Sterling Integrator®

EBICS Banking Server Module Concepts

Version 5.1

Sterling Commerce
An IBM Company

Contents

Copyright

EBICS Banking Server Module Overview

Overview of EBICS Banking Server Module

Electronic Banking Internet Communication Standard (EBICS) is an Internet-based communication and security standard that is primarily used for remote data transfer between your organization and a bank for corporate payment transactions.

EBICS allows data file exchange independent of message standards and formats. EBICS uses established digital signature and encryption procedures. Its features are based on international standards for internet communication and improved security, for example, XML, HTTPS, TLS, and SSL. EBICS also has multibank capability wherein the corporate clients in the countries that have adopted EBICS can transact with any bank in those countries using the same software.

A range of prerequisites must be fulfilled by user under a partner for the users to be able to implement bank-technical EBICS transactions with a particular bank. The basic prerequisite to implement EBICS transactions is the signing of a contract between the partner and the bank. The following details are agreed upon in this contract:

- The nature of business transactions (bank-technical order types) the partner will conduct with the bank
- Information about the user’s bank accounts
- The partner’s users working with the bank's system
- The authorizations and permissions the users possess

The partner receives the bank’s access data (bank parameters) after the contract is signed. The bank will set up the partner and user master data in the bank system in accordance with the contractual agreements.

Other prerequisites are successful subscriber initialization, download of the bank’s public certificates by the user, and successful verification of the user's public certificates by the bank.

EBICS Banking Server Module of Sterling Integrator is a complete EBICS solution involving a bank, a partner, and user management, certificate management, secure file transaction, error recovery, and reporting. Use Sterling Integrator to send and receive EBICS transactions.

Note: EBICS Banking Server Module supports the French implementation of Electronic Banking Internet Communication Standard (EBICS) version 2.4.1. The German implementation of EBICS and VEU are not supported currently.

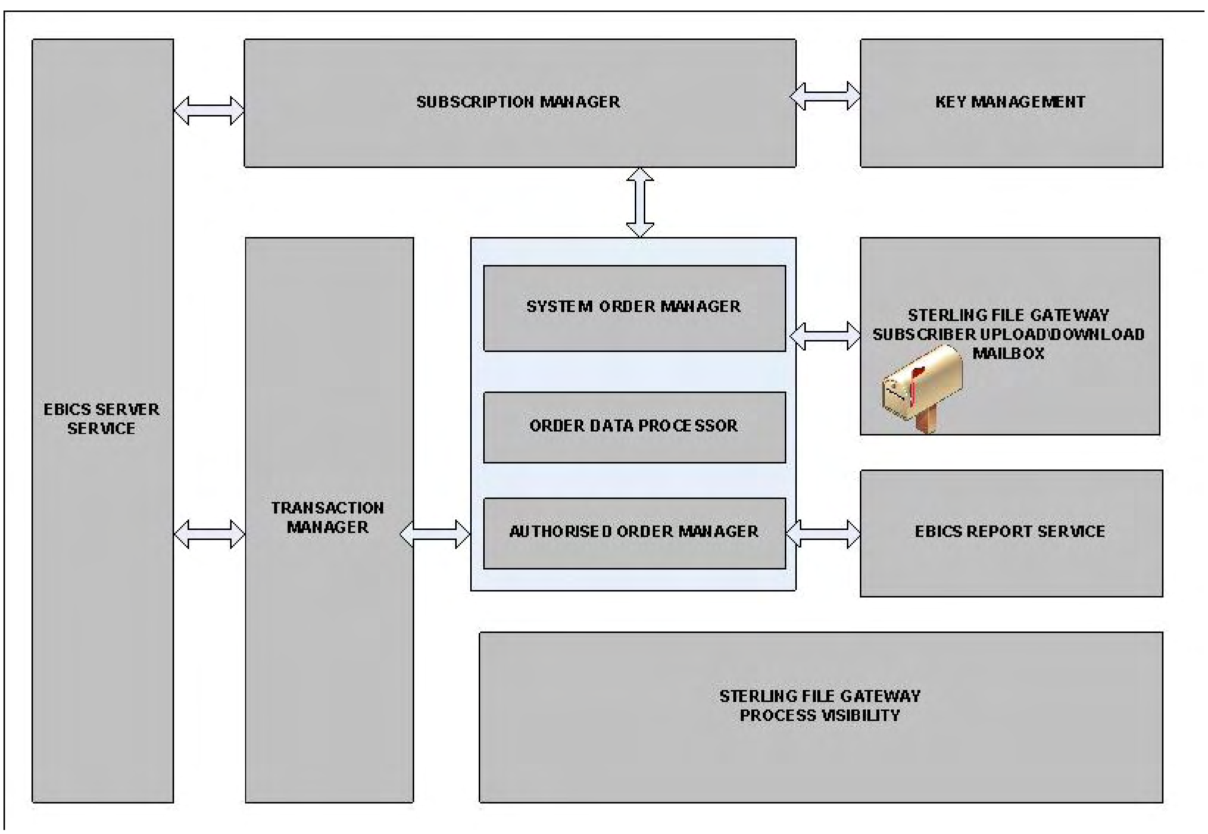
Sterling File Gateway is a Sterling Commerce application that operates on the Sterling Integrator platform, enabling secure file transfer between internal and external partners using either the same or different communication protocols, file naming conventions, and file formats. Sterling File Gateway supports the movement of large and high-volume file transfers, with visibility of file movement in a process-oriented and highly-scalable framework that alleviates file transfer challenges, such as protocol and file brokering, automation, and data security.

