IBM Surveillance Insight for Financial Services
Version 1.2.0

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 69.

Product Information

This document applies to Version 1.2.0 and may also apply to subsequent releases.

Licensed Materials - Property of IBM

Contents

Introduction

The IBM® Surveillance Insight for Financial Services solution uses cognitive and advanced analytic capabilities for real-time supervision and surveillance, enforcement of regulatory rules, and proactive monitoring of fraudulent activities.

IBM Surveillance Insight for Financial Services uses IBM advanced analytics, including graph computing, big data analytics and reasoning, natural language processing, sentiment analysis, stream analysis, and semantic event analysis. It includes a large-scale real-time graphing tool that can handle and analyze trillions of linked pieces of data. A novel reasoning engine and analytics pipeline to enable it to handle various cognitive tasks. Natural language processing capability enables it to understand text information and to distinguish ambiguous terms based on context and knowledge graphs. It includes sentiment analysis capabilities to understand emotions. Stream analysis provides filters to handle very large amounts of streaming data such as trades. Semantic event analysis applies machine learning to map anomaly signals to human events.

You use this solution to

- Identify new trading and behavioral patterns
- Combine information from multiple data sources, such as trades and electronic communications
- Apply learning models, which are continuously updated
- Provide effective and easy to use visualization tools

The solution provides the following outcomes:

- Identification of new predictive patterns by linking various information with Trader profiles, such as "know your trader"
- Improving the accuracy of alerts by using risk profiling, such as reducing false positives and false negatives
- Improving the efficiency of investigations

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).

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 63.

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.

IBM Surveillance Insight for Financial Services provides trade and electronic communications (eComms) surveillance components. On the trade side, the solution generates alerts from structured trade data (such as order data, trade data, and executions data) by detecting patterns through a high-speed and scalable transaction analysis system that can process large volumes of trade data. Factors that contribute to anomalous behavior can include trading behavior anomalies as compared to peers and an individual's past trading history, communication anomalies such as discussing something suspicious before or after trades, and outside public events from news and social media. However, looking at the trade side alone is not enough to conclude that there is evidence of wrong-doing.

By including eComms surveillance the solution can detect the anomalies and correlate them with the trading activity. Combining trade and eComms surveillance presents a holistic picture. Additionally, the solution provides reason based on data observations that are made over a long time period. As well, the solution learns anomalies from data, by comparing traders' behavior with self, population, and affinity groups. This cognitive reasoning engine also hierarchically correlates the causality of semantic events to predict potential risks.

For the eComms surveillance, the solution can understand natural language and detect various signals from it. For instance, the solution can determine the sentiment and the semantics of eComms communication.

A key piece of the solution is to make it intuitive for compliance teams to easily view the scores of evidence data that is gathered and allow them to explore the data to ensure that there is evidence to pursue next steps. IBM Surveillance Insight for Financial Services provides different visualization techniques that can handle large, multivariate, and heterogeneous graphs and transition network graphs to provide the most easily understandable representation of the underlying evidence data.

With the trade and eComms surveillance components and the cognitive capabilities, IBM Surveillance Insight for Financial Services can meet the supervision and surveillance needs of the financial services industry across front, mid, and back-office functions.

Solution capabilities and features

Taking the surveillance to the next level, IBM Surveillance Insight for Financial Services offers the first of a kind—a cognitive reasoning engine which goes beyond a rule-based alert system and leverages real-time trader trading behaviors along with their communications and social network through email, social media, blogs,

and tweets to provide more accurate and actionable insights and alerts. The solution offers real-time surveillance capability through its reasoning engine with unique visualization experience.

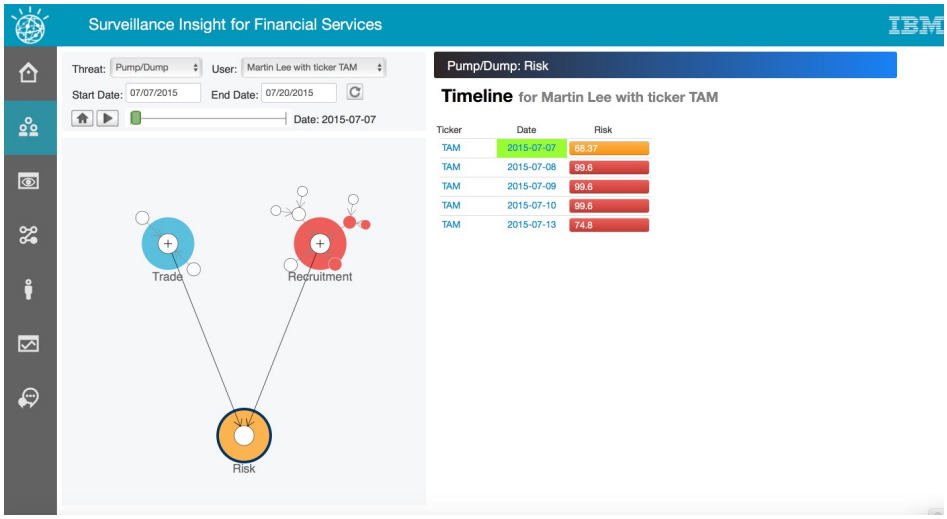
The IBM Surveillance Insight for Financial Services solution provides online, real-time business activity monitoring, with powerful capabilities for helping financial market firms use advanced data mining, statistical models, and sophisticated pattern mining to identify and act on potentially fraudulent activity. In addition, these patterns can be used to generate alerts. The solution takes advantage of IBM's substantial research and development investments to support a real-time surveillance infrastructure that enables firms to:

- Use advanced pattern mining capabilities and statistical models to review market information and identify exceptions within large volumes of data. Leverage cognitive networks that are fundamental form of brains—large-scale Markovian and Bayesian network tools and deep learning tools—to identify surveillance insights.
- Use industry's first ever reasoning engine to combine traditional structured data analysis with unstructured data analysis (such as semantic analysis, behavioral analysis, and emotional analysis) to provide evidence of suspicious transactions.
- Use advanced graph visualization and reasoning that is combined with traditional drill-down and drill-across techniques to help with investigation.

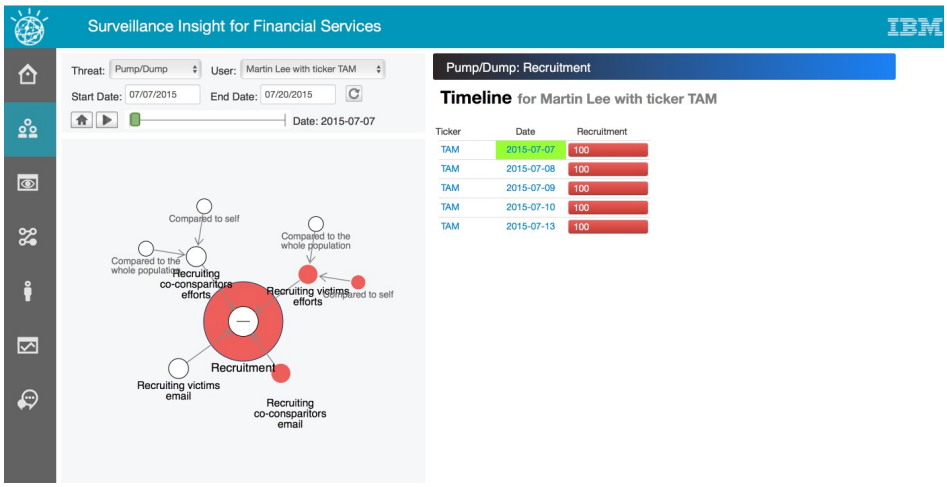
Features such as unstructured data analysis, identity resolution, event correlation, cognitive reasoning, and knowledge management help firms identify scenarios and exceptions for immediate qualification, investigation, and correction. Advanced data mining capabilities help identify suspicious trading behaviors.

The solution interface

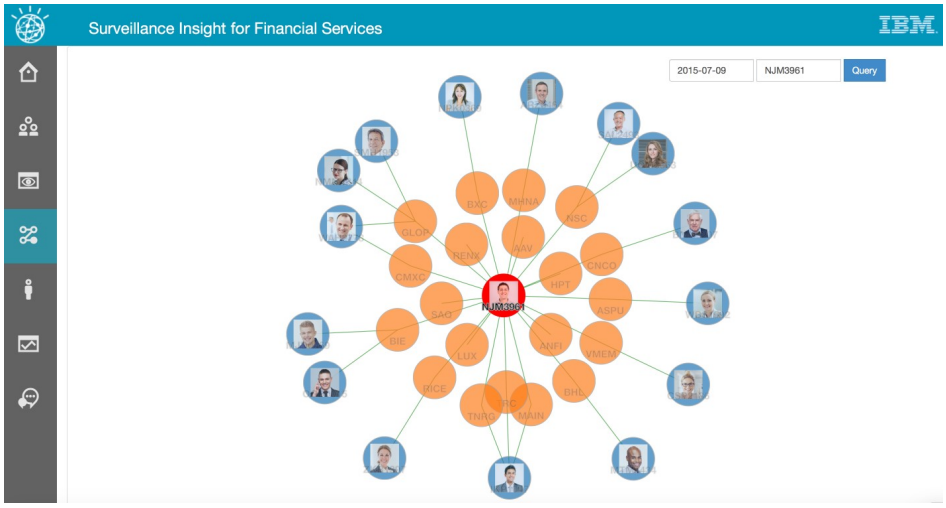
The solution uses integrated analytics to combine trade data with electronic communications to help improve efficiency and accuracy of surveillance results. It applies a reasoning engine to understand traders' long-term behavior, detect anomalies from various 'weak signals' and provide risk estimation to compliance officers.



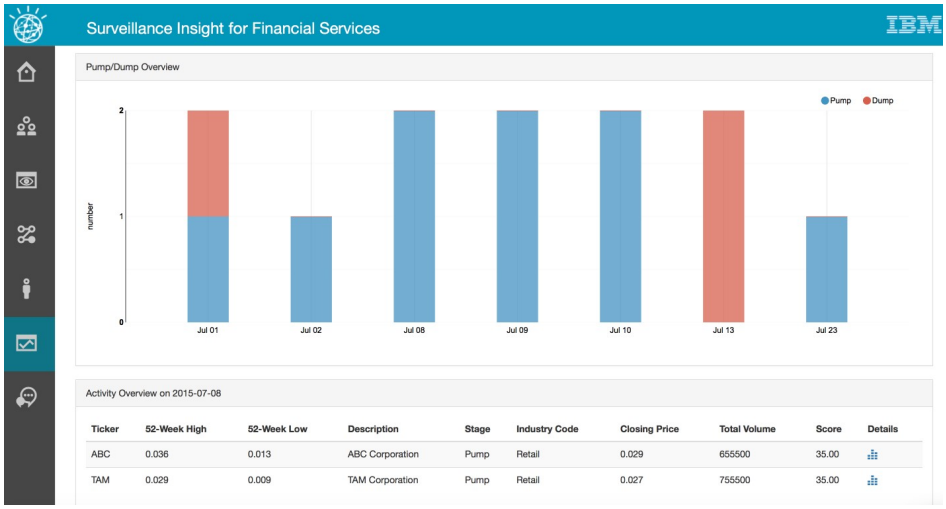
IBM Surveillance Insight for Financial Services learns anomalies from data, by comparing traders' behavior with self, population, and affinity groups. This cognitive reasoning engine also hierarchically correlates the causality of semantic events to predict the potential upcoming risk. Information and alerts are delivered through effective and easy-to-use visualization tools.



The solution uses large-scale graph computing technology to handle and analyze large and linked data sets to gain insights into trading behavior.



Analysis is provided in real time and can scale to handle large volumes of transactions.



Components of the solution

IBM Surveillance Insight for Financial Services offers mid and back office surveillance on market activities and communication in order to detect and report possible market abuse activity. It contains components that include: IBM Base Surveillance Analytics, IBM Electronic Communication Surveillance Analytics, IBM Trade Surveillance Analytics, and IBM Voice Surveillance Analytics.

You must install the IBM Base Surveillance Analytics component before installing other components. To leverage the full capability of the solution it is best to install all components.

IBM Base Surveillance Analytics

This mandatory component offers the Surveillance Platform, which includes IBM Spark, Hadoop, Graph Store, Reasoning Engine, Text analytics framework, Search and Indexing capabilities, and other components that are common for the surveillance framework.

IBM Electronic Communication Surveillance Analytics

This optional component monitors e-communication channels, such as email and instant messaging. It generates alerts based on the policy that you set.

IBM Trade Surveillance Analytics

This optional component monitors the activity of traders to detect market abuse activities. Currently the solution has pre-built models to detect pump-and-dump and spoofing scenarios. In addition, the component offers re-usable operators such as a bulk order detector to detect base alerts.

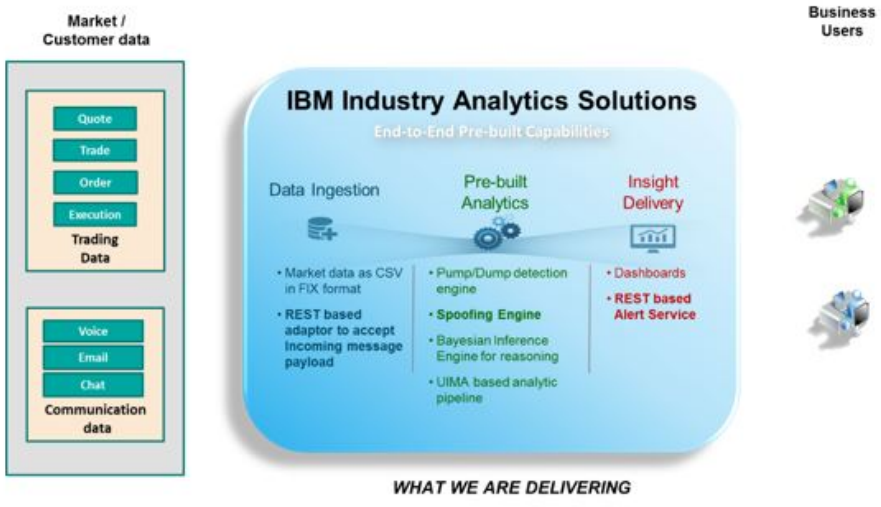
IBM Voice Surveillance Analytics

This optional component monitors the voice communications of traders and employees. It converts the voice to text by using the IBM Watson Speech to Text component and performs analytics on the text to detect and report any possible market abuse patterns. It generates alerts based on the policy that you set.

The IBM Surveillance Insight for Financial Services solution architecture

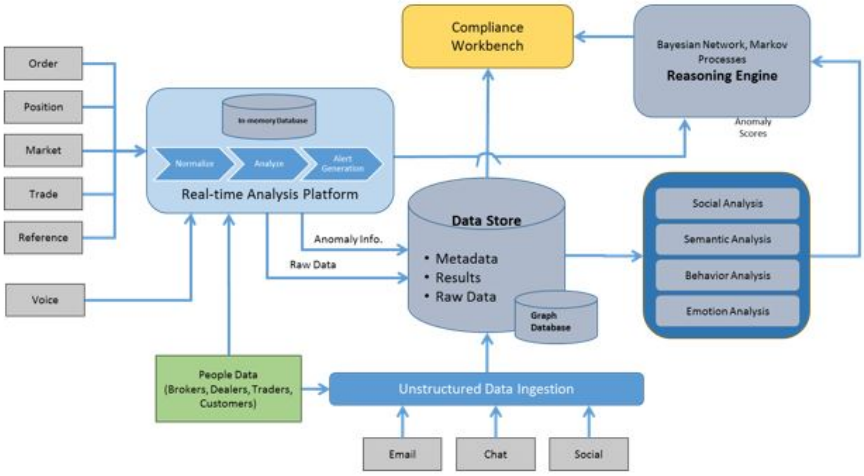
The solution accepts structured market data and unstructured communication data, such as voice, email, and chat messages. It performs real-time and offline analytics on the data and generates relevant alerts for business users. Analytic results are delivered through a web-based user interface.

