

# Security tenets of iBank 2 UA system

Version 2.0.15

# Оглавление

<b>1 Main theses</b>	<b>2</b>
Introduction . . . . .	2
Essences and definitions . . . . .	3
Conception and architecture of iBank 2 UA system . . . . .	5
Realization and parameters of utilized cryptographic algorithms . . . . .	6
IP-security . . . . .	6
<b>2 The client's connection to the bank</b>	<b>7</b>
Bank's auxiliary Web-server . . . . .	8
SoftUpdate and developer's Electronic Digital Signature under Java applets . . . . .	8
<b>3 Secured interaction between Java applet and Application Server</b>	<b>10</b>
Applied Protocol . . . . .	10
Secured Network Protocol . . . . .	12
Permanent and session keys . . . . .	13
<b>4 Security tenets upon working with documents</b>	<b>15</b>
Transition of the document from the client to the bank . . . . .	15
Uploading of the document from iBank 2 UA system to the bank's accounting system	16
Changing document status by bank employee . . . . .	16
Receiving by the client of the bank report . . . . .	17
Verification of the client's public key certificate validity . . . . .	17
Event and error logs . . . . .	17

# Глава 1

## Main theses

### Introduction

iBank 2 UA E-Banking system is designed for granting e-banking services by banks to corporate and private clients with the support of all access channels – Internet (online and offline), WAP, SMS, phone (voice, fax, modem). iBank 2 UA system (bundled software) is installed in bank and allows to serve corporate and private clients. The system is successfully used by first-rate multidivisional banks of Ukraine.

iBank 2 UA belongs to secured documents exchange systems. Documents exchange takes place between the client and the bank. Electronic document, sent by the client and received by the bank, is the reason for financial operations carried out by the bank.

The final goal of all attacks on the system is one of the following:

- Electronic document's substitution (obtrusion) for one of the sides by the malefactor;
- Document's confidentiality violation (document's examination).

Authenticity (prove of authorship) and document integrity are provided by mechanism of Electronic Digital Signature under financial documents. Electronic document signed by Electronic Digital Signature is the reason for financial operations fulfillment and the demonstrative base in conflicts settlement. Algorithms are created according to GOST 34.310-95 (EDS) and GOST 34.311-95 (hash function).

Confidentiality is provided by mechanism of data encryption. All digital documents transmitted through Internet in cipher form. Encryption is created according to GOST 28147-89.

## Essences and definitions

The following essences and definitions are used in iBank 2 UA system:

- **iBank 2 UA Application Server** — server-based software of iBank 2 UA e-banking system, that is installed and used in bank. Application Server (AS) provides service of corporate and private clients.
- **Gateway** — software for iBank 2 UA system integration with bank external accounting systems
- **Document**. In iBank 2 UA system document is any financial document in digital form. Financial document is client's order of funds management in bank. Digital document must be signed by one or several Electronic Digital Signatures. Digital document with client's EDS has legal validity equally traditional paper financial document.
- **Report**. Any standard for iBank 2 UA system report, formed by server on client's demand.
- **Applied Request**. Applied Request is formed by client's software one of the iBank 2 UA modules or additional services. Applied Request is transmitted to bank iBank 2 UA AS for execution. Request contains client's key ID. Before request processing server verifies client's rights.
- **Applied Reply**. Applied Reply is formed by bank AS. Applied Reply is directed to the client's software and contains result of client's request processing.
- **Ordering information** — additional information, that is transmitted in client's request or returnable by server reply.
- **Information flow** — aggregate of all applied requests and applied replies transfers between client's software and iBank 2 UA AS. Information flow is always enciphered.
- **Encryption** — information cryptographic transformation. In the context of secured documents circulation — whole information flow encryption with the purpose to exclude the access of third party to the transferred information.
- **EDS**. Electronic Digital Signature is a result of document cryptographic processing with the use of private EDS key. EDS is used for authenticity guarantee of digital document.
- **Private EDS key** — secret order with the length of 32 bytes. The owner of that secret order is key EDS owner only. The private EDS key is kept by its owner in secret and is used for EDS forming under client's document. The private EDS key is also used for user's authentication.
- **Public EDS key** — secret order with the length of 128 bytes, that depends on EDS validity verification under digital document.
- **Public EDS key Certificate** — information, that includes public EDS key, parameters, terms and scope of public EDS key and information about public EDS key owner. That Certificate is signed by representative bank employee.
- **Paper public EDS key Certificate** — paper document that contains public EDS key in hexadecimal format, parameters, terms and scope of public EDS key and information about public EDS key owner. Paper public EDS key Certificate is attested by owner's signature and company stamp (for corporate clients) at presence of representative bank employee.