EBICS Banking Server Module Architecture

EBICS Banking Server Module Architecture

EBICS Banking Server Module enables you to transact with partners and users using EBICS. Its features include creating and managing profiles (bank, partner, and user), associating partners and users with order types and file formats, assigning user permissions, creating and managing certificates, processing of order data, storing and retrieving profile information, certificates, and messages, managing message flows and transaction flows, transferring files using secure protocols, and so on.

The following diagram illustrates the EBICS Banking Server Module architecture:



Subscription Manager includes the following features:

- Profile Management - For creating and managing bank, partner, and user profiles
- Order Type Configuration - For configuring order types and file formats
- Offer Configuration - For grouping a set of order types and file formats to a list of customers
- User Permission Configuration - For assigning order types and file formats to users
- Import of Subscription Manager Information - For importing configuration details related to bank, partner, user, offer, user permissions, order types, and file formats into the EBICS Banking Server Module from an external repository
- Export of Subscription Manager Information - For exporting configuration details related to bank, partner, user, offer, user permissions, order types, and file formats into an external repository from the EBICS Banking Server Module

Subscriber's upload and download mailboxes are configured in Subscription Manager during the user subscription setup.

Key Management interfaces mainly with Subscription Manager to create, update, delete, and query certificates.

Key Management includes the following features:

- Self-Signed certificates - For generating and managing self-signed certificates using 2048-key length
- CA certificates - For managing CA certificates
- Key storage - For providing the key stores for the certificates and managing the renewal and expiration of certificates
- Import adn Export certificates - For importing and exporting certificates
- Subscriber key validation - For validating user certificate hash values
- Certificate hash value - For supporting the creation of certificate hash value using SHA256

EBICS Server Service interfaces with Subscription Manager to retrieve the profile information of banks, partners, users, and order types necessary for verification and authentication of messages and transactions. It works in close collaboration with Transaction Manager to manage all the EBICS transactions.

EBICS Server Service includes the following features:

- Request and Response - For handling incoming EBICS requests (through HTTP and HTTPS) according to EBICS protocol specifications, and generating an appropriate response back to the requestor
- Message Flow - For managing the message flow for the initialization and file transfer phases of the EBICS transactions
- Authentication and Authorization - For performing message authentication and user authorization checks

Transaction Manager interfaces closely with the EBICS Server Service to manage the upload and download flow of system order types and bank-technical order types.

Transaction Manager includes the following features:

- Asynchronous Transaction - For managing the asynchronous transaction flow for upload bank-technical order type (FUL). It manages the authorized order processing flow in collaboration with the Order Data Processor to unpack the order data and deliver the unpacked order data to the destination upload mailbox as defined in the user profile settings.
- Synchronous Transaction - For managing the synchronous transaction flow for upload and download system order and bank-technical order types. It manages the system order processing, report processing (FDL, PSR) and download bank-technical order (FDL) processing flows.
- Segmentation and Recovery - For managing no-replay, segmentation, and error recovery

System Order Manager is responsible for updating and querying key management information and user referential information.