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



The solution architecture provides the following main components:

- Data Ingestion
- Real-Time Analytics
- Cognitive Analysis and Reasoning
- Data Store
- Compliance Workbench



The solution defines the following data ingestion points:

- For trade data, the solution uses a secure FTP-based interface to ingest trade, order, quote, execution, end of the day, and initial positions data. The schema is based on FIX (http://www.fixtradingcommunity.org/), which makes it easy to import the data.
- For e-communication data, the solution offers secure REST service over DIGEST authentication. You can import your data and create a payload and ingest the data into the solution.
- For voice data, the solution offers secure FTP and also secure Kafka based messaging. You must upload your voice data using a secure FTP method and send the metadata about the voice and its location to the platform so that it can perform analysis and generate alerts.

Your users can be classified depending on their roles in the company. For example, you might set a policy for traders that detects market-related activities and set a different policy for other employees that generates sentiment and emotion-based alerts.

The solution offers secure RESTful services using DIGEST authentication so that you can manage the policies to be applied on the e-communication and voice data.

Real-time analytics platform

The solution uses the IBM InfoSphere Streams platform for all real-time analytics. The platform analyzes incoming data in real time to detect any possible market abuse patterns or patterns that meet the policy criteria for generating alerts.

The solution can integrate with any case management system that is already deployed in your enterprise. Currently, the solution generates a JSON payload for the alert on a given RESTful service. You must provide the RESTful service that is needed to ingest the cases into the customer case management system.

In the trade component, the solution offers the following features:

- Surveillance Base Toolkit
- Pump and Dump use case
- Spoofing use case

The Surveillance base toolkit includes reusable schemas, operators, and an event model that is used by these operators. Events can be intercepted by the use-case modules to detect use-case specific patterns and generate relevant alerts.

The Pump and Dump and Spoofing use cases are detailed in the Trade Surveillance section.

The real-time analytics module persists the insights that are generated from the data and also takes care of persisting raw data.

All raw data for Trade is persisted in Graph using Graph Adaptor services. All the communication data gets persisted to DB2 tables. All the insights are persisted to DB2 tables.

The Cognitive Analysis and Reasoning component

The solution uses an ADAMS – research component for some of the cognitive analysis requirements and the reasoning engine. The solution also provides custom build libraries for detecting emotions, sentiment, concepts, and classifications.

Data store

The solution contains two different data stores. The graph store implementation that is provided by IBM SystemG Graph is used to store raw market data that is uploaded to the system. The relational database SIFS in DB2, is used to store alerts and configuration information.

Note: IBM System G Graph – is a research asset for providing a scalable data store and for maintaining all the relationships, such as person-person, person-trade, and so on.

The solution uses IBM Solr for indexing the e-communication and voice data. The solution does not persist or maintain any copy of the actual communication. It only indexes the extracted insights that can later be leveraged by the compliance workbench to fetch the necessary evidences for the case. It offers secure REST service for data ingestion as well as for queries.

Compliance Workbench

The solution is intended for use by Surveillance officers or Case officers. It offers a prebuilt compliance workbench to provide a unified view of all the alerts on different traders and employees and also helps the officers to navigate through the case to find details on the market activities as well as the communication activities. The compliance workbench is supported by RESTful services from the back end.

The compliance workbench is deployed in Apache HTTP server and it uses the backend REST services that are deployed in IBM WebSphere Application Server and IBM Graph Server.

Chapter 2. Prerequisites and installation preparation

Before you install IBM Surveillance Insight for Financial Services, ensure that the computers that you use meet the minimum requirements for operating systems, prerequisite software, processing, and disk space.

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 a 64-bit Red Hat Enterprise Linux Server Edition 6.8 operating system.

Hardware requirements

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

Processors
2 sockets with 6 cores per socket

RAM 32 GB

Disk space
500 GB

User requirements

You must have **root** or **sudo** access to install IBM Surveillance Insight for Financial Services.

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.

Prerequisite software

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

Install IBM DB2 software on the Data node

You must install IBM DB2® on the data node computer before you can install IBM Surveillance Insight for Financial Services.

After you install the database software, you must create databases that will be used for IBM Surveillance Insight for Financial Services.

For more information about IBM DB2, see the product documentation (www.ibm.com/support/knowledgecenter/SSEPGG_11.1.0).

Installing IBM DB2 Workgroup Server

You must install IBM DB2 Workgroup Server Edition version 11.1.

Note: On Linux operating systems, the setup wizard is a graphical installer that requires an X Windows System (X11) to display the graphical user interface.

Procedure

1. Go to the directory where you decompressed the installation files.
2. Decompress the IBM DB2 installation file (`DB2_AESE_PVU_11.1_Svr_Linux_x86-64.tar.gz`), and go to the `server_r` directory.
3. Start the installer by using the following command: `./db2setup`.
4. In the **DB2 Setup Launchpad**, click **Install a Product**.
5. Under **DB2 Version 11.1 IBM DB2 Advanced Workgroup Server Edition**, click **Install New**.
6. Select **Typical** for the installation type.
7. On the **Select the installation, response file creation, or both** page, accept the default or select **Install DB2 Server Edition on this computer**.
8. Follow the steps in the wizard to install and configure your database server. For more information about any of the settings, see the IBM DB2 documentation on IBM Knowledge Center (www.ibm.com/support/knowledgecenter/SSEPGG_11.1.0/com.ibm.swg.im.dbclient.install.doc/doc/c0023452.html).

Adding the IBM DB2 license

You must add the DB2 license after you install IBM DB2 server for IBM Surveillance Insight for Financial Services.

Procedure

1. Go to the folder where you decompressed the installation files.
2. Decompress the file that is named `DB2_AWSE_AUSI_Activation_11.1.zip`.
3. Change to the database administrator user. For example, `su db2inst1`.
4. Type the following command and press Enter:
   ```
   db2licm -a ese_u/db2/license/db2ese_u.lic
   ```

5. Type the following command to validate that the license was added:
   `db2licm -l`

Securing data at rest for IBM DB2

You must configure data at rest security for IBM DB2 and then create three databases for IBM Surveillance Insight for Financial Services.

Procedure

1. Log in to the database server computer as the root user.
2. Add the IBM Global Security Kit (GSKit) path to the **LD_LIBRARY_PATH** environment variable. For example, enter the following command: `export LD_LIBRARY_PATH=/home/db2inst1/sqllib/gskit/bin:$LD_LIBRARY_PATH`
3. Change to the database instance owner user. For example, enter `su db2inst1`.
4. Go to the `/home/db2inst1/sqllib/gskit/bin` directory.
5. Create a PKCS#12-compliant keystore by entering the following command:
   `./gsk8capicmd_64 -keydb -create -db /home/db2inst1/SIdb2keys.p12 -pw YourPassword -strong -type pkcs12 –stash`
   For more information, see the GSKCapiCmd User's Guide (ftp://public.dhe.ibm.com/software/webserver/appserv/library/v61/ihs/GSK7c_CapiCmd_UserGuide.pdf).
6. Update the DB2 configuration with the new keystore by entering the following command:
   `db2 update dbm cfg using keystore_type pkcs12 keystore_location /home/db2inst1/SIdb2keys.p12`
7. Restart the database instance.
   a. Stop the database. Enter `db2stop`
   b. Start the database. Enter `db2start`
8. Create a 256 bit (32 bytes) master key. You can use the Linux operating system pseudorandom number generator or another utility.
   If you use the pseudorandom number generator, you can use the following command: `dd if=/dev/random of=/home/db2inst1/SIKey bs=1 count=32`
9. Import the generated key into the keystore that you created. You use the `-label` value in the following step.
   `gsk8capicmd_64 -secretkey -add -db /home/db2inst1/SIdb2keys.p12 -file /home/db2inst1/SIKey -label SIlabel.SIdb.SIinstance.SIserver -pw YourPassword`

Creating the databases

After you install the IBM DB2 database server for IBM Surveillance Insight for Financial Services, you must create two databases.

Procedure

1. Change to the database instance owner. For example, `su db2inst1`.
2. Enter the following command to create the `DATAMGT` database:
   `db2 create database DATAMGT pagesize 32 K encrypt cipher aes key length 256 master key label SIlabel.SIdb.SIinstance.SIserver`
3. Enter the following command to create the `RESULTS` database:
   `db2 create database RESULTS pagesize 32 K encrypt cipher aes key length 256 master key label SIlabel.SIdb.SIinstance.SIserver`
4. Enter the following command to create the `SIFS` database:

```
db2 create database SIFS pagesize 32 K encrypt cipher aes key length 256
master key label SIlabel.SIdb.SIinstance.SIserver
```

Securing data in motion for IBM DB2

You must configure data in motion security for IBM DB2 for IBM Surveillance Insight for Financial Services.

Procedure

1. Log in to the database server computer as the database instance owner.
2. Go to the `/home/db2inst1/sqllib/gskit/bin` directory.
3. Add a certificate for your server to your key database.

   The server sends this certificate to clients during the SSL handshake to provide authentication for the server. You can use the IBM Global Security Kit (GSKit) command to create a new certificate request and send it to a certificate authority to be signed. You can use a self-signed certificate for testing or demonstration purposes. For production, you cannot use a self-signed certificate.

   ```
   ./gsk8capicmd_64 -cert -create -db /home/db2inst1/SIdb2keys.p12 -pw
   YourPassword -label "db2-selfsigned" -dn
   "CN=si.ibm.com,O=IBM,OU=IBMAnalytics,L=IN,ST=ON,C=CA"
   ```

   For more information about using this application, see the GSKCapiCmd User's Guide (ftp://public.dhe.ibm.com/software/webserver/appserv/library/v61/ihs/GSK7c_CapiCmd_UserGuide.pdf).
4. Extract the certificate that you created to a file. The file can be distributed to computers that run clients that use SSL communications with the DB2 server.

   For example, you can use `gsk8capicmd` to extract the certificate to a file that is named `SIDB2.arm`:

   ```
   ./gsk8capicmd_64 -cert -extract -db /home/db2inst1/SIdb2keys.p12 -pw
   YourPassword -label "db2-selfsigned" -target "/home/db2inst1/SIDB2.arm"
   -format ascii -fips
   ```
5. Update your IBM DB2 configuration:

   ```
   db2 update dbm cfg using SSL_SVR_KEYDB /home/db2inst1/SIdb2keys.p12
   db2 update dbm cfg using SSL_SVR_STASH /home/db2inst1/SIdb2keys.sth
   db2 update dbm cfg using SSL_SVR_LABEL db2-selfsigned
   db2 update dbm cfg using ssl_svcename 50001
   db2set -i db2inst1 DB2COMM=SSL
   ```
6. Restart the database instance.
   a. Stop the database. Enter `db2stop`
   b. Start the database. Enter `db2start`

Configuring the ODBC data source connections

You must configure ODBC data source connections, including the SSL keystore locations for each database used by IBM Surveillance Insight for Financial Services.

Procedure

1. Add the following to the `/opt/ibm/db2/V11.1/cfg/db2cli.ini` file. If the file does not exist, you must create it.

   ```
   [DATAMGT]
   Hostname=localhost
   Database=DATAMGT
   Protocol=TCPIP
   ServiceName=50001
   Port=50001
   Security=ssl
   ```

```
SSLClientKeystoredb=/home/db2inst1/SIdb2keys.p12
SSLClientKeystash=/home/db2inst1/SIdb2keys.sth

[RESULTS]
Hostname=localhost
Database=RESULTS
Protocol=TCPIP
ServiceName=50001
Port=50001
Security=ssl
SSLClientKeystoredb=/home/db2inst1/SIdb2keys.p12
SSLClientKeystash=/home/db2inst1/SIdb2keys.sth

[SIFS]
Hostname=localhost
Database=SIFS
Protocol=TCPIP
ServiceName=50001
Port=50001
Security=ssl
SSLClientKeystoredb=/home/db2inst1/SIdb2keys.p12
SSLClientKeystash=/home/db2inst1/SIdb2keys.sth
```

Ensure that you set the correct path for the **SSLClientKeystoredb** and **SSLClientKeystash** values.

The values for **ServiceName** and **Port** are the same.

2. If you have an IBM DB2 client on another computer, add the text from step 1 to the `/opt/ibm/db2/client/cfg/db2cli.ini` file.

Installing Apache HTTP server and PHP

You must install Apache HTTP server and PHP. The IBM Surveillance Insight for Financial Services Workbench Content component is installed to the web server.

Important: You must have a subscription to a content repository so that you can install software packages.

Procedure

1. Log on to the computer as the **root** user or as a user with **sudo** permissions.
2. Install Apache HTTP server version 2.4.

   Run the following commands to install all of the required packages.

   ```
   yum -y install httpd24.x86_64
   yum -y install httpd24-apr.x86_64
   yum -y install httpd24-apr-util.x86_64
   yum -y install httpd24-apr-util-openssl.x86_64
   yum -y install httpd24-httpd.x86_64
   yum -y install httpd24-httpd-tools.x86_64
   yum -y install httpd24-libnghttp2.x86_64
   yum -y install httpd24-mod_proxy_html.x86_64
   yum -y install httpd24-mod_session.x86_64
   yum -y install httpd24-mod_ssl.x86_64
   yum -y install httpd24-runtime.x86_64
   ```

   Note: When you install `httpd24.x86_64`, the following packages are also installed as dependencies: `httpd24-apr.x86_64`, `httpd24-apr-util.x86_64`, `httpd24-httpd.x86_64`, `httpd24-httpd-tools.x86_64`, and `httpd24-runtime.x86_64`.

You can verify the version by entering the following command:

```
/opt/rh/httpd24/root/usr/sbin/httpd -v
```

3. Verify that you are using Apache HTTP version 2.4.
4. Install PHP version 5.4.

   Run the following commands to install all of the required packages.

   ```
   yum -y install php54.x86_64
   yum -y install php54-php-cli.x86_64
   yum -y install php54-php-common.x86_64
   yum -y install php54-php-fpm.x86_64
   yum -y install php54-php-pear.noarch
   yum -y install php54-php-process.x86_64
   yum -y install php54-php-xml.x86_64
   yum -y install php54-runtime.x86_64
   ```

   Note: When you install `php54.x86_64`, the following packages are also installed as dependencies: `php54-php-cli.x86_64`, `php54-php-common.x86_64`, `php54-php-pear.noarch`, `php54-php-process.x86_64`, `php54-php-xml.x86_64`, and `php54-runtime.x86_64`.

Managing users for the HTTP server

Use the `htpasswd` utility to create and update the users and passwords for the HTTP server authentication.

Note: In a production environment, it is recommended that you create the `.htpasswd` file in a secure location. Also, ensure that you set the proper file permissions for the `.htpasswd` file.

You must configure the location for the `.htpasswd` file in the `/opt/rh/httpd24/root/etc/httpd/conf.d/surveillance.conf` file.

For more information about `htpasswd`, see the Apache htpasswd - Manage user files for basic authentication page (https://httpd.apache.org/docs/current/programs/htpasswd.html).

Procedure

Generating self-signed certificates

You can use self-signed certificates for test or demonstration environments.

For production environments, you must use a certificate generated by a certificate authority.

Procedure

1. Enter the following command to generate a private key:

   ```
   openssl genrsa –aes256 -out server.key 1024
   ```

   Follow the prompts to complete the key generation.
2. Enter the following command to generate a certificate signing request (CSR):

   ```
   openssl req -new -key server.key -out server.csr
   ```

   Follow the prompts to complete the request.
3. Enter the following commands to remove the passphrase from the key:

   ```
   cp server.key server.key.org
   ```

`openssl rsa -in server.key.org -out server.key`

4. Enter the following command to generate a self-signed key:

   `openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt Signature ok`

   Follow the prompts to complete the key generation.
5. Restart the web server. For example, in a terminal window, enter `service httpd24-httpd restart`.

