host">
        <VirtualHost Name="*:80" />
        <VirtualHost Name="*:443" />
    </VirtualHostGroup>
    <ServerCluster CloneSeparatorChange="false" GetDWLMTable="false"
IgnoreAffinityRequests="true"
        LoadBalance="Round Robin" Name="SIFSServer_default_node_Cluster"
PostBufferSize="0" PostSizeLimit="-1"
        RemoveSpecialHeaders="true" RetryInterval="60"
ServerIOTimeoutRetry="-1">
        <Server CloneID="c682f5f6-e290-4397-a978-eae8b4a62bd3"
ConnectTimeout="5" ExtendedHandshake="false"
            LoadBalanceWeight="20" MaxConnections="-1"
Name="default_node_SIFSServer" ServerIOTimeout="900"
            WaitForContinue="false">
            <Transport Hostname="hostname.domain.com" Port="9080"
Protocol="http" />
            <Transport Hostname="hostname.domain.com" Port="9443"
Protocol="https">
                <Property Name="keyring" Value="/opt/IBM/HTTPServer/cert/
plugin-key.kdb" />
                <Property Name="stashfile" Value="/opt/IBM/HTTPServer/
cert/plugin-key.sth" />
            </Transport>
        </Server>
        <PrimaryServers>
            <Server Name="default_node_SIFSServer" />
        </PrimaryServers>
    </ServerCluster>
    <UriGroup Name="default_host_SIFSServer_default_node_Cluster_URIs">
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/SIFSServices/*" />
```

```
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/SIFSModelServices/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/IBMJMXConnectorREST/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/ibm/saml20/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/ibm/api/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/ibm/adminCenter/
explore-1.0/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/ibm/adminCenter/
serverConfig-1.0/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/surveillancetool/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/surveillance/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/adminCenter/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/ui.complaintsdashboard/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/complaintsservices/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/CommServices/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/SIFSVoiceDataService/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/SIFSVoiceIngestionService/*" />
        <Uri AffinityCookie="JSESSIONID"
AffinityURLIdentifier="jsessionid" Name="/api/*" />
    </UriGroup>
    <Route ServerCluster="SIFSServer_default_node_Cluster"
UriGroup="default_host_SIFSServer_default_node_Cluster_URIs"
        VirtualHostGroup="default_host" />
</Config>
```

Ensure that you change the *hostname.domain.com* value to the fully qualified domain name for the WebSphere Liberty server. This value is usually the hostname of the Kubernetes master node computer.

3. Save and close the file.
4. Copy the Liberty keystore on the IBM HTTP Server.
5. Set the LD_LIBRARY_PATH.

```
export LD_LIBRARY_PATH="/opt/IBM/HTTPServer/gsk8/lib64/"
```

6. Run the following command to convert the WebSphere Liberty JKS keystore to PKCS12:

```
"/opt/IBM/HTTPServer/java/8.0/bin/keytool" -importkeystore -srckeystore
"/opt/IBM/HTTPServer/cert/sifs_universal_ks.jks" -srcstorepass "password"
-destkeystore "/opt/IBM/HTTPServer/cert/sifs_universal_ks.p12" -
srcstoretype JKS -deststoretype PKCS12 -deststorepass "password" -
srcalias "sifskey" -destalias "sifskey" -noprompt
```

Ensure that you change the password and paths to suit your environment.

7. Convert the WebSphere Liberty intermediate PKCS12 keystore to CMS keystore for IHS gskit:

```
"/opt/IBM/HTTPServer/gsk8/bin/gsk8capicmd_64" -keydb -convert -db
"/opt/IBM/HTTPServer/cert/sifs_universal_ks.p12" -pw "password" -type p12
-target "/opt/IBM/HTTPServer/cert/plugin-key.kdb" -expire "1800" -stash
```

Ensure that you change the password and paths to suit your environment.

You can delete the interermediate PKCS12 keystore by using the following command:

```
rm -f "/opt/IBM/HTTPServer/cert/sifs_universal_ks.p12"
```

8. Restart IBM HTTP Server:

```
/opt/IBM/HTTPServer/bin/apachectl restart
```

9. Verify the configuration by accessing the product URLs:
   - `https://<ihs_hostname>/surveillance/dashboard/index.html`
   - `https://<ihs_hostname>/surveillancetool`
   - `https://<ihs_hostname>/ui.complaintsdashboard`

Chapter 5. Configure SAML security

IBM Surveillance Insight for Financial Services uses SAML 2.0 to allow single sign-on.

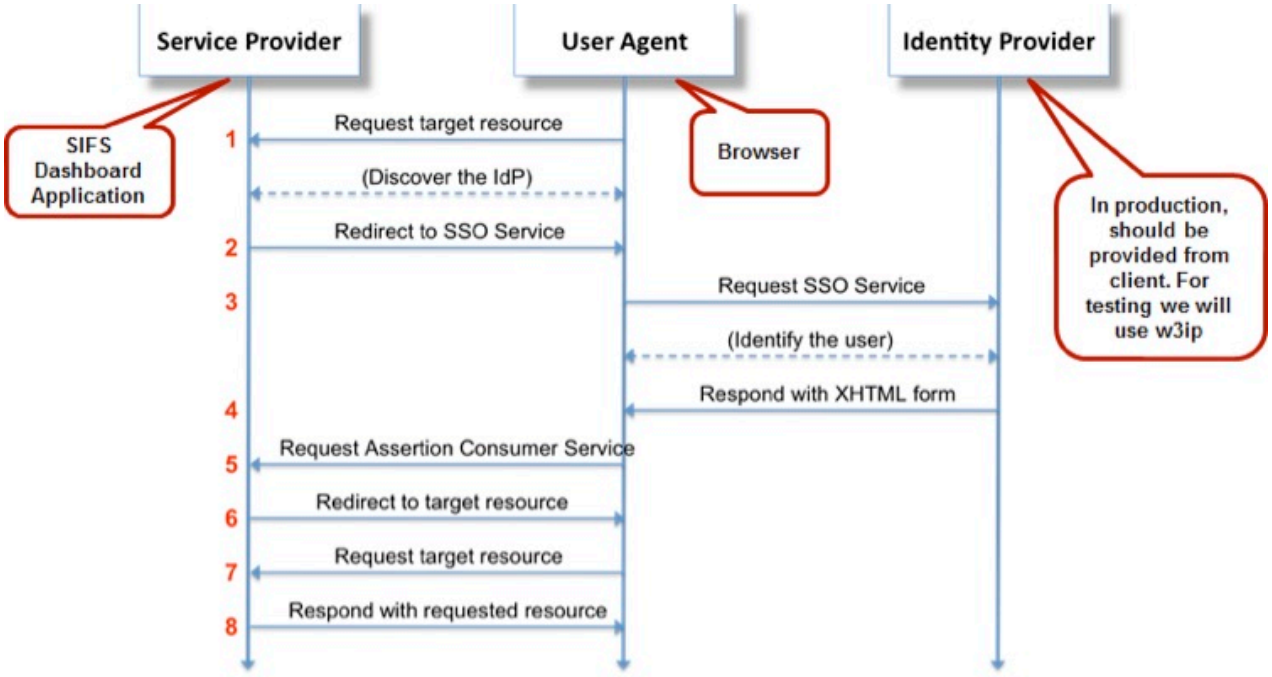


Figure 5: SAML configuration

Configuring WebSphere Liberty ND server for SAML

You must enable web SSO for the WebSphere Liberty ND server.

Procedure

1. Confirm that the keystore was created for Liberty ND. If not, use the following steps to create one.
   a) Open a command line terminal and go to the Java directory that is installed with WebSphere Liberty.
   b) Run following command to create a keystore and a self-signed key.