System Order Manager works closely with Transaction Manager and Subscription Manager to update and query the user's key certificates and referential information, and to download bank parameters and bank certificates.

Authorized Order Manager is responsible for initiating the Order Data Processor to unpack the order data received from the FUL order type request, routing the unpacked order data to the backend subscriber's upload mailbox, and renaming it according to a defined naming convention.

Order Data Processor is responsible for packing and unpacking order data. It interfaces with Subscription Manager and Transaction Manager to retrieve the relevant information required for packing and unpacking the order data. Its features include:

- Packing - For packing order data such as signing, compression, encryption, and base64 encoding depending on the requirement of the order type
- Unpacking - For unpacking order data such as verification, decompression, decryption, and base64 decoding depending on the requirement of the order type

Reporting Service is responsible for generating the Payment Status Report (PSR) associated with the unpacking of order data during an asynchronous upload of bank-technical order transaction flow.

Sterling File Gateway (SFG) uses templates to describe how each EBICS transaction is interpreted to determine how and where it should be delivered and provides visibility into the details of the transfers for auditing and troubleshooting.

Sterling File Gateway includes the following features:

- File or File Name Transformations - For mapping input to output file names, system-wide, group, and partner-specific policies, common file processing tasks such as compression and decompression, PGP encryption and decryption, and signing
- File Transfer Visibility - Events are recorded for monitoring and reporting; detailed tracking for input-output file structure processing and dynamic route determination; ability to view and filter data flows for all users
- Broad Communications Protocol Support - FTP, FTP/S, SSH/SFTP, SSH/SCP, and Connect:Direct are supported upon installation, and additional protocols (such as AS2, AS3, or Odette FTP) can be configured by using the extensibility feature
- Partner Interface (myFileGateway) - Web browser-based interface that enables partners to upload/download files, subscribe to notifications about events, manage passwords, search and view file transfer activity, and generate reports about file transfer activity
- Flexible Mailbox Structures - Ability to specify mailbox structures that leverage pattern-matching policies and specify attributes that must be true for all partners or a subset of partners
- Dynamic Routing - Consumer derived at run time, either through mailbox structure, file name, business process-derived consumer name, or map-derived consumer name

Managing Subscription Manager Information

Managing Subscription Manager Information

The Subscription Manager menu in Sterling Integrator enables you to:

- Create and manage bank, partner, and user profiles in the system database
- Create and manage offers
- Assign order types and file formats to an offer
- Assign permissions to users

A bank can have only one profile with a unique bank ID. A bank profile contains the following information:

- Unique ID of the bank

Note: Each bank ID should have a unique port number.

- Name of the bank
- Address of the bank
- Public and private encryption, authentication and identification certificates
- HTTP URL of the bank
- EBICS protocol version

A bank can have multiple URLs. The corresponding bank URL is given to a user to send requests to the bank. The Uniform Resource Indicator (URI) is configured in the HTTP Server Adapter to listen at the port and receive EBICS requests, if any.

The following versions of bank protocol and process types are supported:

- EBICS protocol version - H003
- Signature versions - A005, A006
- Authentication version - X002
- Encryption version - E002

Each partner can have one or more account information and partner IDs. You must specify the account number, either in national (German) or international (IBAN) format. You can associate a partner ID with an offer. The partner profile contains the following information:

• Unique ID of the partner
• Organization code of the partner
• Name of the partner
• Address of the partner
• Account ID and account holder’s name
• Currency in which transaction is performed
• Account number
• Bank code

A user can be under one or more partners. A bank can create a user with or without associating a user with a partner. To enable exchange of EBICS messages between a partner and a user, you must associate a user ID with a partner ID.

A user transmits the public certificates to the bank through two independent communication paths:

• INI - Sends the public bank-technical key
• HIA - Sends the public identification and authentication key and the public encryption key

When a user is first assigned to a partner, the status of the user is New. If the user sends only the INI request to the corresponding bank, the status is changed to Partly Initialized (INI). If the user sends only the HIA request to the bank, the status is changed to Partly Initialized (HIA). After the user sends both the INI and HIA requests to the bank, the status is changed to Initialized. The user mails the initialization letters of the INI and HIA keys to the bank. When the bank receives the initialization letters pertaining to INI and HIA, it verifies the hash values in the certificates against its database. After successful verification, the status of the user is set to Ready, indicating that the user can now transact with the bank. The user then downloads the bank's public certificates by using the HPB system order type.