Installing IBM Installation Manager on the Services node

You must install IBM Installation Manager so that you can install WebSphere Application Server. IBM Installation Manager and WebSphere Application Server must be installed on the services node computer.

Procedure

1. Go to the `InstallationMgr` directory where you decompressed the installation files.
2. Decompress the Installation Manager installation file.
3. Enter the following command to start the installer:

   `./install`
4. Follow the steps to install IBM Installation Manager.

Installing WebSphere Application Server on the Services node

IBM Surveillance Insight for Financial Services services are deployed to a WebSphere® Application Server instance on the services node computer.

You install WebSphere Application Server by using IBM Installation Manager.

For more information about WebSphere Application Server, see the product documentation (www.ibm.com/support/knowledgecenter/SSEQTP_9.0.0).

Procedure

1. Go to the directory where you downloaded the installation files.
2. Decompress the WebSphere Application Server installation file: `WAS_ND_V9.0_MP_ML.zip`.
3. Start IBM Installation Manager from the **Application Browser** or go to the `/opt/IBM/InstallationManager/eclipse` directory, and run `./IBMIM`.
4. Click **File** > **Preferences**.
5. Click **Add Repository**.
6. Browse to the location where you decompressed the WebSphere Application Server installation files.
7. In the directory where you decompressed the installation files, select `repository.config`, click **OK** to close the dialog.
8. In IBM Installation Manager, click **Install**.
9. Under **IBM WebSphere Application Server Network Deployment**, select **Version 9.0.0.0**.
10. Click **Next**, and follow the steps in IBM Installation Manager to install the product.
11. When prompted for **Which program do you want to start?**, select **Profile Management Tool to create a profile**, and click **Finish**.
12. In the **WebSphere Customization Toolbox 9.0.0.0** window, click **Create**.

13. Select **Application Server**, and click **Next**.
14. Select **Typical profile creation**, and click **Next**.
15. Follow the steps in the wizard to configure your application server. For more information about any of the settings, see the WebSphere Application Server documentation.

Applying fix pack 1 to WebSphere Application Server version 9.0.0

You must apply fix pack 1 to your installation of WebSphere Application Server version 9.0 installation. You can apply the fix pack by using IBM Installation Manager.

For more information about the fix pack, see the 9.0.0.1: WebSphere Application Server traditional V9.0 Fix Pack 1 page (www.ibm.com/support/docview.wss?uid=swg24042153).

Procedure

1. Stop WebSphere Application Server.
   a. Go to the `/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin` directory.
   b. Enter the following command: `./stopServer server1`
2. Start IBM Installation Manager from the **Application Browser** or go to the `/opt/IBM/InstallationManager/eclipse` directory, and run `./IBMIM`.
3. Click **Update**.
4. On the **Update Packages** page, select **IBM WebSphere Application Server V9.0**, and click **Next**.
5. Enter your IBM ID and password, and click **OK**.
6. In the list of updates, select **Version 9.0.0.1**, and the recommended IBM SDK update, and click **Next**.
7. Select the 9.0.0.1 fixes from the list of fixes, and click **Next**.
8. Follow the remaining steps to apply the fix pack, click **Update**, and click **Finish**.
9. Start WebSphere Application Server.
   a. Go to the `/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin` directory.
   b. Enter the following command: `./startServer server1`

Installing WebSphere Software Development Kit Java Technology Edition

You install IBM WebSphere SDK Java™ Technology Edition by using IBM Installation Manager. Install WebSphere SDK Java Technology Edition to the same computer where you installed WebSphere Application Server.

After you install the SDK, you must change your WebSphere Application Server configuration to use this version rather than the default.

Procedure

1. Go to the directory where you downloaded the installation files.
2. Decompress the WebSphere SDK installation file: `ibm-java-sdk-8.0-3.0-linux-x64-installmgr.zip`.
3. Start IBM Installation Manager from the **Application Browser** or go to the `/opt/IBM/InstallationManager/eclipse` directory, and run `./IBMIM`.
4. Click **File** > **Preferences**.

5. Click **Add Repository**.
6. Browse to the location where you decompressed the WebSphere SDK installation files.
7. In the directory where you decompressed the installation files, select `repository.config`, click **OK** to close the dialog.
8. In IBM Installation Manager, click **Install**.
9. Under **IBM WebSphere SDK Java Technology Edition (Optional)**, select **Version 8.0**.
10. Click **Next**.
11. On the **Install Packages** page, select **Use the existing package group**, select **IBM WebSphere Application Server V9.0**, and click **Next**.
12. Follow the steps in IBM Installation Manager to install the product.

Configuring data source connections in WebSphere Application Server

You must create data source connections to the DATAMGT, RESULTS, and SIFS databases in your WebSphere Application Server profile. You create the data source connections in the WebSphere administration console.

Procedure

1. Open a web browser.
2. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is `http://`*`servername`*`:9060/ibm/console` where *`servername`* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.
3. If security is enabled, enter the **User ID** and **Password**, and click **Log in**. Otherwise, click **Log in**.
4. Add the database instance owner credentials to a user alias for WebSphere Application Server.
   a. Expand **Servers** > **Server Types** > **WebSphere application servers**, and click **server1**.
   b. Under **Security**, click **Security domain**.
   c. Under **Authentication**, expand **Java Authentication and Authorization Service**, and click **J2C authentication data**.
   d. Click **New**.
   e. Enter a name for the user credentials in **Alias**.
   f. Enter the database instance owner in **User ID**. For example, enter `db2inst1`.
   g. Enter the database instance owner's password in **Password**.
   h. Click **Apply**, and then click **Save**.
5. Expand **Resources** > **JDBC**, and click **Data sources**.
6. Create a data source connection to the DATAMGT database.
   a. On the **Data source** page, click **New**.
   b. Enter `DATAMGT` in **Data source name** and **JNDI name**, and click **Next**.
   c. Select **Create new JDBC provider**, and click **Next**.
   d. Select **DB2** in **Database type**.
   e. Select **DB2 Using IBM JCC Driver** in **Provider type**.
   f. Select **Connection pool data source** in **Implementation type**.

g. If it was not automatically entered, enter `DB2 Using IBM JCC Driver` in **Name**, and click **Next**.
h. On the Enter database class path information, leave the default values, and click **Next**.

   The **Class path** box should contain the following paths:

   ```
   ${DB2_JCC_DRIVER_PATH}/db2jcc4.jar
   ${UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc_license_cu.jar
   ${DB2_JCC_DRIVER_PATH}/db2jcc_license_cisuz.jar
   ${PUREQUERY_PATH}/pdq.jar
   ${PUREQUERY_PATH}/pdqmgmt.jar
   ```

   Leave the remaining boxes blank.
i. Select `4` for the **Driver type**.
j. Enter `DATAMGT` in **Database name**.
k. Enter the name of the server where IBM DB2 is running in **Server name**.
l. Enter the DB2 server port number in **Port number**. For example, `50001`.
m. Click **Next**.
n. Select the alias that you created for the database instance owner in both **Component-managed authentication alias** and **Container-managed authentication alias**, and click **Next**.
o. Click **Finish**, and then click **Save**.
p. Click the data source connection that you created in the table.
q. Under **Additional Properties**, click **Custom properties**.
r. In the properties table, click **`resultSetHoldability`**.
s. Enter `1` in **Value**.
t. Click **Apply**, and then click **Save**.
u. Select the data source connection that you created in the table, and click **Test connection**.

   Note: Ensure that the test is successful.

7. Create a data source connection to the RESULTS database.
   a. On the **Data source** page, click **New**.
   b. Enter `RESULTS` in **Data source name** and **JNDI name**, and click **Next**.
   c. Select **Select an existing JDBC provider**, choose **DB2 Using IBM JCC Driver**, and click **Next**.
   d. Select `4` for the **Driver type**.
   e. Enter `RESULTS` in **Database name**.
   f. Enter the name of the server where IBM DB2 is running in **Server name**.
   g. Enter the DB2 server port number in **Port number**. For example, `50001`.
   h. Click **Next**.
   i. Select the alias that you created for the database instance owner in both **Component-managed authentication alias** and **Container-managed authentication alias**, and click **Next**.
   j. Click **Finish**, and then click **Save**.
   k. Click the `RESULTS` data source.
   l. Under **Additional Properties**, click **Connection pool properties**.
   m. Enter `50` in **Maximum connections**.
   n. Click **Apply**, and then click **Save**.

o. Select the data source connection that you created in the table, and click **Test connection**.

   Note: Ensure that the test is successful.

8. Create a data source connection to the SIFS database.
   a. On the **Data source** page, click **New**.
   b. Enter `RESULTS` in **Data source name** and **JNDI name**, and click **Next**.
   c. Select **Select an existing JDBC provider**, choose **DB2 Using IBM JCC Driver**, and click **Next**.
   d. Select `4` for the **Driver type**.
   e. Enter `SIFS` in **Database name**.
   f. Enter the name of the server where IBM DB2 is running in **Server name**.
   g. Enter the DB2 server port number in **Port number**. For example, `50001`.
   h. Click **Next**.
   i. Select the alias that you created for the database instance owner in both **Component-managed authentication alias** and **Container-managed authentication alias**, and click **Next**.
   j. Click **Finish**, and then click **Save**.
   k. Click the data source connection that you created in the table.
   l. Under **Additional Properties**, click **Custom properties**.
   m. In the properties table, click **`currentSchema`**.
   n. Enter `SIFS` in **Value**.
   o. Click **Apply**, and then click **Save**.
   p. Select the data source connection that you created in the table, and click **Test connection**.

      Note: Ensure that the test is successful.

Securing communication to the data sources in WebSphere Application Server

You must configure WebSphere Application Server to allow secure access to the data sources.

Procedure

1. Open a web browser.
2. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is `http://`*`servername`*`:9060/ibm/console` where *`servername`* is the name or IP address for the computer where you installed WebSphere Application Server.
3. If security is enabled, enter the **User ID** and **Password**, and click **Log in**. Otherwise, click **Log in**.
4. Expand **Security** and then click **SSL Certificates and Key Management**.
5. Under **Related Items**, click **Key stores and certificates**.
6. In the list, click **NodeDefaultTrustStore**.

   In a multi-node environment, click the link for the cell truststore.
7. Under **Additional Properties**, click **Signer certificates**.
8. Click **Retrieve from port**.
9. Enter the information for your IBM DB2 instance:
   a. Enter the server name in **Host**.

b. Enter the DB2 port number in **Port**. For example, enter `50001`.
c. Enter a name for this server in **Alias**. The **Alias** is used only by WebSphere.

10. Click **Retrieve signer information**.
11. Click **OK**, and then click **Save**.
12. Update the datasource connections with the certificates.
    a. Expand **Resources** and then click **JDBC** > **Data sources**.
    b. Click the datasource name. For example, click **DATAMGT**.
    c. Under **Additional Properties**, click **Custom properties**.
    d. In the **Name** box, enter `sslConnection`.
    e. In the **Value** box, enter `true`.
    f. Click **OK**.
    g. Enter the DB2 port number in the **Common and required data source properties** box.
    h. Click **OK**, and then click **Save**.
    i. Repeat these steps for the RESULTS and SIFS databases.
13. Restart WebSphere Application Server.

Securing communication to the datasource for the Analytics Content

Use the following steps to allow standalone programs to connect to a secured DB2 data source over JDBC. For example, The Analytics Content uses this kind of connection.

Procedure

1. Open a web browser.
2. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is `http://`*`servername`*`:9060/ibm/console` where *`servername`* is the name or IP address for the computer where you installed WebSphere Application Server.
3. If security is enabled, enter the **User ID** and **Password**, and click **Log in**. Otherwise, click **Log in**.
4. Expand **Security** and then click **SSL Certificates and Key Management**.
5. Under **Related Items**, click **Key stores and certificates**.
6. In the list, click **NodeDefaultTrustStore**.

   In a multi-node environment, click the link for the cell truststore.
7. Under **Additional Properties**, click **Signer certificates**.
8. Select the IBM DB2 certificate that you added in "Securing communication to the data sources in WebSphere Application Server" on page 19.
9. Click **Extract**, enter a path to save the file, and click **Apply**. For example, in the path, enter `/home/db2inst1/Db2Cert.pem`.
10. Use the following command to import the certificate that you extracted:

    ```
    keytool -importcert -alias db2-selfsigned -file /home/db2inst1/
    Db2Cert.pem -keystore /opt/IBM/WebSphere/AppServer/java/8.0/jre/lib/
    security/cacerts -storepass cacerts_keystore_password
    ```

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/0.10.0.0/kafka_2.11-0.10.0.0.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_2.11-0.10.0.0.tgz -C /opt`
4. Set the **JAVA_HOME** environment variable to be the Java that is provided with WebSphere Application Server.

   ```
   export JAVA_HOME=/opt/IBM/WebSphere/AppServer/java/8.0/
   ```
5. Set the **PATH** environment variable to include the Java that is provided with WebSphere Application Server.

   ```
   export PATH=/opt/IBM/WebSphere/AppServer/java/8.0/bin:$PATH
   ```
6. Optional: If you are using IBM Java, you must edit the `kafka-run-class.sh` file.
   a. Go to the `/opt/kafka_2.11-0.10.0.0/bin` directory.
   b. Open `kafka-run-class.sh` in a text editor.
   c. Locate the **loggc** value in the following line:

      ```
      KAFKA_GC_LOG_OPTS="-Xloggc:$LOG_DIR/$GC_LOG_FILE_NAME -verbose:gc
      -XX:+PrintGCDetails -XX:+PrintGCDateStamps -XX:+PrintGCTimeStamps "
      ```
   d. Change **loggc** to **verbosegclog**:

      ```
      KAFKA_GC_LOG_OPTS="-Xverbosegclog:$LOG_DIR/$GC_LOG_FILE_NAME
      -verbose:gc -XX:+PrintGCDetails -XX:+PrintGCDateStamps
      -XX:+PrintGCTimeStamps "
      ```
   e. Save and close the file.

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 IBM Streams 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=AES
keylength=128
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.
15. On the computer where WebSphere Application Server is installed, create a file that is named `kafka_encryption.properties` in the `/home/SIUser` directory.
16. Enter the following text into the `kafka_encryption.properties` file.

```
com.ibm.si.encryption.algorithm.name=AES
com.ibm.si.encryption.key.length=128
com.ibm.si.encryption.keystore.location=/home/SIUser/SIKafkaEncrypt.jks
com.ibm.si.encryption.keystore.password=YourPassword
com.ibm.si.encryption.certificate.alias=SIKafkaSecurityKey
```

17. Save and close the 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 IBM Streams and Apache Kafka are 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
    ```

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 IBM Streams and Apache Kafka are 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 22 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` directory.
10. Enter the following text into the `producer.properties` file.

    ```
    bootstrap.servers=<IP>:<PORT>
    serializer.class=kafka.serializer.StringEncoder
    request.required.acks=1
    value.serializer=org.apache.kafka.common.serialization.ByteArraySerializer
    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/SIKafkaServerSSLTruststore.jks
    ssl.truststore.password=YourPassword
    ssl.keystore.location=/home/streamsadmin/SIKafkaServerSSLKeystore.jks
    ssl.keystore.password=YourPassword
    ssl.key.password=YourKeyPassword
    ```
11. Save and close the file.
12. Save a copy of the `producer.properties` file to the `/home/streamsadmin/config/properties` directory.

    `producer.properties` should exist in both the `/home/streamsadmin` and `/home/streamsadmin/config/properties` directories.
13. Create a file that is named `consumer.properties` in the `/home/streamsadmin/config/properties` directory.
14. Enter the following text into the `consumer.properties` file.