- **Client.** iBank 2 UA system supports work with two types of clients:
  1. Corporate clients – artificial persons (organizations) or private entrepreneurs;
  2. Private clients – natural persons, that don't occupy with business activity.
- **Client's software** – iBank 2 UA software, that is used by client for working with financial documents, reports etc. Client's software provides encryption of information flow during exchange with bank AS and also it provides work with client's EDS. Clients modules are following (AWM – automated working module):
  - AWM **Registrar** – Java applet that is downloaded to the client's Web-browser;
  - AWM **Internet-Banking** – Java applet that is downloaded to the client's Web-browser;
  - AWM **PC-Banking** – Java application installed on client's PC;
  - AWM **Mobile-Banking** – .Net application installed on client's PC;
  - AWM **Ticker** – Java application installed on client's PC;
  - AWM **SMS-Banking** – Internet-Banking Java applet option;
  - AWM **WAP-Banking** – special web-pages;
  - AWM **Phone-Banking** – e-banking service via phone line;
  - AWM **Web-Banking** – special web-pages;
  - AWM **Corporate autoclient** – application installed on client's PC.
- **Bank's AWM.** iBank 2 UA system contains four tailored AWMs for bank employees:
  - AWM **System administrator**;
  - AWM **Bank administrator**;
  - AWM **Operationist**;
  - AWM **Bank employee's registrar**.

## Conception and architecture of iBank 2 UA system

iBank 2 UA system has been developed in accordance with «thin client» conception — user doesn't need to install any dedicated software. To work with the system client needs any Web-browser with built-in virtual Java-machine (JVM), that corresponds to JDK 1.1 and higher specification.

Client may use one of the following Web-browsers:

- Microsoft Internet Explorer version 4.0 and higher with built-in Microsoft JVM version 3272 and higher;
- Microsoft Internet Explorer version 4.0 and higher with Sun Java Plugin;
- Netscape version 6.0 and higher;
- Mozilla version 1.0 and higher;
- Opera version 5.0 and higher with Sun Java Plugin.

Secured work is provided by regular security services of the Web-browser, Java virtual machine and additional iBank 2 UA system built-in security mechanisms.

iBank 2 UA system developed in classic trimeric architecture. Downloaded into client's Web-browser Java applet is used as representative component. During working process applet interacts with bank iBank 2 UA AS, where is performed whole business logic. In the same time AS interacts with iBank 2 UA DBS (Oracle, MS SQL or PostgreSQL), where whole information is kept: clients' documents, references, clients' settings etc.

Client's work in **Internet-Banking** occurs in two steps. On the first step client connects to the bank's site and loads Java applet into Web-browser. Second step includes client's work in Java applet and Java applet's interaction with bank's Application Server via secured connection.

Work in **PC-Banking** and **Mobile-Banking** starts from the second step, i.e. Java application's interaction with bank's Application Server.

Confidentiality in **Internet-Banking**, **PC-Banking**, **Mobile-Banking** and additional iBank 2 UA system services is provided by data encryption mechanism. Transferred data encryption and integrity control, sides' cryptographic authentication are carried out during interaction via Internet. Algorithms are created according to GOST 28147-89.

Confidentiality in **Web-Banking** is provided by SSL standard cryptographic protocol, that is built-in all modern Web-browsers. Transferred data encryption and integrity control, bank's cryptographic authentication by client are carried out during interaction via Internet.

**WAP-Banking** provides data encryption and bank's authentication via WTLS and SSL standard cryptographic protocols. Client's authentication is carried out by ID (login) and password.