Use the EBICS Subscription Manager Service to validate the keys on the INI and HIA initialization letters. On successful validation, the status of the user is updated, for example, Ready, indicating that the user has sent the HIA and INI initialization letters to the bank. You can also use this service to import or export subscription manager data to or from the bank system database.

The user profile contains the following information:

• Unique ID of the user
• Name of the user
• Address of the user
• Partner ID to which the user is associated
• Mailbox settings to enable uploading, downloading, and archiving of messages

EBICS order types specify the various transactions that can take place between the EBICS server and an EBICS client. An order type can have zero or more file formats. You can associate file formats with the bank-technical upload and download order types. You can use upload order types to upload order data from an EBICS client to an EBICS server and download order types to download order data from an EBICS server to an EBICS client. An order type contains the following attributes:

• The order type
• EBICS protocol version
• Transfer type - Upload or Download
• Order data type - System or Technical

A file format contains the following attributes:

- The file format
- Country code of the file format

A bank can create one or more offers. An offer provides an easy method of grouping a set of order types and file formats to a list of partners. Each partner is allocated a list of order types to enable transactions between the bank and the partner. An offer provides an easy way for the bank to set up a contract with the partner. An offer contains the following information:

- Bank ID
- Name of the offer
- The order types and file formats using which the partner can exchange messages
- Level of authorization for the order type
- Number of signatures required to authorize the order

A partner can be associated with one or more users. A bank assigns the following permissions to a user:

- The order types and file formats using which the user can exchange messages
- Level of authorization for the order type
- The maximum amount a user can transact
- The currency in which the maximum amount for the user is specified

Managing EBICS Transactions

Managing EBICS Transactions

Transaction Manager in the EBICS Server is responsible for maintaining the transaction states. It determines the segment that is required to generate the XML response message.

Transaction Manager handles the upload and downloadtransaction flows and supports segmentation and recovery of order data.

Upload From a Subscriber (FUL)

The upload transaction comprises the following phases:

- Initialization
- Data Transfer

The user sends the upload (FUL) request to the bank. FUL is a bank-technical upload order type.

The EBICS Order Authorization service handles incoming order requests for the bank-technical upload order type. If an order has obtained the number of signatures required, this service forwards the order to the subscriber upload mailbox. Otherwise, this service retains the order data in the database until all the required number of signatures is obtained.

The handleEBICSRequest business process receives a user's request. If the user's request contains the last segment of the order data, it invokes the EBICSOrderAuthorizationProcessing business process asynchronously to unpack the order data and generate the following files:

Note: Unpacking order data includes decoding, decrypting, and decompressing the order data.

- .DAT - Contains the unpacked order data in a user's upload mailbox
- .SIG - Contains the signature of the order data in a user's upload mailbox
- .PRM - Contains the order parameters in the user's upload mailbox
- .PSR - Contains a status report of asynchronous processing in the user's download mailbox

Processing Initialization

A user initiates a transaction by submitting the requests containing information about the incoming order. Based on this information, the EBICS Server verifies the order type, performs the message replay test, verifies message authentication, and checks user authorization before accepting the request.

After successful verification of the order data, the bank generates a transaction ID and includes the ID in its response to the user.

Processing Data Transfer

When more than one segment is required to transfer order data, the bank performs message authentication, verifies the transaction, verifies the segment number and size. After the EBICS Server receives the last segment of the order data, the complete order data is forwarded to the EBICSOrderAuthorizationProcessing business process asynchronously and the transaction ends.

The EBICSOrderAuthorizationProcessing business process unpacks the order data and forwards it to the user upload mailbox. The EBICSOrderAuthorizationProcessing business process generates post processing report (PSR) and routes it to the user's download mailbox. This business process also generates the .SIG and .PRM files to be forwarded to the user's upload mailbox. An .err file is generated when EBICSOrderAuthorizationProcessing business process encounters an error, for example, invalid electronic signature. Use the .err file to inspect an invalid order data file, if necessary.