    ```
    bootstrap.servers=<IP>:<PORT>
    serializer.class=kafka.serializer.StringEncoder
    request.required.acks=1
    value.serializer=org.apache.kafka.common.serialization.ByteArraySerializer
    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=YourPassword
    ssl.keystore.location=/home/streamsadmin/security/SIKafkaServerSSLKeystore.jks
    ssl.keystore.password=YourPassword
    ```
15. Save and close the file.

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

Starting the Apache Kafka services

After you start the Apache Kafka services, you must create topics for the IBM Surveillance Insight for Financial Services data.

Procedure

1. Go to the `/opt/kafka_2.11-0.10.0.0` directory.
2. Enter the following command to start the zookeeper process:

   ```
   bin/zookeeper-server-start.sh config/zookeeper.properties &
   ```
3. Enter the following command to start the Kafka process:

   ```
   bin/kafka-server-start.sh config/server.properties &
   ```
4. Enter the following commands to create topics in Apache Kafka:

   ```
   bin/kafka-topics.sh --create --zookeeper localhost:2181
   --replication-factor 1 --partitions 1 --topic sifs.email.in
   bin/kafka-topics.sh --create --zookeeper localhost:2181
   --replication-factor 1 --partitions 1 --topic sifs.chat.in
   bin/kafka-topics.sh --create --zookeeper localhost:2181
   --replication-factor 1 --partitions 1 --topic sifs.policy.in
   bin/kafka-topics.sh --create --zookeeper localhost:2181
   --replication-factor 1 --partitions 1 --topic sifs.alert.in
   ```
5. List the topics to verify that the topics were created:

   ```
   bin/kafka-topics.sh --list --zookeeper.localhost:2181
   ```

   Four topics should be listed: `sifs.email.in`, `sifs.chat.in`, `sifs.alert.in`, and `sifs.policy.in`.

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.

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

Before you begin

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 `groupadd` command to create the `streamsadmin` group. For example, in a terminal window, enter `groupadd streamsadmin`. The group must exist before you can add the user to the group.
- 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 streamsadim`.
- Use the `passwd` command to set the `streamsadmin` user's password. For example, enter `passwd streamsadmin`, and follow the prompts to set the password.

Procedure

1. Go to the directory where you decompressed the installation files.
2. Decompress the installation files: `Streams-4.2.0.0-x86_64-el6.tar.gz`:

   ```
   tar xvf Streams-4.2.0.0-x86_64-el6.tar.gz
   ```
3. Go to the `StreamsInstallFiles` directory.
4. Start the installer by using the following command: `./InfoSphereStreamsSetup.bin`.
5. In the installer, on the **Select the edition to install** page, select **IBM InfoSphere Streams**, and click **Next**.
6. Ensure that you install all of the missing software packages that are identified on the **Software Dependencies** page.

   For more information about installing the software packages for InfoSphere Streams, see the IBM Streams documentation on IBM Knowledge Center (www.ibm.com/support/knowledgecenter/SSCRJU_4.2.0/com.ibm.streams.install.doc/doc/ibminfospherestreams-install-rpms-streams-package.html).
7. 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**.
8. Click **Next**, and follow the steps in the wizard to install the product.
9. Decompress the speech to text component installation files: `Str-4.2.0.0-Wat-S2T-x86_64-el6.tar.gz`

   ```
   tar xvf Str-4.2.0.0-Wat-S2T-x86_64-el6.tar.gz
   ```

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.

What to do next

When you install the product, you must verify the location and password for the keystore in the `install_ecomm.sh` script. The values that you need to ensure are correct are **`db2ssltruststore`** and **`db2ssltruststorepasswd`**.

Installing Apache Ant libraries on the Analytics node

You must install Apache Ant and Ant Contrib on the analytics node computer.

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.

Installing perl-JSON and PERL modules on the Analytics node

You must install perl-JSON and PERL modules on the analytics node computer.

Procedure

1. Log on to the computer as the **root** user or as a user with **sudo** permissions.
2. Install perl-JSON. For example, in a terminal window, enter `yum install perl-JSON`.
3. Enter the following commands to install the PERL modules:

```
cpan YAML
cpan strict
cpan warnings
cpan File::Basename
cpan LWP::Simple
cpan CGI CGI::Carp
cpan JSON
cpan LWP::UserAgent
cpan URI::Escape
cpan LWP::Protocol::https
```

   If you are prompted for values, you can select the defaults or choose an appropriate option.

Installing Boost C++ libraries on the Analytics node

You must install Boost C++ libraries on the analytics node computer.

Procedure

1. Log on to the computer as the **root** user or as a user with **sudo** permissions.
2. Download the Boost C++ libraries. For example, go to https://sourceforge.net/projects/boost/files/boost/1.57.0/, and download `boost_1_57_0.zip`.
3. Decompress `boost_1_57_0.zip`. For example, go to the directory where you downloaded `boost_1_57_0.zip`, and enter the following command: `unzip boost_1_57_0.zip`.
4. Go to the `boost_1_57_0` directory.
5. Enter the following command to build the installer: `./bootstrap.sh`
6. Enter the following command to run the installer: `./b2 install`

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:

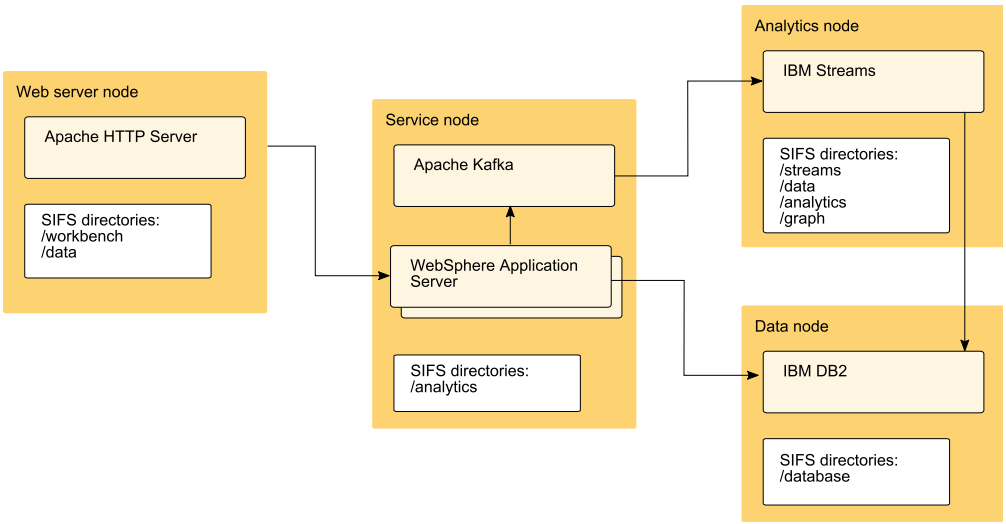
   ```
   username ALL=(ALL) ALL
   ```

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.

Chapter 3. Install IBM Surveillance Insight for Financial Services

Use the solution installer to deploy the IBM Surveillance Insight for Financial Services component programs on computers or virtual machines that are running Red Hat Enterprise Linux operating systems.

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



The solution installer provides a web interface that you use to define to target location to deploy the IBM Surveillance Insight for Financial Services installation files.

After you deploy the installation files, you run a script to install and configure the solution components.

There is a separate installer for each of the four components that comprise IBM Surveillance Insight for Financial Services. The installers are:

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

After you install the IBM Surveillance Insight for Financial Services component, you must install the other components individually.

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, try using 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. When the download is complete, a **Download Complete** message is displayed. The location of the downloaded files are displayed in the message window.
5. Decompress the IBM Surveillance Insight for Financial Services installation files:
   - IBM Surveillance Insight for Financial Services 1.2 Linux x86-64 English
   - IBM Trade Surveillance Analytics 1.2 Linux x86-64 English
   - IBM Electronic Communication Surveillance Analytics 1.2 Linux x86-64 English
   - IBM Voice Surveillance Analytics 1.2 Linux x86-64 English

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 `surv_ins_fin_srvc_1.2_i86-64_en/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 run the IBM Surveillance Insight for Financial Services solution installer on a Red Hat Enterprise Linux operating system. You use the solution installer to define the environment you want to install. After the solution installer is running, you can access the installer interface from another computer by using 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 `surv_ins_fin_srvc_1.2_i86-64_en/SolutionInstaller` directory where you decompressed the solution installer files.
3. Enter the following command: `./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 installation files

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 the Enter key.
   b. Enter the server name in **Host Name**, and press the Enter key.
   c. Enter the user who has access to install the components in **User Name**, and press the Enter key. For example, enter `root` or a user with sudo permissions.
   d. Enter the user's password in **User Password**, and press the Enter key.
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.

   Workbench Content
   Contains the deployment files for the Surveillance workbench component.

   You must deploy this component to a computer where you installed the web server.

   Database Content
   Contains the sample data content.

   You must deploy this content to the same computers where you deploy the Analytics Content and the Streams Content components. If the Analytics Content and Streams Content components are deployed on different computers, you must deploy the Database Content component to both computers.

   Analytics Content
   Contains the deployment files for the Trader Surveillance component.

   Services Content
   Contains the deployment files for the Trade Stream Analytics component.

   Deploy the Services Content and Database Content components together.

   The files content files are copied to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2` 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 four components that you must install for IBM Surveillance Insight for Financial Services.

Before you can install the remaining IBM Surveillance Insight for Financial Services components, you must uninstall the solution installer and then install the solution installer for the next component. You must repeat this task for each component installer.

When you uninstall the solution installer, the processes that are used by the solution installer are stopped and removed from your computer.

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.
3. Install the IBM Electronic Communication Surveillance Analytics component:
   a. Go to the `surv_ins_fin_srvc_ecomm_1.2_i86-64_en/SolutionInstaller` directory where you decompressed the solution installer files.
   b. Enter the following command: `./setup.sh` *`username first_name last_name email password`*.
   c. After the solution installer starts, open a web browser, and go to the solution installer URL: `https://`*`servername`*`:8080/UI/index.html`.
   d. Use the solution installer to deploy the Services Content component.
      The files content files are copied to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_EComm_1.2` directory. You can change that value in the **Property Editor**.
   e. Run the `cleanupClient.sh` script on the client node computers.
   f. Run the `cleanup.sh` script on the installer computes.

g. Restart the installation computer.

4. Install the IBM Trade Surveillance Analytics component:
   a. Go to the `surv_ins_fin_srvc_trade_1.2_i86-64_en/SolutionInstaller` directory where you decompressed the solution installer files.
   b. Enter the following command: `./setup.sh` *`username first_name last_name email password`*.
   c. After the solution installer starts, open a web browser, and go to the solution installer URL: `https://`*`servername`*`:8080/UI/index.html`.
   d. Use the solution installer to deploy the components.

      The files content files are copied to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Trade_1.2` directory. You can change that value in the **Property Editor**.
   e. Run the `cleanupClient.sh` script on the client node computers.
   f. Run the `cleanup.sh` script on the installer computes.
5. Install the IBM Voice Surveillance Analytics component:
   a. Go to the `surv_ins_fin_srvc_voice_1.2_i86-64_en/SolutionInstaller` directory where you decompressed the solution installer files.
   b. Enter the following command: `./setup.sh` *`username first_name last_name email password`*.
   c. After the solution installer starts, open a web browser, and go to the solution installer URL: `https://`*`servername`*`:8080/UI/index.html`.
   d. Use the solution installer to deploy the components.

      The files content files are copied to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Voice_1.2` directory. You can change that value in the **Property Editor**.

Set up the web server node

Setting up the Workbench Content

The Workbench Content component provides the interface for IBM Surveillance Insight for Financial Services.

Procedure

1. Log onto the computer where you deployed the Workbench Content.

   This computer must have a web server installed. For more information, see "Installing Apache HTTP server and PHP" on page 13.
2. Go to the `IS_FinancialMkts_SurveillanceInsight_1.2/Workbench/FinancialMkts_SurveillanceInsight_WorkbenchContent` directory. By default, `IS_FinancialMkts_SurveillanceInsight_1.2` is in `/opt/IBM`.

   If you did not automatically decompress the content files, decompress `FinancialMkts_SurveillanceInsight_WorkbenchContent.zip` in the `IS_FinancialMkts_SurveillanceInsight_1.2/Workbench` directory.
3. From the `FinancialMkts_SurveillanceInsight_WorkbenchContent` directory, copy the `surveillance` and `surveillance_auth` directories to the `/opt/rh/httpd24/root/var/www/html` directory.
4. From the `FinancialMkts_SurveillanceInsight_WorkbenchContent` directory, copy the `surveillance.conf` file to the `/opt/rh/httpd24/root/etc/httpd/conf.d` directory.
5. Open the `surveillance.conf` file in a text editor.

a. Change the file paths and URL values so that they match your environment.
b. Change the `AuthUserFile` value to point to the location where you store the user information.

   For more information, see "Managing users for the HTTP server" on page 14.

   `AuthUserFile` appears in two locations in the file.
c. Change the `SSLCertificateFile` value to point to the location where the SSL certificate is located.

   For more information, see "Generating self-signed certificates" on page 14.
d. Change the `SSLCertificateKeyFile` value to point to the locations where the SSL keystore is located.
e. Save and close the file.

6. Go to the `/opt/rh/httpd24/root/etc/httpd/conf.d` directory, and rename the `ssl.conf` file extension.

   Both the `ssl.conf` and `surveillance.conf` files contain security information. To ensure that only the `surveillance.conf` is read by the Apache HTTP server, rename the `ssl.conf` extension or move the file to another location.
7. Go to the `/opt/rh/httpd24/root/etc/httpd/conf` directory, and open the `httpd.conf` file in a text editor.
   a. Comment out the line that says `Listen 80`.
   b. Save and close the file.
8. Restart the web server. For example, in a terminal window, enter `service httpd24-httpd restart`.
9. Restart the PHP server. For example, in a terminal window, enter `service php54-php-fpm restart`.
10. To access the interface, open a web browser, and enter the following URL: `https://`*`servername`*`:443/surveillance`.

Set up the data node

Creating the database tables and loading metadata

You create the database tables and load the metadata that is used by IBM Surveillance Insight for Financial Services by running scripts on the data node computer.

Procedure

1. Log on to the database node computer.
2. Go to the `IS_FinancialMkts_SurveillanceInsight_1.2/Database/FinancialMkts_SurveillanceInsight_DatabaseContent` directory.

   By default, `IS_FinancialMkts_SurveillanceInsight_1.2` is in `/opt/IBM`.
3. Copy the `Scripts` folder to the database instance owner's home directory. For example, copy `Scripts` to the `/home/db2inst1` directory.
4. Change the ownership of the folders that you copied to the database instance owner user. For example, **`db2inst1`**.
5. Change to the database administrator user. For example, `su db2inst1`.
6. Connect to the SIFS database:

   ```
   db2 connect to SIFS
   ```
7. Run the following command to create the databases and tables:

```
db2 -tvf /home/db2inst1/Scripts/Script.sql
```

8. Run the following command to create the pre-defined KPIs:

```
db2 -tvf /home/db2inst1/Scripts/MetaData.sql
```

Set up the analytics node

Configuring IBM InfoSphere Streams

You must create and start a domain for IBM Streams.

For more information about configuring IBM Streams, see IBM Knowledge Center (www.ibm.com/support/knowledgecenter/SSCRJU_4.2.0/com.ibm.streams.cfg.doc/doc/creating-basic-domain-and-instance.html).

Procedure

1. Go to the `IS_FinancialMkts_SurveillanceInsight_1.2/Analytics/FinancialMkts_SurveillanceInsight_AnalyticsContent` directory. By default, `IS_FinancialMkts_SurveillanceInsight_1.2` is in `/opt/IBM`.