```
bin/keytool -genkeypair -alias "sifskey" -keyalg RSA -sigalg
SHA256withRSA -keysize 2048 -storetype jks -keystore
"<path_to_liberty>/usr/servers/SIFSServer/resources/security/
sifs_universal_ks.jks" -dname "CN=localhost,OU=SIFSServer,O=ibm,C=US"
```

   Ensure that you use the appropriate paths for your environment.

   Note: Use the same password for the keystore and the self-signed key.
2. Create a file that is named `server_saml.xml`, and add the following content to the file:

```
<server>
    <featureManager>
        <feature>samlWeb-2.0</feature>
    </featureManager>

    <samlWebSso20 id="defaultSP" enabled="true"
```

```
authFilterRef="samlAuthFilter"
        idpMetadata="${server.config.dir}/resources/security/
idpMetadata.xml" httpsRequired="true"
        signatureMethodAlgorithm="SHA256" spHostAndPort="$
{env.sifs_saml_spHostAndPort}" keyStoreRef="sifsKeyStore"
        keyAlias="sifskey" mapToUserRegistry="No"
createSession="false"></samlWebSso20>

    <authFilter id="samlAuthFilter">
        <requestUrl id="sifsURL" urlPattern="/surveillance|/
surveillancetool|/ui.complaintsdashboard"
            matchType="contains" />
    </authFilter>

    <authCache initialSize="100" maxSize="50000" timeout="15m" />
</server>
```

Enter the appropriate value for *spHostAndPort*. If you have a proxy server, such as IBM HTTP Server as a front end to the WebSphere Liberty application server, then use the IP address of proxy server.

Modify the value of `signatureMethodAlgorithm` to match the identity provider.

More attributes for `samlWebSso20` can be configured depending upon the requirements of the identity provider. For more information about all of the available attributes, see Configuring SAML Web Browser SSO in Liberty (https://www.ibm.com/support/knowledgecenter/en/SSAW57_liberty/com.ibm.websphere.wlp.nd.multiplatform.doc/ae/twlp_config_saml_web_sso.html).

3. Edit the `server.xml` file for WebSphere Liberty ND to include the `server_saml.xml` file.

   For example, add the following line to the `server.xml` file:

```
<include location="/path/server_saml.xml" />
```

4. Confirm that the IBM Surveillance Insight for Financial Services applications WAR files are built for accepting SAML.

   You must confirm this for `ui.sifsworkbench-2.0.2-SNAPSHOT.war`, `ui.sifsdesignstudio-2.0.2-SNAPSHOT.war`, and `ui.complaintsdashboard-2.0.2-SNAPSHOT.war`.

   To confirm this, open the `web.xml` of each `war` file, and ensure that the following section is enabled:

```
<filter>
    <filter-name>SAMLLoginFilter</filter-name>
    <filter-class>com.ibm.sifs.ui.SAMLLoginFilter</filter-class>
</filter>
<filter-mapping>
    <filter-name>SAMLLoginFilter</filter-name>
    <url-pattern>/*</url-pattern>
</filter-mapping>
```

   Also, ensure that the following section is disabled:

```
<login-config>
    <auth-method>FORM</auth-method>
    <realm-name>BasicRealm</realm-name>
    <form-login-config>
        <form-login-page>/formBasedLogin.jsp</form-login-page>
        <form-error-page>/formBasedLogin.jsp?Retry=true</form-error-page>
    </form-login-config>
</login-config>

<filter>
    <filter-name>FormBasedLoginFilter</filter-name>
```

```
    <filter-class>com.ibm.sifs.ui.FormBasedLoginFilter</filter-class>
</filter>
<filter-mapping>
    <filter-name>FormBasedLoginFilter</filter-name>
    <url-pattern>/*</url-pattern>
</filter-mapping>
```

5. Modify security role application binding of the UI application `war` configuration.

   You must modify this for `server_sifsworkbench_war.xml`, `server_sifsdesignstudio_war.xml`, and `server_sifscomplaintsdashboard_war.xml`.

   Provide appropriate values for group name and group access details from identity provider:

```
<server>
    <webApplication id="ui.sifsworkbench"
location="ui.sifsworkbench-2.0.2-SNAPSHOT.war" name="ui.sifsworkbench"
        contextRoot="/surveillance">
        <classloader commonLibraryRef="sifslib" />