Download From EBICS Server (FDL)

The download transaction comprises the following phases:

- Initialization
- Data Transfer
- Acknowledgement

A user submits the FDL order type to the bank. The user requests the download of the .PSR report to get the status of the FUL request. The user can also request to download valid file formats other than .PSR by using the FDL order type.

Processing Initialization

The bank verifies the message from the user. After the bank verifies the user's request, the bank collects the order data from the user's download mailbox based on the file format information in the request.

If more than one message matches the file format, the bank joins the contents of each message into a single order data and invokes the order data processor synchronously to pack the order data.

If the encoded form of the order data exceeds 1 MB, the order data is separated into segments. The first segment of the order data and the transaction ID is included in the response to the user.

Processing Data Transfer

The user sends the request for the next data segment. The bank authenticates the message, verifies the transaction, and the segment number and size.

In each transfer phase, the bank transfers all the segments until the last segment of the order data is included in its response to the user.

Processing Data Acknowledgement

After receiving the last segment of the order data from the bank, the user initiates the last phase, the acknowledgement request, to indicate that the data transfer has been successful.

If the bank receives a positive acknowledgement (receipt code=0) from the user, the bank moves the downloaded messages from the user download mailbox to the user archive mailbox. If the bank receives a negative acknowledgement from the user, the bank retains the downloaded messages in the user's download mailbox.

If a user wants to download valid file formats other than the .PSR reports from the user’s archive mailbox, the user must specify a date range in the EBICS request. The user must ensure that the date range matches the drop date of the .DAT file when moved from the user’s download mailbox to the user’s archive mailbox.

Segmentation and Recovery

The order data request (upload or download) cannot exceed 1 MB in compressed, encrypted, base64 encoded form. If the order data request exceeds 1 MB, the encoded form must be separated into segments. EBICS Banking Server Module is responsible for combining all these segments in order to reinstate the order data to its original form.

If an error occurs during the delivery of the order data segments, recovery can be performed. The user can download or upload the appropriate segment according to the recovery point sent in response by the server.

Recovery allows the transmission of an order to continue despite the occurrence of an error, without necessitating the retransmission of all order data segments that have been transmitted successfully.

A recovery point can be used to continue transactions from the transaction step that follows this recovery point in the transaction step sequence. Recovery points must be set during the recovery process:

- For upload transactions, the recovery point is the last transaction step wherein the bank has successfully received the request message and transmitted a response to the user. The recovery point is determined by the state of the transaction in the bank system.
- For download transactions, several recovery points may exist. All the previous transaction steps of the transaction wherein the bank has successfully received the request message and transmitted a response to the user.

Managing Keys

Managing Keys

You can insert, update, and retrieve certificates present in the Sterling Integrator repository. You can insert a base64-encoded certificate (public or private) and import and export certificates into the Sterling Integrator repository. You can also perform the following tasks in Sterling Integrator:

- Create a self-signed certificate with the key length 2048 for EBICS
- Manage CA certificates
- Store certificates, and manage the renewal and expiration of certificates
- Accept a public certificate of a user
- Validate the following subscriber keys using SHA256 as the hash algorithm:
  - Identification and Authentication Key Hash Value (in Hex format)
  - Encryption Key Hash Value (in Hex format)
  - Electronic Signature Key Hash Value (in Hex format)

Use the EBICS Export Certificate service to export the certificates present in Sterling Integrator to an external system. Use this service when you want to synchronize the certificates present in Sterling Integrator with an external database or system.

Use the EBICS Import Certificate service to add certificates from an external repository to Sterling Integrator. You can also delete the expired or invalid certificates.

Generating and Retrieving EBICS Reports

Generating and Retrieving EBICS Reports

Use the EBICS Reporting service to generate a payment status report (PSR) with every upload order (FUL) request. The .PSR report is in an XML format and follows the pain.002.001.02 schema. After the .PSR report is generated successfully, it is placed in the EBICS user's download mailbox.

A .PSR report is generated after asynchronous order processing of each FUL. A user can send an FDL request with the FileFormat pain.002.001.02.ara in order to retrieve the .PSR report. If no date range is specified in the EBICS request, the bank concatenates the PSR reports in the user’s download mailbox, and packages the order data in the EBICS response.