   If you did not automatically decompress the content files, decompress `FinancialMkts_SurveillanceInsight_AnalyticsContent.zip` in the `IS_FinancialMkts_SurveillanceInsight_1.2/Analytics` directory.
2. Copy the following directories to the `/home/streamsadmin` directory:
   - `bin`
   - `components`
   - `config`
   - `security`
3. Change the ownership of the folders that you copied to the streamsadmin user.
4. Change to the IBM Streams user. For example, `su streamsadmin`.
5. Go to the `/home/streamsadmin` directory.
6. Create a `.bashrc` file, and add the following lines:

```
source /opt/ibm/InfoSphere_Streams/4.2.0.0/bin/streamsprofile.sh
export STREAMS_ADAPTOR=/home/streamsadmin/components/toolkits/graphdb_adaptor
export SIFS_HOME=/home/streamsadmin
```

7. Save and close the file.

Setting up and initializing the Graph Content

Use the Graph content to create the graph stores and run the analytics for structured data. The Graph content is deployed with the Analytics Content component.

Before you begin

Ensure that you installed the Boost C++ libraries on the computer where you deployed the Analytics Content. For more information, see "Installing Boost C++ libraries on the Analytics node" on page 27.

Procedure

1. Log on to the analytics node as the streamsadmin user.
2. Go to the `IS_FinancialMkts_SurveillanceInsight_1.2/Analytics/FinancialMkts_SurveillanceInsight_GraphContent` directory. By default, `IS_FinancialMkts_SurveillanceInsight_1.2` is in `/opt/IBM`.

If you did not automatically decompress the content files, decompress FinancialMkts_SurveillanceInsight_AnalyticsContent.zip in the IS_FinancialMkts_SurveillanceInsight_1.2/Analytics directory.

3. Copy the contents of the FinancialMkts_SurveillanceInsight_GraphContent directory to the /home/streamsadmin/Graph directory. You must create the Graph directory.
4. Copy the contents of the FinancialMkts_SurveillanceInsight_DataContent directory to the /home/streamsadmin/Data directory. You must create the Data directory.
5. Start the graph services.
   a. Make a directory in which to store the graph data, and enter that directory path in the start_restapi.sh file. Change the <graphstore_dir> value to the directory that you created.

      The directory must exist in the same directory as the start_restapi.sh file. For example, change ./gShell http localhost 8002 gshell *<graphstore_dir>*/database/ & to something like ./gShell http localhost 8002 gshell ./*database* &
   b. Start the graph services by entering the following command: ./start_restapi.sh.
   c. Press Enter to return to the command line.
6. Enter the following commands to initialize the graphs:
   a. Enter ./create_graphs.sh *<path_to_data_content>*

      <path_to_data_content> is the path to the IS_FinancialMkts_SurveillanceInsight_1.2/Data/FinancialMkts_SurveillanceInsight_DataContent directory. For example, run the command as ./create_graphs.sh /opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/Data/FinancialMkts_SurveillanceInsight_DataContent
   b. Enter ./create_pd_graphs.sh
   c. Enter ./update_graphs.sh

Running the graph adaptor stream jobs

Procedure

1. Log on to the analytics node as the streamsadmin user.
2. Go to the /home/streamsadmin/components/toolkits directory.
3. Copy the libgShell_LMDB.so file to the /usr/lib64 directory.
4. Go to the /home/streamsadmin/graphdb_adaptor directory. If the graphdb_adaptor directory does not exist, create it.
5. Create a soft link to the directory that contains the start_restapi.sh file that you used in "Setting up and initializing the Graph Content" on page 36.

   For example:

   ln -s /home/streamsadmin/Graph/database .
6. Open run-graphstream.sh in a text editor.
   a. Change the **STINSTANCE** value to the name of the IBM Streams instance that you created. For example, change the **STINSTANCE** value to SIInstance.
   b. Change the **STDOMAIN** value to the name of the IBM Streams domain that you created. For example, change the **STDOMAIN** value to StreamsDomain.
   c. Save and close the file.
7. Go to the graphdb_adaptor/impl/java/lib directory.

8. Open `graph.properties` in a text editor.
   a. Change the **`graphurl`** value to the match your web server configuration. For example, if you used the default web server port, 80, change the URL to `http://localhost:3000/cgi-bin/surveillance/restapi/rest/graphs/`.
   b. Save and close the file.
9. Go to the `/home/streamsadmin/components/toolkits/graphdb_adaptor` directory.
10. Enter the following command:
    ```
    sh build.sh
    ```
11. Verify that the IBM Streams and graph integration is correct:
    a. Go to the `/home/streamsadmin/components/toolkits/graphdb_adaptor` directory.
    b. Enter the following command:
       ```
       ./run-graphstream.sh verifygraph /opt/IBM/
       IS_FinancialMkts_SurveillanceInsight_1.2/Analytics/
       FinancialMkts_SurveillanceInsight_Data Content/PumpDumpSolution/
       PDZDataSet/testgraph.csv
       ```
    c. Enter the following command to verify that the content was loaded:
       ```
       curl -X GET -H "X-SystemG-User: sguser_data" http://localhost:8002/
       gshell/graphs/testgraph/vertices?_cmd=count
       ```
       The command should return the following results:
       ```
       {"numberof vertices":28}.
       ```
    d. Enter the following command to verify that the content was loaded:
       ```
       curl -X GET -H "X-SystemG-User: sguser_analytics"
       http://localhost:8002/gshell/graphs/pumpdump/vertices?_cmd=count
       ```
       The command should return the following results:
       ```
       {"numberof vertices":2200}.
       ```
12. Go to the `/home/streamsadmin/components/toolkits/graphdb_adaptor` directory.
13. Run the following commands separately:
    a. `sh run-graphstream.sh BaseGraphs`
    b. `sh run-graphstream.sh EODProcess`
    c. `sh run-graphstream.sh PumpDumpScore`
    d. `sh run-graphstream.sh Trader_Positions`
    e. `sh run-graphstream.sh TradeSummaryAndMA`
    f. `sh run-graphstream.sh Ticker_Detail`
    g. `sh run-graphstream.sh OrderVertex`

Configuring the graph tools content

You must run a script to configure the graph tools content on the analytics node computer.

Procedure

1. On the analytics node computer, change to the IBM Streams user. For example, `su streamsadmin`.
2. Go to the `/home/streamsadmin/components/toolkits/GraphTools` directory.
3. Run the following command:
   ```
   sh build-graphstream.sh
   ```

Configuring the Streams Content

To configure the Streams Content, you must create some directories, run some InfoSphere® Streams jobs, and then install a graph adapter.

Before you begin

Ensure that you ran the database scripts. For more information, see "Creating the database tables and loading metadata" on page 35.

Ensure that you configured the ODBC datasource connections. For more information, see "Configuring the ODBC data source connections" on page 12.

Ensure that you configured security for Apache Kafka. For more information, see "Installing Apache Kafka on the Services node" on page 20.

Procedure

1. Change to the InfoSphere Streams user. For example, change to the streamsadmin user.
2. Go to the `/home/streamsadmin/config/db` directory.
3. Open `connections.xml` in a text editor.
4. Edit the **user** and **password** values in the following line:

   ```
   <ODBC database="SIFS" user="db2inst1" password="db2inst1" /">
   ```
5. Save and close the file.

Installing Apache Solr on the Analytics node

Apache Solr is used to index unstructured data, such as emails, chats, and voice transcripts. You must install Apache Solr on the Analytics node.

For more information about Apache Solr, see the product documentation (lucene.apache.org/solr/6_2_1/index.html).

Procedure

1. Go to the `IS_FinancialMkts_SurveillanceInsight_1.2/Analytics/FinancialMkts_SurveillanceInsight_AnalyticsContent/Installables` directory. By default, `IS_FinancialMkts_SurveillanceInsight_1.2` is in `/opt/IBM`.

   If you did not automatically decompress the content files, decompress `FinancialMkts_SurveillanceInsight_AnalyticsContent.zip` in the `IS_FinancialMkts_SurveillanceInsight_1.2/Analytics` directory.
2. In the `IBM Solr` directory, decompress `solr-6.2.1.tgz` to the `/opt` directory. For example, enter `tar xvf solr-6.2.1.tgz -C /opt`
3. Start Solr:
   a. Go to the `/opt/solr-6.2.1/bin` directory.
   b. Enter the following command: `./solr -p 8984`
4. Create a Solr repository that is named `sifs`:

   ```
   ./solr create -c sifs
   ```
5. In the `/opt/solr-6.2.1/server/solr/sifs/conf` directory, replace `managed-schema` with the schema file in `/home/streamsadmin/config/IndexSchema`.
6. Restart Solr:
   a. Go to the `/opt/solr-6.2.1/bin` directory.

b. Enter the following command: `./solr stop -all`
c. Enter the following command: `./solr -p 8984`

7. Verify that Apache Solr is running. In a web browser, go to `https://servername:8984/solr/#/sifs`
8. Change to the IBM Streams user. For example, `su streamsadmin`.
9. Build and submit the indexing job.
   a. Go to the `/home/streamsadmin/components/Solr/SolrIndexing` directory.
   b. Run the following command:
      ```
      sh build.sh
      ```
   c. Open `submitjob.sh` in a text editor.
   d. Ensure that **INDEXURL** value matches the Solr server URL.
   e. Run the following command:
      ```
      sh submitjob.sh
      ```
10. Configure the policy execution streams content.
    a. Go to the `/home/streamsadmin/config/db` directory.
    b. Open `connections.xml` in a text editor.
    c. Ensure that the database user name and password are correct for your environment.
    d. Open `build.sh` in a text editor.
    e. Ensure that the IBM Streams installation location, IBM Streams home directory, truststore password, and truststore location of the `jks` file are correct for your environment.
    f. Open `submitjob.sh` in a text editor.
    g. Ensure that the database host name, user name, and password are correct for your environment.
    h. Run the following command:
       ```
       sh build.sh
       ```
    i. Run the following command:
       ```
       sh submitjob.sh
       ```

Enabling SSL for Apache Solr

You must create a self-signed key to enable SSL for Apache Solr.

Procedure

1. On the computer where you installed Apache Solr, go to the *SolrInstallation*`/server/etc`.
2. Enter the following command to create a keystore.
   ```
   keytool -genkeypair -alias solr-ssl -keyalg RSA -keysize 2048 -keypass
   YourPassword -storepass YourPassword -validity 365 -keystore
   solr-ssl.keystore.jks -ext SAN=DNS:localhost,IP:XX.XX.XX.XX -dname
   "CN=XX.XX.XX.XX, OU=IBM, O=IBM, C=IN"
   ```
3. Go to the *SolrInstallation*`/bin` directory.
4. Open `solr.in.sh` in a text editor.
5. Uncomment and edit the following lines so that they match your environment.

```
SOLR_SSL_KEY_STORE=etc/solr-ssl.keystore.jks
SOLR_SSL_KEY_STORE_PASSWORD=YourPassword
SOLR_SSL_TRUST_STORE=etc/solr-ssl.keystore.jks
SOLR_SSL_TRUST_STORE_PASSWORD=YourPassword
SOLR_SSL_NEED_CLIENT_AUTH=false
SOLR_SSL_WANT_CLIENT_AUTH=false
```

6. Save and close the file.
7. Restart Solr:
   a. Go to the `/opt/solr-6.2.1/bin` directory.
   b. Enter the following command: `./solr stop -all`
   c. Enter the following command: `./solr -p 8984`

Set up the services node

Setting up and initializing ADAMs

The IBM Surveillance Insight for Financial Services ADAMs component installation downloads and installs additional software components, including Apache Maven.

When you set up the IBM Surveillance Insight for Financial Services ADAMs component, WAR files are generated that you must deploy to WebSphere Application Server.

Before you begin

Ensure that you installed IBM DB2, IBM WebSphere Application Server, and IBM WebSphere SDK Java Technology Edition on the computer where you deployed the Services Content. For more information, see "Install IBM DB2 software on the Data node" on page 10 and "Installing WebSphere Application Server on the Services node" on page 15.

Procedure

1. Log on to the computer where you deployed the IBM Surveillance Insight for Financial Services Services Content as the root user or as a user with sudo permissions.
2. Update your Java environment.
   a. Update your **JAVA_HOME** variable to use the Java that you installed for WebSphere Application Server. For example, enter the following command:

      ```
      export JAVA_HOME=/opt/IBM/WebSphere/AppServer/java/8.0
      ```

   b. Add the Java bin directory to your **PATH** environment variable. For example, enter the following command:

      ```
      export PATH=/opt/IBM/WebSphere/AppServer/java/8.0/bin:$PATH
      ```

3. In the `home` directory, create a directory that is named `IBM_Adams`. For example, create a `/home/IBM_Adams` directory.
4. Go to the `IS_FinancialMkts_SurveillanceInsight_1.2/Services/FinancialMkts_SurveillanceInsight_ServicesContent` directory. By default, `IS_FinancialMkts_SurveillanceInsight_1.2` is in `/opt/IBM`.

   If you did not automatically decompress the content files, decompress `FinancialMkts_SurveillanceInsight_ServicesContent.zip` in the `IS_FinancialMkts_SurveillanceInsight_1.2/Services` directory.
5. Copy the following files to the `/home/IBM_Adams` directory:
   - `eaves-ship-pom-1.2-SNAPSHOT-source.zip`

- install.py
- settings.xml

6. In the /home/IBM_Adams directory, decompress eaves-ship-pom-1.2-SNAPSHOT-source.zip.
7. From the /opt/ibm/db2/V11.1/java directory, copy the db2jcc4.jar file to the /home/IBM_Adams directory. db2jcc4.jar must exist in the same location as install.py.
8. In the /home/IBM_Adams directory, enter the following command in a terminal window:

   ./install.py <*target_directory*> <*download_directory*>

   The <*target_directory*> is where the trade surveillance components and software such as Apache Maven is installed. If you do not provide a path, /home/IBM_Adams is used.

   The <*download_directory*> is where the downloaded software, such as the Apache Maven installer, is saved. If you do not provide a path, /home/IBM_Adams/Downloads is used.

   Tip: If you encounter any issues during the installation, you can correct the issue and then rerun the install.py command. Any tasks that are complete are skipped.
9. When prompted, enter the appropriate values for your environment or accept the defaults.
   - For the JDBC URL of the database to be used value, the connection information for the IBM DB2 database server and the DATAMGT database that you created. Enter the URL in the following format: "jdbc:db2://*database_node*:50001/DATAMGT:sslConnection=true;".

     Important: The database name is case-sensitive.

     Ensure that you use the correct port number for the IBM DB2 database instance.
   - For the User ID to be used to access the above database value, enter the database user name. The default is test. Ensure that you enter the user name that has privileges for the database.
   - For the Password for the above database user ID value, enter a password for the database user. The default is testpw.
   - For the Enter Y to use the same DB server and access credentials for the RESULTS database value, you can choose to create a RESULTS database in the same database server as the DATAMGT database, or you can choose another database server.

     Important: If you enter the database name, ensure that you enter the correct case for the database name.
   - For the URL of the email text server value, enter the URL for the email text server. The default is http://localhost:3000/emailtext/. The localhost value should point to the server and port number where Apache web server is running.

     For the sample data, you must enter the trailing slash as shown. The emailtext directory is included in the Data Content. For more information, see steps 18 - 19 in "Setting up the sample unstructured data" on page 56.
   - For the the PATH of the WebSphere Install Folder value, enter a path to where the WAR files that you must deploy to WebSphere Application Server are created. The default is /opt/IBM/WebSphere/AppServer/runtimes/.

- For the `server on which the graph store resides` value, enter the URL for the computer where you install the Streams Content component. The default is `localhost`. This value is the server where you install and run the IBM System G component as part of the Streams Content.