        <application-bnd>
            <security-role name="Compliance Officer">
                <group name="<group_name_from_idp>" access-
id="group:<access_details_from_idp>"/>
            </security-role>
            <security-role name="Case Officer">
                <group name="<group_name_from_idp>" access-
id="group:<access_details_from_idp>"/>
            </security-role>
            <security-role name="L1/L2 Officer">
                <group name="<group_name_from_idp>" access-
id="group:<access_details_from_idp>"/>
            </security-role>
        </application-bnd>
    </webApplication>
</server>
```

6. Get the identity provider metadata and rename it to `idpMetadata.xml`. This file is case-sensitive.
7. Copy `idpMetadata.xml` to the server's `resources/security` directory.
8. Restart the server:

```
bin/server start <server_name>
```

9. Download and save the metadata file (also called the service provider metadata file) by accessing `https://<spHostAndPort>/ibm/saml20/defaultSP/samlmetadata` in a web browser.
10. Import this service metadata file into the identity provider application and activate it.

Chapter 6. Use SLM tags to track licensing

Software License Metric (SLM) tag files provide a standardized capability for a product to report its consumption of license metrics (resources that are related to the use of the software asset). After SLM is enabled in a product, a runtime XML file is generated to self-report its license usage. The SLM tag files are based on the ISO/IEC 19770-4 standard draft for Resource Utilization Measurement.

Secure communication via SLM tag

You can secure the communication of the SLM tag by importing the `SIDB2.arm` certificate that you used earlier in the installation into a keystore for the SLM tag generator utility. For example, use the following command:

```
keytool -keystore /home/sifsuser/security/SISLMTagClient.jks -alias
DB2SLMTag -import -file /home/db2inst1/SIDB2.arm
```

And then enter the password that you used.

SLM tag files

The SLM tag files are stored in XML format, and new metric records are appended to the end of the file.

The following is a sample SLM tag for the voice component:

```
<SchemaVersion>2.1.1</SchemaVersion>
  <SoftwareIdentity>
     <PersistentId>a490d40f839049ea881d9aedf8b3d60f</PersistentId>
        <Name>IBM Voice Surveillance Analytics</Name>
        <InstanceId>/home/test/</InstanceId>
 </SoftwareIdentity>
         <Metric logTime="2017-05-17T01:01:41+05:30">
         <Type>FEED</Type>
         <SubType>TOTAL_VOICE_SECONDS</SubType>
         <Value>1821</Value>
         <Period>
                 <StartTime>2017-04-17T01:01:41+05:30</StartTime>
                <EndTime>2017-05-17T01:01:41+05:30</EndTime>
        </Period>
       </Metric>
```

The following is a sample SLM tag for the trade component:

```
<SchemaVersion>2.1.1</SchemaVersion>
   <SoftwareIdentity>
      <PersistentId>c6ede63c6002493f82281c89982fcc32</PersistentId>
      <Name>IBM Trade Surveillance Analytics</Name>
      <InstanceId>/home/test/</InstanceId>
     </SoftwareIdentity>
          <Metric logTime="2017-05-17T01:06:23+05:30">
          <Type>USER</Type>
          <SubType>NO_OF_PARTY</SubType>
          <Value>151</Value>
          <Period>
                <StartTime>2017-05-16T01:06:23+05:30</StartTime>
                <EndTime>2017-05-17T01:06:23+05:30</EndTime>
          </Period>
         </Metric>
```

The following is a sample SLM tag for the e-comm component:

```
<SchemaVersion>2.1.1</SchemaVersion>
   <SoftwareIdentity>
      <PersistentId>fe953daa1dbc4446905c4b3dd21e8f81</PersistentId>
      <Name>IBM Electronic Communication Surveillance Analytics</Name>
      <InstanceId>/home/test/</InstanceId>
   </SoftwareIdentity>
  <Metric logTime="2017-05-17T01:07:06+05:30">
       <Type>USER</Type>
       <SubType>NO_OF_PARTY</SubType>
       <Value>151</Value>
       <Period>
           <StartTime>2017-05-16T01:07:06+05:30</StartTime>
           <EndTime>2017-05-17T01:07:06+05:30</EndTime>
      </Period>
</Metric>
```

When IBM Surveillance Insight for Financial Services is installed, the SLM tag files (`*.slmtag`) are available on the data node computer in the `/var/ibm/common/slm` directory.

The SLM scripts are configured to run as cron jobs.

Updating your software tag file if you change product usage

If you change your usage of IBM Surveillance Insight for Financial Services, such as to a non-production environment from a production environment, you must switch the software tags for your installation.

Follow these steps to change your usage to non-production or to change your usage back to production.

Procedure

1. Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/Analytics/tag_prod_nonprod` directory.
2. Open the `build.properties` file in the text editor, modify the settings, and save the file.
3. Run the following command:

   `./Switch_Tag_SIFS.sh`
4. Repeat steps 1 -3 on each IBM Surveillance Insight for Financial Services node.

Chapter 7. Load sample data

Sample data is provided for trade, e-comm, and voice surveillance.

Loading trade sample data

Sample data is provided for the off-market, spoofing, and pump-and-dump use cases.

Procedure

1. Load the off-market sample data:
   a) Log on to the HDFS node as the sifsuser.
   b) Run the following commands:

```
hdfs dfs -mkdir /user/sifsuser/transactions/
```

```
hdfs dfs -mkdir /user/sifsuser/marketReference/
```

```
hdfs dfs -put /opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/
Database/Sample_Data/OffMarketData/Off-MarketData/
transactions_2017-02-22.csv /user/sifsuser/transactions/
```

```
hdfs dfs -put /opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/
Database/Sample_Data/OffMarketData/Off-MarketData/
marketReference_2017-02-22.csv /user/sifsuser/marketReference/
```

2. Load the spoofing sample data:
   a) Log on to the HDFS node as the sifsuser, and go to the `/home/sifsuser/data` directory.
   b) Run the following commands:

```
hdfs dfs -put /opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/
Database/Sample_Data/SpoofingData/Trade_2017-04-12.csv /user/sifsuser/
trade/
```

```
hdfs dfs -put /opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/
Database/Sample_Data/SpoofingData/Quote_2017-04-12.csv /user/sifsuser/
quote/
```

```
hdfs dfs -put /opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/
Database/Sample_Data/SpoofingData/Order_2017-04-12.csv /user/sifsuser/
order/
```

```
hdfs dfs -put /opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/
Database/Sample_Data/SpoofingData/Execution_2017-04-10.csv /user/
sifsuser/execution/
```

3. Load the pump-and-dump sample data:
   a) Log on as the sifsuser.
   b) Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/Database/Sample_Data/PumpDumpSolution/PDZDataSet/Pnd_dailydata` directory.
   c) Edit the `PnD Load.sh` file and change the parameter to point to the current directory.

d) Run the script: `sh PnD Load.sh`

Loading e-comm sample data

Sample data is provided for policy, email, and chat data.

Procedure

1. Load the policy data:
   a) Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/Database/Sample_Data/EComm/Policy` directory.
   b) Run the following commands:

```
curl -k -H 'Content-Type: application/json' -H 'source:Actiance' -X
POST --data-binary @policy.json -v --user actiance1:actpwd1 --digest
https://localhost:9443/CommServices/ecomm/policy
```

```
curl -k -H 'Content-Type: application/json' -H 'source:Actiance' -X
POST --data-binary @policy2.json -v --user actiance1:actpwd1 --digest
https://localhost:9443/CommServices/ecomm/policy
```

2. Load the email data:
   a) Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/Database/ /sampledata/snapshots` directory.
   b) Open `generatecmd.sh` in a text editor, and ensure that the user name and password for the REST URL is correct for your environment. The script uses ibmrest1/ibmrest@pwd1 as the default.
   c) The script prompts for the REST URL to ingest the email data. Ensure that the IP address and host name are correct for your environment: `https://localhost:9443/CommServices/data/email`
   d) Run the command: `./generatecmd.sh`.

      The command creates a file that is named `ingest_emails.sh`.
   e) Give executable permissions to `ingest_emails.sh`.
   For example, enter `chmod +x ingest_emails.sh`
   f) Run the command: `./ingest_emails.sh`.
3. Load the chat data:
   a) Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/Database/Sample_Data/EComm/SnapshotChats` directory.
   b) Run the following commands:

```
curl -k -H 'Content-Type: text/plain' -H 'source:Actiance' -X POST --
data-binary @snapshot_chat_1.txt -v --user actiance1:actpwd1 --digest
https://localhost:9443/CommServices/data/chat
```

```
curl -k -H 'Content-Type: text/plain' -H 'source:Actiance' -X POST --
data-binary @snapshot_chat_2.txt -v --user actiance1:actpwd1 --digest
https://localhost:9443/CommServices/data/chat
```

```
curl -k -H 'Content-Type: text/plain' -H 'source:Actiance' -X POST --
data-binary @snapshot_chat_3.txt -v --user actiance1:actpwd1 --digest
https://localhost:9443/CommServices/data/chat
```

4. Load the email and chat data body.
   a) Go to the `Base/Database/Sample_Data/EComm` directory.

b) Copy the `commevidence` folder to your HTTP server document root folder.

Loading voice sample data

Sample data is provided for voice surveillance.

Procedure

1. Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_2.0.2/Database/Sample_Data/Voice/` directory where you ran the installer.
2. Open `processvoice.sh` in an editor.
3. Run each command that is listed in the file separately.

Appendix A. Accessibility features

Accessibility features help users who have a physical disability, such as restricted mobility or limited vision, to use information technology products.

For information about the commitment that IBM has to accessibility, see the IBM Accessibility Center (www.ibm.com/able).

HTML documentation has accessibility features. PDF documents are supplemental and, as such, include no added accessibility features.

Appendix B. Troubleshooting

This section provides troubleshooting information.

CDISI5060E No default Java found

You receive the following message: `CDISI5060E No default Java found`.

To resolve this error, install Java version 1.6 or later and set it as the default version. Then, try the command again.

Update the PATH variable in your `.bashrc` file to point to the JAVA location.

```
export PATH=<location of jre/bin directory>:$PATH
```

CDISI3059W You may be running a firewall which may prevent communication between the cluster hosts

You receive the following error message: `Warning: CDISI3059W You may be running a firewall which may prevent communication between the cluster hosts`

To resolve this, run the following command to stop the firewall service, and try the command again:

```
systemctl stop firewalld
```

For more information, see Firewall configuration guidelines for IBM Streams.

CDISI5070E The perl-XML-Simple software dependency is not installed

You receive the following error message: `Error: CDISI5070E The perl-XML-Simple software dependency is not installed`

To resolve this error, install the following RPMs as the root user:

- `perl-XML-NamespaceSupport-1.11-10.el7.noarch.rpm`
- `perl-XML-SAX-0.99-9.el7.noarch.rpm`
- `perl-XML-SAX-Base-1.08-7.el7.noarch.rpm`
- `perl-XML-Simple-2.20-5.el7.noarch.rpm`

Use the following command to install each RPM:

```
rpm -ivh rpm_name
```

Notices

This information was developed for products and services offered worldwide.

This material may be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service. This document may describe products, services, or features that are not included in the Program or license entitlement that you have purchased.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Software Group
Attention: Licensing
3755 Riverside Dr.
Ottawa, ON
K1V 1B7
Canada

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

IBM Surveillance Insight for Financial Services includes Brat (v 1.3) from the following source and licensed under the following agreement:

- http://weaver.nlplab.org/~brat/releases/brat-v1.3_Crunchy_Frog.tar.gz
- https://creativecommons.org/licenses/by-sa/3.0/legalcode

IBM Surveillance Insight for Financial Services includes spaCy Models (v 1.2.0) from the following source and licensed under the following agreement:

- https://github.com/explosion/spacy-models (en_core_web_sm 1.2.0)
- https://creativecommons.org/licenses/by-sa/3.0/legalcode

Trademarks

IBM, the IBM logo and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at " Copyright and trademark information " at www.ibm.com/legal/copytrade.shtml.

The following terms are trademarks or registered trademarks of other companies:

- Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

- Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.