When the bank receives a positive acknowledgement from the user based on the parameter value provided under the FDLOrderParams element in the FDL request, the .PSR reports in the user’s download mailbox are moved to the user’s archive mailbox. If no positive acknowledgement is received after a specified time-out period, the EBICS Server Service scheduler changes the Extractable Count back to 1 for the .PSR reports in the user’s download mailbox, enabling the user to download the .PSR reports again.

If the user wants to download the .PSR reports from the user’s archive mailbox, the user must specify a date range in the EBICS request. The user must ensure that the date range matches the drop date of the .PSR reports when moved from the user’s download mailbox to the user’s archive mailbox.

Managing the EBICS Server

Managing the EBICS Server

The EBICS Server is implemented as a service in Sterling Integrator. The EBICS Server service is responsible for handling incoming EBICS requests (through HTTP and HTTPS) according to the EBICS protocol specifications, and generating and sending the appropriate response back to users.

The EBICS Server processes the generation and verification of electronic signature (ES), and identification and authentication of EBICS messages. It also interfaces with Subscription Manager to retrieve the profile information of banks, partners, users, and order types necessary for verification and authentication of messages and transactions. The processing flows (asynchronous and synchronous) of requests, such as, FUL and FDL, are also managed by the service. You can configure the service to update the EBICS repository and send event notifications to an external application during a synchronous transaction. Managing the message flow for the initialization and transfer phases of EBICS transactions is also one of the key responsibilities of the service. The lifecycle of the EBICS transactions in the bank system and the status of open transactions are managed by the EBICS Server, which also acts as an intermediate storage for transmitted order data segments and Electronic Signatures (ES).

When downloading bank-technical order data, the EBICS Server collects all the available order data in the user's mailbox, and concatenates them into a single document and sends the document to the order data processor to pack the document, that is, sign, compress, encrypt, and encode the document.

For information about configuring EBICS Server Service, see *EBICS Server Service* in the Sterling Integrator online documentation library.

Managing System Order

Managing System Order

System Order Manager works closely with Transaction Manager and Subscription Manager to update and query a user's key certificates and referential information, and to download bank parameters and bank certificates. It generates and retrieves XML order data based on the profile information.

System Order Manager also handles the implementation of upload and download system orders. The following table lists the supported upload system order types for EBICS transaction:

Upload System Order Type	Description
INI	Used in subscriber initialization. Sends the bank-technical public certificate of a customer to the EBICS Banking Server Module. The order data is compressed and base64-encoded.
HIA	Used to transmit user public certificates for identification, authentication and encryption within the framework of subscriber initialization. The order data is compressed and base64-encoded.
PUB	Used to update customer's certificates. Sends the bank-technical public certificate of the customer for updating the EBICS Banking Server Module. The order data is signed, compressed, encrypted, and base64-encoded.
HCA	Used to update customer's certificate. Sends the following certificates for updating the EBICS Banking Server Module: • Identification and authentication public certificate • Encryption public certificate The order data is signed, compressed, encrypted, and base64-encoded.
HCS	Used to update customer's certificate. Sends the following certificates for updating the EBICS Banking Server Module: • Bank-technical public certificate • Identification and authentication public certificate • Encryption public certificate

Upload System Order Type	Description
	The order data is signed, compressed, encrypted, and base64-encoded.
SPR	Used to suspend a user's access authorization. The order data is a blank character.

The following table lists the supported download system order types for EBICS transaction:

Download System Order Type	Description
HPB	Used to download bank public certificates from the EBICS Banking Server Module. The order data is compressed, encrypted, and base64-encoded. The response message and the order data are not signed.
HPD	Used to download bank parameters from the EBICS Banking Server Module. The order data is signed, compressed, encrypted, and base64-encoded.
HEV	Used to download information on supported EBICS versions. The order data is compressed and base64-encoded.

Processing Order Data

Processing Order Data

To ensure secure transfer of order data, the order data must be packed. Packing of order data includes signing, compression, encryption, and base64 encoding depending on the requirement of the order type. The receiver must unpack the order data to view the attributes. Unpacking of order data includes verification, decompression, decryption, and base64 decoding depending on the requirement of the order type.