Enabling security in WebSphere Application Server to deploy the REST applications

You must enable security in WebSphere Application Server before you deploy the IBM Surveillance Insight for Financial Services REST applications.

Procedure

1. On the services node computer, go to the `IS_FinancialMkts_SurveillanceInsight_1.2/Services/FinancialMkts_SurveillanceInsight_ServicesContent/lib` directory. By default, `IS_FinancialMkts_SurveillanceInsight_1.2` is in `/opt/IBM`.

   If you did not automatically decompress the content files, decompress `FinancialMkts_SurveillanceInsight_ServicesContent.zip` in the `IS_FinancialMkts_SurveillanceInsight_1.2/Services` directory.
2. Copy `com.ibm.sifs.security.jar` to the `/opt/IBM/WebSphere/AppServer/lib/ext` directory.
3. Open a web browser.
4. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is `http://`*`servername`*`:9060/ibm/console` where *`servername`* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.
5. If security is enabled, enter the **User ID** and **Password**, and click **Log in**. Otherwise, click **Log in**.
6. Expand **Security**, and click **Global security**.
7. Select **Enable application security**.
8. Select **Federated repositories** in **Available realm definitions**.
9. Under **Web and SIP security**, click **Trust association**.
10. Under **Additional Properties**, click **Interceptors**.
11. Click **New**.
12. In **Interceptor class name**, enter `com.ibm.sifs.security.tai.CustomDigestTAI`.
13. Click **Apply**, and then click **Save**.
14. On the **Trust association** page, **Enable application security**.
15. Click **Apply**, and then click **Save**.

Creating shared libraries for Apache Kafka for the REST applications

You must create a shared library in WebSphere Application Server to access the `jar` files that are needed by Apache Kafka when you run the REST application.

Procedure

1. On the services node computer, go to the `IS_FinancialMkts_SurveillanceInsight_1.2/Services/FinancialMkts_SurveillanceInsight_ServicesContent/lib` directory. By default, `IS_FinancialMkts_SurveillanceInsight_1.2` is in `/opt/IBM`.

2. Copy com.ibm.sifs.security.kafka-<latest_version>.jar to the /home/streamsadmin/ directory.
3. Open a web browser.
4. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is http://*servername*:9060/ibm/console where *servername* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.
5. Expand **Environment**, and click **Shared libraries**.
6. Click **New**.
7. In **Name**, enter KAFKALIB.
8. In **Classpath**, enter the following text:

   ```
   <kafka_installed_location>/libs/kafka_<available_version>.jar
   <kafka_installed_location>/libs/log4j-<available_version>.jar
   <kafka_installed_location>/libs/slf4j-api-<available_version>.jar
   <kafka_installed_location>/libs/metrics-core-<available_version>.jar
   <kafka_installed_location>/libs/scala-library-<available_version>.jar
   <kafka_installed_location>/libs/kafka-clients-<available_version>.jar
   <kafka_installed_location>/libs/kafka-tools-<available_version>.jar
   /home/streamsadmin/com.ibm.sifs.security.kafka-<latest_version>.jar
   ```
9. Click **Apply**, and then click **Save**.

Deploy the REST application for Actiance integration

You must deploy the com.ibm.sifs.service.data.war file in WebSphere Application Server to use the REST services for Actiance integration.

Ensure that you enable security in WebSphere Application Server before you deploy the REST application. For more information, see "Enabling security in WebSphere Application Server to deploy the REST applications" on page 43.

Ensure that you have created the Apache Kafka shared libraries. For more information, see "Creating shared libraries for Apache Kafka for the REST applications" on page 43.

Identifying users who access the email and chat services

You must identify the users who will access the email and chat services from the REST application that is used for the Actiance integration.

You must add the users in the WebSphere Administrative Console and then add the users and their encoded passwords to a .json file.

Procedure

Adding JNDI variables for the REST application for Actiance integration

You must add JNDI variables WebSphere Application Server so that you can run the REST application for Actiance integration.

Procedure

1. Open a web browser.
2. In the address bar, type the address for the WebSphere Admin Console.

The Admin console address is `http://servername:9060/ibm/console` where *servername* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.

3. Expand **Environment** > **Naming** > **Name space bindings**, and then click **New**.
4. Add a binding type:
   a. For the **Binding type**, select **String**, and click **Next**.
   b. In **Binding identifier**, enter `ACTIANCE_EMAIL_FILENAME`.
   c. In **Name**, enter `ACTIANCE_EMAIL_FILENAME`.
   d. In **String value**, enter `snapshot_email`.
   e. Click **Finish**.
5. Repeat step 4 to add the following binding types:

Binding identifier	Name	String value
ACTIANCE_REST_URL	ACTIANCE_REST_URL	http://actiance.com:8089/v1/snapshot
ACTIANCE_REST_USER	ACTIANCE_REST_USER	Ibm1
ACTIANCE_REST_PASSWORD	ACTIANCE_REST_PASSWORD	ibmpwd1
ACTIANCE_USERS	ACTIANCE_USERS	/home/*actiance_user*/files/users.json
SERVER_TRUSTSTORE	SERVER_TRUSTSTORE	The default path is <WASROOT>/profiles/<ProfileName>/config/cells/<cellname>/nodes/<node name>/trust.p12
SERVER_TRUSTSTORE_PWD	SERVER_TRUSTSTORE_PWD	WebAS
ACTIANCE_EMAIL_TOPIC	ACTIANCE_EMAIL_TOPIC	sifs.email.in
ACTIANCE_CHAT_TOPIC	ACTIANCE_CHAT_TOPIC	sifs.chat.in
KAFKA_PROP_LOC	KAFKA_PROP_LOC	/home/*SIUser*/producer.properties
POLICY_KAFKA_TOPIC	POLICY_KAFKA_TOPIC	sifs.policy.in
KAFKA_ENCRYPTION_PROP_LOC	KAFKA_ENCRYPTION_PROP_LOC	/home/*SIUser*/kafka_encryption.properties
TRADE_URL	TRADE_URL	https://localhost/surveillance/know_your_employee
COMMUNICATION_URL	COMMUNICATION_URL	https://localhost:9443/surveillance/data/commevidence/

Configuring logging for the REST application

The default log level in WebSphere Application Server is INFO. You must change the logging level for IBM Surveillance Insight for Financial Services, if you want to check logs other than INFO.

Procedure

1. Open a web browser.
2. In the address bar, type the address for the WebSphere Admin Console.

The Admin console address is `http://servername:9060/ibm/console` where *servername* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.

3. Expand **Troubleshooting**, click **Logs and trace**, and then click the server name. For example, click **server1**.
4. Click **Diagnostic Trace**.
5. Click **Change log detail levels**.
6. In the box, add the following text:

   `com.ibm.sifs*=all`

   If you already have an entry in the box, use a colon to separate them. For example,

   `*=info: com.ibm.sifs*=all`

   `*=info` is the default value, which set the logging level to INFO for all applications.
7. Click **Apply**, and then click **Save**.

Note: If you no longer want detailed logging for IBM Surveillance Insight for Financial Services, you can change the logging level back to the original setting.

Deploying the REST application in WebSphere Application Server

You deploy the REST application by using a war file.

Procedure

1. Open a web browser.
2. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is `http://servername:9060/ibm/console` where *servername* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.
3. Expand **Applications**, click **New Application**, and click **New Enterprise Application**.
4. Browse to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/Services/FinancialMkts_SurveillanceInsight_ServicesContent/webapps` directory, select `com.ibm.sifs.service.data.war`, and click **Open**.
5. Click **Next**.
6. On the **How do you want to install the application** page, select **Detailed**, and click **Next**.
7. Click **Continue**.
8. In the list of steps, click **Step 4**.
9. Select the application, and then click **Reference shared libraries** to select the library that you created for Apache Kafka. For more information, see "Creating shared libraries for Apache Kafka for the REST applications" on page 43. Set the shared library for both the application and the module.
10. In the list of steps, click **Step 8**.
11. Enter `/surveillance` in **Context Root**.
12. In the list of steps, click **Step 9**.
13. Map the security role to the admin group.
14. Click **Next** to complete the wizard, and then click **Finish**.
15. Click **Save**.

16. Expand **Applications** > **Application Types**, and click **WebSphere enterprise applications**.
17. Select **com_ibm_sifs_service_data_war**, and click **Start**.

Deploy the ADAMs REST application

You must deploy the `com.ibm.sifs.service.adams.war` file in WebSphere Application Server to use the REST services for the ADAMs component.

Ensure that you have enabled security in WebSphere Application Server before you deploy the REST application. For more information, see "Enabling security in WebSphere Application Server to deploy the REST applications" on page 43.

Ensure that you have created the Apache Kafka shared libraries. For more information, see "Creating shared libraries for Apache Kafka for the REST applications" on page 43.

Adding JNDI variables for the ADAMs REST application

You must add JNDI variables WebSphere Application Server so that you can run the ADAMs REST application.

Procedure

1. Open a web browser.
2. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is `http://`*`servername`*`:9060/ibm/console` where *`servername`* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.
3. Expand **Environment** > **Naming** > **Name space bindings**, and then click **New**.
4. Add a binding type:
   a. For the **Binding type**, select **String**, and click **Next**.
   b. In **Binding identifier**, enter `ADAMS_HOME`.
   c. In **Name**, enter `ADAMS_HOME`.
   d. In **String value**, enter `/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary`.
   e. Click **Finish**.
5. Repeat step 4 to add the following binding types:

Binding identifier	Name	String value
`ADAMS_BASELINE`	`ADAMS_BASELINE`	`/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/resources/config/jar/baseline_trade.properties`
`ADAMS_ETL_RUNPROPS`	`ADAMS_ETL_RUNPROPS`	`/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/resources/config/jar/run-finance-sample-email-etl.props`

Binding identifier	Name	String value
SERVER_TRUSTSTORE	SERVER_TRUSTSTORE	The default path is <WASROOT>/profiles/<ProfileName>/config/cells/<cellname>/nodes/<node name>/trust.p12
SERVER_TRUSTSTORE_PWD	SERVER_TRUSTSTORE_PWD	WebAS

Creating shared libraries for the ADAMs REST application

You must create a shared library in WebSphere Application Server to access the jar files that are needed when you run the ADAMs REST application.

Procedure

1. On the services node computer, go to the IS_FinancialMkts_SurveillanceInsight_1.2/Services/FinancialMkts_SurveillanceInsight_ServicesContent/lib directory. By default, IS_FinancialMkts_SurveillanceInsight_1.2 is in /opt/IBM.
2. Copy com.ibm.sifs.security.kafka-<latest_version>.jar to the /home/streamsadmin/ directory.
3. Open a web browser.
4. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is http://*servername*:9060/ibm/console where *servername* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.
5. Expand **Environment**, and click **Shared libraries**.
6. Click **New**.
7. In **Name**, enter ADAMSAPI.
8. In **Classpath**, enter the following text:

```
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/eaves-pipeline-1.1-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/datamgt-store-2.1-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/
   analytic-copy-store-1.0-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/
   analytic-bayesiannvstore-1.0-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/
   analytic-email-opennlp-1.0-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/
   analytic-emotion-aggregation-1.1-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/
   analytic-emotion-scoring-1.1-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/analytic-graph-1.1-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/analytic-lof-1.0-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/
   analytic-email-core-1.13-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/
   analytic-natstore-to-relational-1.1-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/analytic-ui-done-1.0-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/analytic-alerts-1.0-SNAPSHOT.jar
/home/IBM_Adams/eaves-pipeline-pom-1.1-SNAPSHOT-binary/lib/flowmgt-wrapper-1.6-SNAPSHOT.jar
```

9. Click **Apply**, and then click **Save**.

Adding users and groups for the ADAMs REST application

You add users for the ADAMs REST application by using the WebSphere Administrative Console.

Procedure

1. Open a web browser.
2. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is `http://servername:9060/ibm/console` where *servername* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.
3. Expand **Users and Groups**, and click **Manage Groups**.
4. Click **Create**.
5. Enter `admin` in **Group name**, and click **Create**.
6. Click **Close**.
7. Click **Manage Users**.
8. Click **Create**.
9. Enter `ibmrest1` in **User ID**.
10. Click **Group Membership**.
11. From the **Available** box, select `admin`, and click **< Add** to move it to the **Mapped To** box.
12. Click **Close**.
13. Enter values in the remaining required fields, and click **Create**.
14. Click **Close**.

Deploying the ADAMs REST application in WebSphere Application Server

You deploy the ADAMs REST application by using a war file.

Procedure

1. Open a web browser.
2. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is `http://servername:9060/ibm/console` where *servername* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.
3. Expand **Applications**, click **New Application**, and click **New Enterprise Application**.
4. Browse to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/Services/FinancialMkts_SurveillanceInsight_ServicesContent/webapps` directory, select `com.ibm.sifs.service.adams.war`, and click **Open**.
5. Click **Next**.
6. On the **How do you want to install the application** page, select **Detailed**, and click **Next**.
7. Click **Continue**.
8. In the list of steps, click **Step 4**.
9. Select the application, and then click **Reference shared libraries** to select the library that you created for Apache Kafka. For more information, see "Creating shared libraries for Apache Kafka for the REST applications" on page 43. Set the shared library for both the application and the module.
10. In the list of steps, click **Step 8**.
11. Enter `/adams` in **Context Root**.
12. In the list of steps, click **Step 9**.

13. Map the security role to the admin group.
14. Click **Next** to complete the wizard, and then click **Finish**.
15. Click **Save**.
16. Expand **Applications** > **Application Types**, and click **WebSphere enterprise applications**.
17. Select **com_ibm_sifs_service_adams_war**, and click **Start**.

Deploying the Insights REST application in WebSphere Application Server

You deploy the Insights REST application by using a war file. The Insights REST application provides the user interface.

Procedure

1. Open a web browser.
2. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is `http://servername:9060/ibm/console` where *servername* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.
3. Expand **Applications**, click **New Application**, and click **New Enterprise Application**.
4. Browse to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/Services/FinancialMkts_SurveillanceInsight_ServicesContent/webapps` directory, select `com.ibm.sifs.service.insights.war`, and click **Open**.
5. Click **Next**.
6. On the **How do you want to install the application** page, select **Fast Path**, and click **Next**.
7. In the list of steps, click **Step 2: Map modules to servers**.
8. Select the check box for the module.
9. In the list of steps, click **Step 4: Map context roots for Web modules**.
10. Enter `/surveillanceui` in **Context Root**.
11. Click **Next** to complete the wizard, and then click **Finish**.
12. Click **Save**.
13. Expand **Applications** > **Application Types**, and click **WebSphere enterprise applications**.
14. Select **com_ibm_sifs_service_insights_war**, and click **Start**.

Deploying the Analytics Content application to WebSphere Application Server

After you run the `install.py` script for the Analytics Content, you must deploy the applications to WebSphere Application Server. The applications are generated by the `install.py` script.

Procedure

1. Go to the <*target_directory*>`/webapps` directory, where <*target_directory*> is the directory that you used when you ran `install.py`.

Ensure that the directory contains several `war` files that were generated by the `install.py` script. If no `war` files are in the directory, you must run the `install.py` script again.

2. Open a web browser.
3. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is `http://`*`servername`*`:9060/ibm/console` where *`servername`* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.
4. Enter the **User ID** and **Password**, and click **Log in**.
5. Expand **Applications** > **Application Types** > **WebSphere enterprise applications**.
6. Install the application war files from the <*`target_directory`*>`/webapps` directory.
   a. Click **Install**.
   b. Select **Local file system**, click **Browse**, locate and select `admin-webapp.war`, and click **Open**.
   c. Click **Next**.
   d. Select **Fast Path**, and click **Next**.
   e. Leave the default settings and click **Next** for steps 1 - 3.
   f. On Step 4, enter `/admin-webapp` as the **Context Root** value, and click **Next**.
   g. Click **Finish**, and then click **Save**.
   h. Click **Install**, and repeat the steps for each of the applications.