**SMS-Banking** provides data protection via standard cellular communication means. Client's authentication is carried out by cellular phone number, ID and password.

**Phone-Banking** doesn't allow voice and fax messages encryption. Client's authentication is carried out by ID and password entered in tonal mode.

**WAP-Banking's**, **Web-Banking's** and **Phone-Banking's** private clients can carry out operations dealing with funds transfer from accounts and cards using sign manual analogues in the form of individual tables of disposable passwords or hardware authentication means like OTP (One-Time Password) tokens created by Alladin Knowledge Systems or VASCO Data Security International.

## Realization and parameters of utilized cryptographic algorithms

iBank 2 UA has built-in Java cryptographic library «Standard-Java», that is developed by «SITaS» company. Java cryptographic library «Standard-Java» is certified by DSTSIP of SSU (Department for Special Telecommunication Systems and Information Protection of Security Service of Ukraine). Certificate of conformance is № UA1.112.0135242-06 of 11 September 2006.

«Standard-Java» crypto-library is built in and distributed with iBank 2 UA system according to the Contract number 1 dated 11 September 2006 «On transfer of PC program «Software product» «Standard-Java». Cryptographic transformation classes» between «BIFIT» Ltd and «SITaS» company.

Crypto-library is developed on Java and consists of compact Java classes set, that are built in clients' Java applets and clients' server Java applications. Crypto-library is designed to provide protection of confidential information not being State secret from confidentiality and integrity violation by means of using cryptographic procedures built in applied programs. Crypto-library provides all work modes for GOST 28147-89 (encryption), GOST 34.310-95 (Electronic Digital Signature), GOST 34.311-95 (hash function).

iBank 2 UA system is always provided with built-in «Standard-Java» crypto-library. There is no need to separately purchase crypto-library license for each client or each workplace. «BIFIT» Ltd possess required DSTSIP of SSU license for providing services in the field of secured systems developed, technical support and distribution of encryption tools.

## IP-security

iBank 2 UA system introduction is always followed by modification of bank's current IP security policy. iBank 2 UA system servers are located in separated network segment with access from Internet and internal bank's network controlled by network screen. iBank 2 UA system introduction should be followed by the thorough tuning of iBank 2 UA system servers' OS. Support of the unused protocols and network services should be excluded. iBank 2 UA system servers should have network access to file system forbidden. OS built-in audit mechanisms should be enabled. Correctly designed and thoroughly introduced IP security policy and constant IP monitoring can provide assured level of iBank 2 UA system and internal bank's network security.

## Глава 2

# The client's connection to the bank

Client's work starts from connection to bank's Web-server. Client enters full bank's Web-server URL including protocol's type HTTPS in Web-browser, e.g. <https://ibank.bankname.com>.

HTML pages, XML configuration files, Java applets and other data loaded in Web-browser are open. Client doesn't enter authentication identifiers and passwords, doesn't load private parameters, so there is no need to encrypt traffic on the step of Internet-Banking connection. Malefactor's attacks on the stage of connection can have the following goals:

- Bank's Web-server substitution
- Starting HTML pages loaded in client's Web-browser modification
- XML configuration files loaded in client's Web-browser modification
- Java applets loaded in client's Web-browser modification

The final goal of all attacks is the substitution of client's loaded Java applet with the following client's private Electronic Digital Signature key theft.

To prevent the listed attacks on the connection stage SSL protocol built in Web-browser is employed. It provides the following:

- Assured client's connection to bank's Web-server
- Integrity of data (HTML pages, XML configuration files, Java applets etc.) loaded by the client from bank's Web server

While working via SSL, client's Web-browser provides bank's Web-server authentication by comparing domain name in client entered URL with domain name listed in certificate. SSL also provides integrity of data, loaded in Web-browser. Presence or absence of export limitations of SSL cryptographic protocol realization in early versions of Web-browsers doesn't affect the security level. Export limitations existing in Web-browsers' early versions affected data encryption algorithms and session keys length exclusively. They didn't affect Web-browser's authentication and transferred data integrity provision mechanisms.