The Order Data Processor is responsible for packing and unpacking the order data. It interfaces with the Subscription Manager and Transaction Manager to retrieve the relevant information required for packing and unpacking the order data. For example, the profile information may include the transaction ID, the direction of the flow (upload or download), response type (synchronous or asynchronous), type of processes required, object ID of the encrypted key, and object ID of the Electronic Signature (ES). EBICS Order Processing service performs EBICS transactions and user retrieval, and packing and unpacking of encrypted symmetric keys. Based on the profile information that is retrieved, the EBICS Order Processing service determines if packing or unpacking of the order data is required, and invokes the appropriate packing or unpacking service.

Authorized Order Manager is responsible for initiating the Order Data Processor to unpack the order data received from the FUL order type request, routing the unpacked order data to the backend subscriber's upload mailbox, and renaming it according to a defined naming convention.

Apart from the EBICS Order Processing service, the following services are available in Sterling Integrator to process order data:

- The EBICS Order Authorization service handles incoming order requests for the bank-technical upload order type (FUL). If an order has fulfilled the number of signatures required, this service will forward the order to the subscriber upload mailbox. Otherwise, this service forwards the order to the pending order mailbox.
- The EBICS Order Streaming service packs and unpacks order type data using the pipeline functionality in Sterling Integrator.
- The EBICS ES Packaging service either packs or unpacks key information that is used when signing and verifying the ES.
- The EBICS Compression service performs compression and decompression of order data using zlib in pipeline mode.
- The EBICS Encryption service performs encryption and decryption of order data using the AES-128 algorithm in pipeline mode. E002 encryption algorithm is supported.

- The EBICS Encoding service performs encoding and decoding of order data using the base64 method in pipeline mode.
- The EBICS Signing service performs the signing and verification of order data on the SHA-256 digest computed in pipeline mode. A005 and A006 signing algorithm is supported.

Order data must be unpacked for upload transactions, and packed for download transactions.

The packing process involves the following sequence. However, based on the order type, one or more of the following processes may not be required:

1. Signing
2. Compressing
3. Encrypting
4. Base64 encoding

The following example illustrates encryption of an order type. A business process invokes the Encryption service. If the order data has been signed, the business process passes the symmetric key to the Encryption service. If the order data has not been signed, the Encryption service generates and returns the symmetric key to the business process. If the symmetric key was created, the business process invokes the EBICS Order Processing service with the output message type set to setEncryptedKey.

The unpacking process involves the following sequence. However, based on the order type, one or more of the following processes may not be required:

1. Base64 decoding
2. Decrypting
3. Decompressing
4. Verifying the signature

The following example illustrates decryption of an order type. A business process invokes the EBICS Order Processing service with the output message type set to getEncryptedKey. The base64-encoded secret key is retrieved and set in the process data for use by the Encryption service.

Integrating with Sterling File Gateway

Integrating with Sterling File Gateway

Sterling File Gateway enables secure file transfer between internal and external partners using the same or different communication protocols, file naming conventions, and file formats. Sterling File Gateway supports EBICS for movement of large and high-volume file transfers, with end-to-end visibility of file movement in a process-oriented and highly-scalable framework that alleviates file transfer challenges, such as protocol and file brokering, automation, and data security.

Files move between the EBICS server and Sterling File Gateway through shared mailboxes and partners. The Subscription Manager creates mailboxes in the structure of User/Partner/Inbox during partner creation.

Sterling File Gateway uses Provisioning Facts as part of the Routing Channel Template definition. Routing channel templates used in EBICS scenarios must include the configuration of provisioning facts. Routing channels using the templates must include the specification of values for provisioning facts.

For inbound scenarios, the EBICS Order Data Processor (ODP) takes an EBICS order file upload (FUL) from an EBICS client to an EBICS Server, unpacks the payload and deposits into a User/Partner/Inbox mailbox structure. Sterling File Gateway is configured to route from that mailbox for downstream processing and ultimate delivery to a consumer.

In the outbound scenario, Sterling File Gateway is configured to deposit a message in a consumer mailbox, which will be routed and stored in User/Partner/Outbox. On an EBICS order file download (FDL) from an EBICS client to an EBICS Server, the EBICS Order Data Processor (ODP) packages the message and makes it available to the client.

Sterling File Gateway enables operators to search for transactions and view details of routes and deliveries.

Certain procedures are necessary to initiate integration with Sterling File Gateway. For more information about integrating with Sterling File Gateway, refer to the *Sterling File Gateway* - *EBICS Integration Guide*.