      As you install the applications, ensure that you use the following **Context Root** values for each application:

Application	Context Root
`shared-web-deps.war`	`/shared-web-deps`
`ui-datamgt-birt.war`	`/ui-datamgt-birt`
`ui-v3.war`	`/ui-v3`

7. Add the shared library to the `ui-datamgt-birt` application.
   a. In the **Enterprise Applications** list, click `ui-datamgt-birt_war`.
   b. Under **Detail Properties**, click **Class loading and update detection**.
   c. Under **Class loader order**, select **Classes loaded with local class loader first (parent last)**.
   d. Click **Apply**, and then click **Save**.
   e. In the **Enterprise Applications** list, click `ui-datamgt-birt_war` again.
   f. Under **Modules**, click **Manage Modules**.
   g. Click **Eclipse BIRT Report Viewer**.
   h. In **Class loader order**, select **Classes loaded with local class loader first (parent last)**.
   i. Click **Apply**, and then click **Save**.
8. Restart the WebSphere Application Server instance. For example, in a terminal window, go to the `/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin` directory, and enter the following commands:
   a. `./stopServer.sh server1`
   b. `./startServer.sh server1`

Deploying the SIFS Content application to WebSphere Application Server

You deploy the SIFS Content application to WebSphere Application Server.

Procedure

1. Open a web browser.
2. In the address bar, type the address for the WebSphere Admin Console.

   The Admin console address is `http://servername:9060/ibm/console` where *servername* is the name or IP address for the computer where you installed WebSphere Application Server and WebSphere Software Development Kit Java Technology Edition.
3. Enter the **User ID** and **Password**, and click **Log in**.
4. Expand **Applications** > **Application Types** > **WebSphere enterprise applications**.
5. Install the application war files from the `IS_FinancialMkts_SurveillanceInsight_1.2/Services/FinancialMkts_SurveillanceInsight_ServicesContent/webapps` directory.
   a. Click **Install**.
   b. Select **Local file system**, click **Browse**, locate and select `SpoofingRestService.war`, and click **Open**.
   c. Click **Next**.
   d. Select **Fast Path**, and click **Next**.
   e. Leave the default settings and click **Next** for steps 1 - 3.
   f. On Step 4, enter `/rest` as the **Context Root** value, and click **Next**.
   g. Click **Finish**, and then click **Save**.
   h. Click **Install**, and repeat the steps for each of the applications.

      As you install the applications, ensure that you use the following **Context Root** values for each application:

Application	Context Root
com.ibm.sifs.service.adams.war	/adams
com.ibm.sifs.service.data.war	/surveillance
com.ibm.sifs.services.insights.war	/surveillanceui
com.ibm.sifs.services.nlp.war	/surveillance/comm/analytics

6. Restart the WebSphere Application Server instance. For example, in a terminal window, go to the `/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin` directory, and enter the following commands:
   a. `./stopServer.sh server1`
   b. `./startServer.sh server1`

Configuring the Services content for IBM Trade Surveillance Analytics

To configure the Services content for IBM Trade Surveillance Analytics, you must create some directories, run some IBM Streams jobs, and then install a graph adapter.

Before you begin

You must install the IBM Trade Surveillance Analytics component.

- For information about downloading IBM Trade Surveillance Analytics, see "Downloading and decompressing the installation files" on page 29.
- For information about deploying the content, see "Using the solution installer to deploy the remaining solution components" on page 33.

Procedure

1. Change to the IBM Streams user. For example, change to the streamsadmin user, `su streamsadmin`.
2. Go to the `/home/streamsadmin` directory.
3. Create the following directories:
   - `EOD`
   - `trade`
   - `execution`
   - `order`
   - `quote`
   - `SILogs`
   - `processed`
   - `score`
   - `data`
   - `output`
4. From the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/Analytics/FinancialMkts_SurveillanceInsight_DataContent/PumpDumpSolution/PDZDataSet/Pnd_dailydata` directory, copy `initPositions.csv` to the `/home/streamsadmin` directory.
5. In the `/home/streamsadmin` directory, change the permissions for all of the files to the streamsadmin user and group.

   For example, enter the following command:

   ```
   chown -R streamsadmin:streamsadmin /home/streamsadmin
   ```
6. Go to the `/home/streamsadmin/components/trade` directory, and run the following commands in the indicated directories:
   a. In `MarketStreamSimulator`, run

      ```
      sc -M com.ibm.sifs.utils.loader::LoaderMain
      ```
   b. In `SurveillanceBase`, run

      ```
      sc -M com.ibm.sifs.common::Composite
      ```
   c. In `SpoodingRules`, run

      ```
      sc -M com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain -t
      /home/streamsadmin/components/trade/SurveillanceBase
      ```

      ```
      sc -M
      com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer -t
      /home/streamsadmin/components/trade/SurveillanceBase
      ```
   d. In `RDBPersistence`, open `etc/connections.xml` in a text editor, and set the SIFS database user name and password. Save the file, and then run the following command:

      ```
      sc -M com.ibm.sifs.persistence.relational::RDBMain -t
      /opt/ibm/InfoSphere_Streams/4.2.0.0/toolkits/com.ibm.streams.db
      ```

e. In PumpDumpSolution/TradeSummary, run

```
sc -M com.ibm.sifs.pnd.sa.tradesummary::TradeSummaryMain
```

f. In PumpDumpSolution/PositionSummary, run

```
sc -M com.ibm.sifs.pnd.sa.positionsummary::PositionSummaryMain
```

g. In PumpDumpSolution/com.ibm.sifs.pnd.sa.pndrules, run

```
sc -M com.ibm.sifs.pnd.sa.pndrules::PnDRulesMain
```

7. Go to the /home/streamsadmin/trade/MarketDataLoader directory, and run the following commands in the indicated directories to start the Pump and Dump analysis:

a. In MarketDataLoader, run

```
streamtool submitjob -C data-directory=/home/streamsadmin/
./output/com.ibm.sifs.utils.loader.LoaderMain.sab --embeddedzk -d
StreamsDomain -i SIInstance
```

b. In RDBPersistence, run

```
streamtool submitjob -C data-directory=/home/streamsadmin/
./output/com.ibm.sifs.persistence.relational.RDBMain.sab --embeddedzk
-d StreamsDomain -i SIInstance
```

c. In SpoofingRules, run

```
streamtool submitjob -C data-directory=/home/streamsadmin -d StreamsDomain
-i SIInstance -P com.ibm.sifs.common::DetectHighOrderCancellation.threshold=0.9,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.PostSpoofingInterval=30,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.orderWindowSize=45,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.bulkOrderThreshold=40000,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.priceTrendTolerance=10.0,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.cancelRatioWindowduration=45,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.dropthreshold=0.003,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.profitThreshold=15000,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.risethreshold=0.00008,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.bulkExecThreshold=5000,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.execWindowSize=1,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.priceChangeWindowslide=1.0,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.priceChangeWindowduration=300,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.usecaseID=2,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.SpoofingInterval=360,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.cancellationInterval=45,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.eventWindowSize=325,
com.ibm.sifs.spoofing.sa.spoofingrules::SpoofingRulesMain.cancelRatioWindowslide=0.5
./output/com.ibm.sifs.spoofing.sa.spoofingrules.SpoofingRulesMain.sab
```

```
streamtool submitjob -C data-directory=/home/streamsadmin -d StreamsDomain
-i SIInstance -P com.ibm.sifs.common::DetectHighOrderCancellation.threshold=0.9,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.PostSpoofingInterval=30,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.orderWindowSize=45,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.bulkOrderThreshold=40000,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.priceTrendTolerance=10.0,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer
  .cancelRatioWindowduration=15,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.dropthreshold=0.007,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.profitThreshold=15000,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.risethreshold=0.00008,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.bulkExecThreshold=5000,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.execWindowSize=1,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer
  .priceChangeWindowslide=1.0,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer
  .priceChangeWindowduration=300,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.usecaseID=2,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.SpoofingInterval=300,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.cancellationInterval=45,
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer.eventWindowSize=190,
```

```
com.ibm.sifs.spoofing.sa.spoofingrules.buyer::SpoofingRulesBuyer
  .cancelRatioWindowslide=0.5
./output/com.ibm.sifs.spoofing.sa.spoofingrules.buyer.SpoofingRulesBuyer.sab
```

d. In PumpDumpSolution/TradeSummary, run

```
streamtool submitjob –C data-directory=/home/streamsadmin/
output/com.ibm.sifs.pnd.sa.tradesummary.TradeSummaryMain.sab
--embeddedzk -d StreamsDomain -i SIInstance
```

e. In PumpDumpSolution/PositionSummary, run

```
streamtool submitjob –C data-directory=/home/streamsadmin/
output/com.ibm.sifs.pnd.sa.positionsummary.PositionSummaryMain.sab
--embeddedzk -d StreamsDomain -i SIInstance
```

f. In PumpDumpSolution/com.ibm.sifs.pnd.sa.pndrules, run

```
streamtool submitjob –C data-directory=/home/streamsadmin/
output/com.ibm.sifs.pnd.sa.pndrules.PnDRulesMain.sab --embeddedzk -d
StreamsDomain -i SIInstance
```

Loading the structured sample data for Pump and Dump

You load the structured sample data by running the LoadPnd.sh script.

Procedure

1. Go to the /home/streamsadmin/Data/PumpDumpSolution/PDZDataSet/Pnd_dailydata directory.
2. Run the following command:

   ```
   ./LoadPnD.sh
   ```

   The script loads the day wise data for the ticker PDZ, triggers the Pump and Dump jobs, and updates the following graphs:
   - trade summary data in **ticker_summary_<symbol>** and **movingaverage** graphs
   - position summary data in **_position** and **Top5Traders** graphs
   - EOD record in **symbol_eod** graph
   - Pump and Dump scores in **pumpdump** and **<date>_PD** graph
   - trader scores in score and **<date>_score** graph
3. Verify that the content was loaded. Enter the following command:

   ```
   curl -X GET -H "X-SystemG-User: sguser_analytics" http://localhost:8002/
   gshell/graphs/pumpdump/vertices?_cmd=count
   ```

   The command should return a result that includes {"number of vertices":2265}. Then, you can continue with the remaining steps.
4. From the /home/streamsadmin/Data/PumpDumpSolution/PDZDataSet directory, copy pnd_pdz_ticker_price.csv to the $*STREAMS_ADAPTOR*/data/price directory.
5. From the /home/streamsadmin/Data/PumpDumpSolution/PDZDataSet/Pnd_dailydata directory, copy pnd_pdz_execs.csv to the $*STREAMS_ADAPTOR*/data/order directory.

   This task can take about 20 minutes to complete.
6. Verify that the content was loaded. Enter the following command:

   ```
   curl -X GET -H "X-SystemG-User: sguser_data" http://localhost:8002/
   gshell/graphs/order_vertex/vertices?_cmd=count
   ```

   The command should return a result that includes {"number of vertices":657121}.

Setting up the sample unstructured data

Use the following steps to configure the system to load unstructured data, and to test the configuration by loading the sample data that is provided with IBM Surveillance Insight for Financial Services.

Before you begin

The Analytics Content component must already be set up in your cloud environment before you can load the sample data.

Procedure

1. Go to the <*target_directory*> directory, where <*target_directory*> is the directory you entered when you ran `install.py`.
2. Enter the following command to set your environment variables:
   `. ./setenv.sh`
3. Go to the <*target_directory*>`/eaves-pipeline-pom-1.1-SNAPSHOT-binary` directory.
4. Enter the following command to create an **EAVES** environment variable:
   `export EAVES=$(pwd)`
5. Go to the <*target_directory*>`/eaves-pipeline-pom-1.1-SNAPSHOT-binary/resources/config/jar` directory.
6. Open `run-finance-sample-email-etl.props` in a text editor.
7. Edit the following values to match your environment:
   a. Change **FILESTORE_dir** to the `IS_FinancialMkts_SurveillanceInsight_1.2/Analytics/FinancialMkts_SurveillanceInsight_DataContent/EmailData` directory. For example, change the value to `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/Analytics/FinancialMkts_SurveillanceInsight_DataContent/EmailData`.
   b. Change **EMAIL_STORENAME** value to `emailtable150.csv`, which is the name of the file in the **FILESTORE_dir** directory.
8. Save and close the file.
9. Go to the <*target_directory*>`/eaves-pipeline-pom-1.1-SNAPSHOT-binary/scripts` directory.
10. Ensure that your Java variables are set.
    a. Update your **JAVA_HOME** variable to use the Java that you installed for WebSphere Application Server. For example, enter the following command:
       `export JAVA_HOME=/opt/IBM/WebSphere/AppServer/java/8.0`
    b. Add the Java bin directory to your **PATH** environment variable. For example, enter the following command:
       `export PATH=/opt/IBM/WebSphere/AppServer/java/8.0/bin:$PATH`
11. Set the **UIMA_HOME** environment variable to point to the <*target_directory*>`/opt/apache-uima/bin` directory.
    For example, `export UIMA_HOME=/home/IBM_Adams/opt/apache-uima/bin`.
12. Enter the following command to load the data:
    `./run-finance-sample-email-etl.sh`
13. After the script is run, you can verify that the data was loaded by using the IBM Surveillance Insight for Financial Services Application Management console.
    a. In a web browser, enter the following URL:

`http://servername:port/ui-datamgt-birt`

Where *servername* is that name of the server where you installed the Trade Surveillance component, and *port* is the port number that you are using for the WebSphere Application Server profile. The default port number is 9080.

b. Click **All Batch Runs**.

c. Click the batch name for the job that you ran. Each time that you run `./run-finance-sample-email-etl.sh`, a batch name is created based on the time stamp of when you ran the script.

Note: The **Batch Name** value is used in the following step.

d. Under **Actions**, click **details**.

e. Click **View Data** to see the data that was loaded.

14. Go to the `<target_directory>/eaves-pipeline-pom-1.1-SNAPSHOT-binary/resources/config/jar` directory.
15. Open `baseline_trade.properties` in a text editor.
16. Change the `ETL_SAMPLE_BATCHNAME_EMAIL` value to be the same as the **Batch Name** value.
17. Save and close the file.
18. Go to the `FinancialMkts_SurveillanceInsight_DataContent/EmailData` directory, and decompress the `emailtext-sample.tar.gz` file. For example, in a terminal window, enter `tar xvf emailtext-sample.tar.gz`

    The file is decompressed to a directory that is named `emaildata` in the current location.
19. From the `EmailData` directory, copy the `emailtext` directory to the document root directory for your web server. For example, copy the `emailtext` directory to the `/opt/rh/httpd24/root/var/www/html` directory.
20. Go to the `<target_directory>/eaves-pipeline-pom-1.1-SNAPSHOT-binary/scripts` directory.
21. Enter the following command to combine the analysis of the structured data with the unstructured (email) data:

    This step can take some time to run. For example, on a computer with 32 GB of RAM, this step can take 30 - 60 minutes.

    `./run-finance-pipeline.sh`

Loading the sample data for spoofing

You load the sample data for spoof detection by moving csv files.

Procedure

1. Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/SampleData/FinancialMkts_SurveillanceInsight_SampleDataContent/SpoofingData` directory.
2. Decompress `SpoofingData-daywise.zip`.

   This file contains data in 6 csv files for May6th and May8th for the ticker that is named KO.
3. Copy the quotes, orders, and execution files to their respective `/home/streamsadmin/quote`, `home/streamsadmin/orders`, and `/home/streamsadmin/execution` folders.

   The **quotes_sample**, **orders**, and **execution** graphs are updated. The **ALERT** and **SCORE** tables in the SIFS database are updated.

Configuring the Services content for IBM Electronic Communication Surveillance Analytics

To configure the Services content for IBM Electronic Communication Surveillance Analytics, you must create some directories and run some IBM Streams jobs.

Before you begin

You must install the IBM Electronic Communication Surveillance Analytics component.

- For information about downloading IBM Electronic Communication Surveillance Analytics, see "Downloading and decompressing the installation files" on page 29.
- For information about deploying the content, see "Using the solution installer to deploy the remaining solution components" on page 33.

Procedure

1. Change to the IBM Streams user. For example, change to the streamsadmin user, `su streamsadmin`.
2. Go to the `/home/streamsadmin` directory.
3. Create the following directories:
   - `emailcsv`
4. From the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Ecomm_1.2/Services/components` directory, copy the `ecomm` directory to the `/home/streamsadmin/components` directory.
5. From the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Ecomm_1.2/Services/components/bin` directory, copy the contents to the `/home/streamsadmin/bin` directory.
6. Ensure that you created the `consumer.properties` file and configured it. For more information, see "Configuring SSL for Apache Kafka" on page 23.
7. Go to the `/home/streamsadmin/bin` directory.
8. Run the following command:
   ```
   ./install_ecomm.sh
   ```
9. If you want to run the product for a demonstration or with the sample data that is provided with the product:
   a. Open `start.ecomm.sh` in a text editor.
   b. Change **`ANALYSISMODE`** to **`OFFLINE`**.
   c. Save, and close the file.
10. Run the following command:
    ```
    ./start_ecomm.sh
    ```

Loading the sample data for IBM Electronic Communication Surveillance Analytics

Procedure

1. Load the master data.
   a. Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/SampleData/ FinancialMkts_SurveillanceInsight_SampleDataContent/EComm/MasterData` directory.
   b. Change the database instance owner user. For example, `su db2inst1`.

c. Connect to the SIFS database:

   ```
   db2 connect to SIFS user db2inst1 using password
   ```

d. Run the following commands to the sample data into the SIFS database:

   - `db2 IMPORT FROM channel.csv OF DEL "INSERT INTO SIFS.CHANNEL"`
   - `db2 IMPORT FROM party.csv OF DEL "INSERT INTO SIFS.PARTY"`
   - `db2 IMPORT FROM party_contact_point.csv OF DEL "INSERT INTO SIFS.PARTY_CONTACT_POINT(PARTY_ID,CHANNEL_ID,CONTACT_POINT)"`

2. Change back to the root user.
3. Publish the policy document to the REST application.
   a. Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/SampleData/ FinancialMkts_SurveillanceInsight_SampleDataContent/EComm/Policy` directory.
   b. Publish the policy document to the REST service:

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

      Ensure that the paths and user name and password are correct for your environment.
4. Post the email data to the REST application.
   a. Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/SampleData/ FinancialMkts_SurveillanceInsight_SampleDataContent/EComm/SnapshotEmails` directory.
   b. Post the emails in ascending order to the REST service. For example, use the following command to post the email data:

      ```
      curl -k -H 'Content-Type: text/plain' -H 'source:Actiance' -X POST --data-binary @snapshot_002a5d12-4956-4c8b-8fcf-11041eed86d72585479296070162.local.txt --user actiance1:actpwd1 --digest https://localhost:9443/surveillance/data/email
      ```

      Ensure that the paths and user name and password are correct for your environment.
5. Post the chat data to the REST application.
   a. Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/SampleData/ FinancialMkts_SurveillanceInsight_SampleDataContent/EComm/SnapshotEmails` directory.
   b. Post the chat data in ascending order to the REST service. For example, use the following command to post the chat data:

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

      Ensure that the paths and user name and password are correct for your environment.
6. Copy the `commdata` directory to the `/opt/rh/httpd24/root/var/www/html/` directory on the computer where you web server is installed.

   You can verify that the data was loaded on the **Know Your Employee** page.

Configuring the Services content for IBM Voice Surveillance Analytics

To configure the Services content for IBM Voice Surveillance Analytics, you must create some directories and run some IBM Streams jobs.

Before you begin

You must install the IBM Voice Surveillance Analytics component.

- For information about downloading IBM Voice Surveillance Analytics, see "Downloading and decompressing the installation files" on page 29.
- For information about deploying the content, see "Using the solution installer to deploy the remaining solution components" on page 33.

Procedure

1. Change to the IBM Streams user. For example, change to the streamsadmin user, `su streamsadmin`.
2. Go to the `/home/streamsadmin` directory.
3. Create the following directories:
   - `audio`
4. From the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Voice_1.2/Services/components` directory, copy the `Voice` directory to the `/home/streamsadmin/components` directory.
5. Ensure that you created the `consumer.properties` file and configured it. For more information, see "Configuring SSL for Apache Kafka" on page 23.

   You can copy the `consumer.properties` file from the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Base_1.2/config/properties` directory as an example. Ensure that you update the file for your environment.
6. Ensure that you created the `decrypt.properties` file and configured it. For more information, see "Securing data at rest for Apache Kafka" on page 21.

   You can copy the `decrypt.properties` file from the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_Base_1.2/config/properties` directory as an example. Ensure that you update the file for your environment.
7. Go to the `/home/streamsadmin/components/Voice` directory.
8. Open the `build.sh` file in a text editor.
9. Edit the paths in the file to match your environment, and save and close the file.
10. Run the following command:

    ```
    sh build.sh
    ```
11. Open the `submitjob.sh` file in a text editor.
12. Edit the paths in the file to match your environment, and save and close the file.
13. Run the following command:

    ```
    sh submitjob.sh
    ```

Loading the sample data for IBM Voice Surveillance Analytics

Procedure

1. Go to the `/opt/IBM/IS_FinancialMkts_SurveillanceInsight_1.2/Data/Voice` directory.
2. Open the `VoiceClient/lib/kafka.properties` file in a text editor.
3. Update the Kafka SSL and encryption keystore location details. For example, refer to the `/home/streamsadmin/config/properties` directory for the `consumer.properties` and `encryption.properties` files.

4. Save and close the file.
5. Copy the `wav` files to the `/home/streamsadmin/audio` directory.
6. Run the following command:

   ```
   ./processvoice.sh voicemetadata.csv
   ```
7. Go to the `/home/streamsadmin/audio` directory.
8. Copy the directories in `/home/streamsadmin/audio` to `/opt/rh/httpd24/root/var/www/html/commdata/voice`.
9. Verify that the data was loaded by going to the **Know Your Employee** page and searching for `*` in the Communication search page.
10. To reload the data, edit the **`gcid`** value (the last column) in the `voicemetadata.csv` file, and repeating steps 5 - 9.

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_1.2/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.

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.

Useful Linux commands for installing IBM Surveillance Insight for Financial Services

Check your Apache HTTP version

Use the following command to check your Apache HTTP version:

```
/opt/rh/httpd24/root/usr/sbin/httpd -v
```

Check your PHP version

Use the following command to check the version of PHP:

```
/opt/rh/php54/root/usr/bin/php -v
```

Locating the `httpd.conf` file

Use the following command to locate the `httpd.conf` file on your file system:

```
/opt/rh/httpd24/root/usr/sbin/httpd -V
```

The results show variables and values. For example, use the command to confirm the Apache HTTP location and the location of the `httpd.conf` file.

```
-D HTTPD_ROOT="/opt/rh/httpd24/root/etc/httpd"

-D SERVER_CONFIG_FILE="conf/httpd.conf"
```

Viewing the installed Apache HTTP modules

Use the following command to see a list of the Apache HTTP modules that are installed:

```
/opt/rh/httpd24/root/usr/sbin/httpd -V
```

Starting and stopping Apache HTTP

Use the following command to start Apache HTTP server:

```
service httpd24-httpd start
```

Use the following command to stop Apache HTTP server:

```
service httpd24-httpd stop
```

Use the following command to restart Apache HTTP server:

```
service httpd24-httpd restart
```

Starting and stopping PHP

Use the following command to start PHP:

```
service php54-php-fpm start
```

Use the following command to stop PHP server:

```
service php54-php-fpm stop
```

Use the following command to restart PHP server:

```
service php54-php-fpm restart
```

Enabling both secure and non-secure communication for IBM DB2

Use the following command to enable IBM DB2 for both secure and non-secure communication:

```
db2set -i db2inst1 DB2COMM=SSL, TCPIP
```

You must restart the database to apply this setting.

Enabling archive logging for IBM DB2

By default, circular logging is already enabled for IBM DB2. To enable archive logging, use the following command:

```
db2 update db configuration for mydb using logarchmeth1 disk:/usr/db2inst1/
archived_logs
```

Note: Ensure that you set the path so that it is appropriate for your environment. The path in the command is `/usr/db2inst1/archived_logs`.

Starting and stopping IBM DB2

As the database instance owner (db2inst1), use the following command to start IBM DB2:

```
db2start
```

As the database instance owner (db2inst1), use the following command to stop IBM DB2:

```
db2stop
```

Removing a database

As the database instance owner (db2inst1), use the following command to drop a database:

```
db2 drop database database_name
```

Spoofing interface does not show the complete spoofing chart

The spoofing interface does not show the complete spoofing chart. Only the quote lines appear without the bars or only the bars appear.

To resolve the problem, restart the System G graph REST server and the HTTP server. Ensure that you kill the processes before you restart them.

Solution installer is unable to create the chef user

You are installing the components by using the solution installer. When you run the `setup.sh` script, you receive an error message that the solution installer is "Unable to create the chef user".

You can resolve this problem, by running the `cleanup.sh` script and then restarting the computer.

Naming exception trying to lookup DataSource:DATAMGT error

If you receive the following error, ensure that the WebSphere Application Server profile where you deployed the Surveillance Insight applications is running.

```
[2016-10-22 08:30:39,629(main)()]  DEBUG: DataSourceHelper.getDataSource:
   Lookup DataSource :DATAMGT
[2016-10-22 08:30:40,036(P=239653:O=0:CT)()]  FATAL: DataSourceHelper.getDataSource:
   Naming exception trying to lookup DataSource :DATAMGT
javax.naming.ServiceUnavailableException: A communication failure occurred while attempting
   to obtain an initial context with the provider URL: "corbaloc:iiop:localhost:2809".
   Make sure that any bootstrap address information in the URL is correct and that the target
   name server is running.  A bootstrap address with no port specification defaults to port 2809.
   Possible causes other than an incorrect bootstrap address or unavailable name server include
   the network environment and workstation network configuration.
   [Root exception is org.omg.CORBA.TRANSIENT: java.net.ConnectException:
   Connection refused:host=127.0.0.1,port=2809  vmcid: IBM  minor code: E02  completed: No]
      at com.ibm.ws.naming.util.WsnInitCtxFactory.mapInitialReferenceFailure(WsnInitCtxFactory.
        java:2512)
      at com.ibm.ws.naming.util.WsnInitCtxFactory.getWsnNameService(WsnInitCtxFactory.java:1642)
      at com.ibm.ws.naming.util.WsnInitCtxFactory.getRootContextFromServer(WsnInitCtxFactory.
        java:1154)
      at com.ibm.ws.naming.util.WsnInitCtxFactory.getRootJndiContext(WsnInitCtxFactory.java:1074)
      at com.ibm.ws.naming.util.WsnInitCtxFactory.getInitialContextInternal(WsnInitCtxFactory.
        java:618)
      at com.ibm.ws.naming.util.WsnInitCtx.getContext(WsnInitCtx.java:128)
      at com.ibm.ws.naming.util.WsnInitCtx.getContextIfNull(WsnInitCtx.java:765)
      at com.ibm.ws.naming.util.WsnInitCtx.lookup(WsnInitCtx.java:164)
      at com.ibm.ws.naming.util.WsnInitCtx.lookup(WsnInitCtx.java:179)
      at javax.naming.InitialContext.lookup(InitialContext.java:428)
```

NameNotFoundException from the run-finance-sample-email-etl.sh script

You are running the `run-finance-sample-email-etl.sh` script and you receive a `javax.naming.NameNotFoundException` error.

The full error message appears as:

```
javax.naming.NameNotFoundException: Context: civcez052Node01Cell/nodes/civcez052Node01/
   servers/server1, name: sslConnection=true;: First component in name sslConnection=true;
   not found. [Root exception is org.omg.CosNaming.NamingContextPackage.NotFound:
   IDL:omg.org/CosNaming/NamingContext/NotFound:1.0]
at com.ibm.ws.naming.jndicos.CNContextImpl.mapNotFoundException(CNContextImpl.java:4564)
at com.ibm.ws.naming.jndicos.CNContextImpl.doLookup(CNContextImpl.java:1822)
```

```
at com.ibm.ws.naming.jndicos.CNContextImpl.doLookup(CNContextImpl.java:1777)
at com.ibm.ws.naming.jndicos.CNContextImpl.lookupExt(CNContextImpl.java:1434)
at com.ibm.ws.naming.jndicos.CNContextImpl.lookup(CNContextImpl.java:616)
at com.ibm.ws.naming.util.WsnInitCtx.lookup(WsnInitCtx.java:165)
at com.ibm.ws.naming.util.WsnInitCtx.lookup(WsnInitCtx.java:179)
at javax.naming.InitialContext.lookup(InitialContext.java:428)
at com.ibm.research.eaves.datamgt.store.util.DataSourceHelper.getDataSource
   (DataSourceHelper.java:420)
at com.ibm.research.eaves.datamgt.store.util.DataSourceHelper.findOrCreateDataSource
   (DataSourceHelper.java:233)
at com.ibm.research.eaves.datamgt.store.util.DataSourceHelper.getDatabaseConnection
   (DataSourceHelper.java:87)
at com.ibm.research.eaves.datamgt.store.impl.store.relational.RelationalTableFactory.
   createRelationalTable(RelationalTableFactory.java:47)
at com.ibm.research.eaves.datamgt.store.impl.store.relational.RelationalStoreDataImpl.initialize
   (RelationalStoreDataImpl.java:444)
at com.ibm.research.eaves.datamgt.store.impl.store.StoreDAOImpl.createStoreData
   (StoreDAOImpl.java:552)
at com.ibm.research.eaves.datamgt.store.impl.store.StoreDAOImpl.createStoreData
   (StoreDAOImpl.java:537)
at com.ibm.research.eaves.flowmgt.wrapper.UIMAWrapperBase.process(UIMAWrapperBase.java:1591)
at org.apache.uima.analysis_component.JCasAnnotator_ImplBase.process(JCasAnnotator_ImplBase.java:48)
at org.apache.uima.analysis_engine.impl.PrimitiveAnalysisEngine_impl.callAnalysisComponentProcess
   (PrimitiveAnalysisEngine_impl.java:377)
at org.apache.uima.analysis_engine.impl.PrimitiveAnalysisEngine_impl.processAndOutputNewCASes
   (PrimitiveAnalysisEngine_impl.java:295)
at org.apache.uima.analysis_engine.asb.impl.ASB_impl$AggregateCasIterator.processUntilNextOutputCas
   (ASB_impl.java:567)
at org.apache.uima.analysis_engine.asb.impl.ASB_impl$AggregateCasIterator.<init>(ASB_impl.java:409)
at org.apache.uima.analysis_engine.asb.impl.ASB_impl.process(ASB_impl.java:342)
at org.apache.uima.analysis_engine.impl.AggregateAnalysisEngine_impl.processAndOutputNewCASes
   (AggregateAnalysisEngine_impl.java:267)
at com.ibm.research.eaves.flowmgt.wrapper.Runner.main(Runner.java:221)
```

To resolve this issue, ensure that the parameters in the `baseline_trade.properties` file are correct.

For example, ensure that the following values are correct:

```
EAVESDefault_DB_Url=jdbc:db2://datanode_computer:50001/RESULTS

SAMPLE_STORESPEC_basic=db2.datanode_computer.50001.RESULTS
```

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.

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.

Index