## Bank's auxiliary Web-server

Bank's Web-server reliability and security should meet very high requirements. iBank 2 UA Application Server has the built-in auxiliary secured Web-server. The Web-server has necessary functional limitations to make impossible even potential attacks aimed on HTML pages, Java applets and other resources modification. Web-server uses Sun Microsystems realization of SSL cryptographic protocol.

iBank 2 UA system is provided with utilities for auxiliary Web-server's private and public keys (SSL) generation, request creation for X.509 Certificate receiving, Certification Center provided certificate import.

X.509 Certificate for auxiliary Web-server should be received from one of the world's Certification Centers - VeriSign, Thawte etc.

## SoftUpdate and developer's Electronic Digital Signature under Java applets

All modern Web-browsers have built-in mechanism for Java applets loading acceleration in case of user's repeated connection. In Microsoft Internet Explorer this mechanism is named SoftUpdate, in Netscape this mechanism is named SmartUpdate. Loading acceleration mechanisms (further called SoftUpdate for the sake of brevity) may differ in details but are designed for the common task. SoftUpdate mechanism is used by including several JavaScript code lines in the HTML pages.

During the very first connection Web-browser loads Java applet from the Web-server in form of CAB archive for MS Internet Explorer or JAR archive for other browsers. The archive is saved on user's local disk in one of the Web-browser's subdirectories.

On the following repeated connections Web-browser compares previously loaded version, saved in the subdirectory, with the current version on the Web-server. If the versions coincide with each other, previously loaded version of Java applet is used. If Web-server has the newer version of Java applet, Web-browser automatically loads, saves in subdirectory and then uses the newer version.

SoftUpdate mechanism is used in iBank 2 UA system by default. System also includes starting HTML pages designed for loading Java applets without using SoftUpdate.

All client loaded Java applets in iBank 2 UA system are signed by developer's Electronic Digital Signature. Standard mechanism for developer's Electronic Digital Signature check is built in all modern Web-browsers.

Developer's («BIFIT»Ltd) Electronic Digital Signature check is done based upon X.509 Certificate issued by world's Certification Center Thawte Consulting. World's Certification Center's root certificates are included in all types and versions of Web-browsers' distributives.

Developer's Electronic Digital Signature is used for the following:

- Provision of integrity and authenticity for Java applets loaded and stored in Web-browsers subdirectory while using SoftUpdate mechanism (protection from attacks on modification of files with Java applets stored on client's PC)

- Provision of extended privileges for Java applet (client's local disks operation for access to the client's private Electronic Digital Signature keys storage, printing, interaction with hosts having IP address different from auxiliary Web-server IP address) by Web-browser Java virtual machine.

## Глава 3

# Secured interaction between Java applet and Application Server

The initiator of secured interaction between bank's Application Server and client's Java applet is always the client. When the need to send the document to the bank or to receive information arises, client's Java applet carries out secured interaction with bank's Application Server.

Secured interaction can be divided into two layers, upper layer being called Applied Protocol and lower layer being called Secured Network Protocol.

Layer	Function
Applied Protocol	Client's authentication Electronic Digital Signature mechanism Applied requests and replies
Secured Network Protocol	Data encryption Data integrity provision Bank's authentication

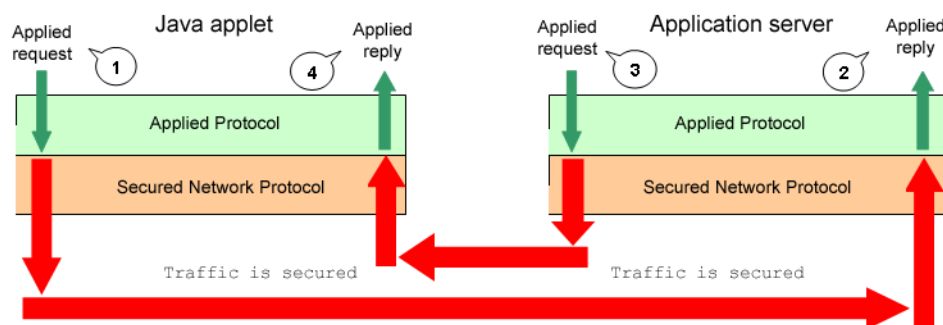


Рис. 3.1. Secured interaction between Java applet and Application Server

### Applied Protocol

Applied Protocol provides mechanisms for user's authentication and Electronic Digital Signature. Applied Protocol is formed in the following way:

CONNECTION ESTABLISHING  $\implies$  REQUEST  $\implies$  REPLY  $\implies$  CONNECTION CLOSING

Transaction occurs in the following way:

1. Java applet opens the connection with the Application Server.
2. Java applet creates applied request and sends it to the Application Server.
3. Application Server receives and processes applied request.
4. Application Server creates applied reply and sends it to the Java applet.
5. Java applet receives and processes applied reply.
6. Java applet closes the connection with the Application Server.

Applied request consists of the header and the data area. In the Applied request header following parameters are passed:

- Applied request code;
- Public customer key ID;
- Session password of 32 bytes in length;
- Time marker;
- Length of data area.

Data area is used to send applied request's parameters values. Client's Electronic Digital Signature is represented in data area by the same parameter as document's number, document's date etc.

Applied reply also consists of the header and the data area. The header contains applied reply error code and data area length. Data area contains applied reply parameters values.

Session password is generated by Java applet during initialization level (generation is carried out by SecureRandom – cryptographic generator of pseudo-random numbers) and installed during first Java applet access to bank AS with the use of client's EDS under session password. Password length (32 bytes) provides  $2^{256}$  combinations. Probability of session password matching is  $\sim 10^{-78}$ .

Session password life time defined in AS settings (30 minutes by default). After life time ending AS requires installation of session password again with use of client's EDS.

Applied Protocol provides two-phase client's authentication with Electronic Digital Signature mechanism and sessional password. It allows to provide assured security and to significantly lower the load on Application Server by excluding Electronic Digital Signature check under each applied request.

Applied requests with client's Electronic Digital Signature are not used for conflicts settlement and are not saved (journalized) on bank's side. The only necessary material for conflicts settlement is client's Electronic Digital Signature signed documents.

Applied Protocol uses following cryptographic algorithms:

- GOST 34.310-95 — procedure of forming and verification of Electronic Digital Signature. It has three mounted rigidly open parameters:  $a$ ,  $q$ ,  $p$ . Customer EDS keys parameters  $a$  and  $p$  have size of 1024 bits;

- GOST 34.311-95 — procedure of hash function calculation. Replacement tables are fixed, those tables were taken from GOST test case.

Applied Protocol work requires customer key pair (the private and the public one) . The private customer key is kept cipher with the password file. The public customer key is kept in bank in system iBank 2 UA DBS. Paper certificate of the public customer key is also kept in bank.

## Secured Network Protocol

Secured Network Protocol carries out the following functions:

- Provides encryption of the data transferred between Java applet and Application Server
- Provides integrity of the data transferred between Java applet and Application Server
- Provides authentication of Application Server by Java applet

Secured Network Protocol is a modified SSL v.3 protocol with simplified session keys co-ordination and predefined cryptographic algorithms, work modes, key lengths and other parameters.

Secured Network Protocol allows client to work via HTTP Proxy-server (MS Proxy, WinGate, Win Proxy, Squid etc.) if required. The same tunnelling mechanism as in SSL protocol is used. Modes allowing working via Proxy-server with or without basic authentication are supported.

When it's needed Secured Network Protocol can be replaced by any other secured protocol, that's working over TCP. But in this case some changes should be made.

The main advantages of Secured Network Protocol are extremely low size of system data transferred while coordinating session keys and low load on bank's server CPU during secured interaction between Java applets or Java applications and bank's Application Server.

RSA algorithm with the key length of 1024 bits is used in procedure of session keys agreement. All transferred data is encrypted with session keys according to GOST 28147-89. Data integrity is also provided by GOST 28147-89 (authentication code).

Secured Network Protocol uses following cryptographic algorithms:

- RSA-1024 — asymmetric cryptographic algorithm with  $N$  module (length 1024 bits).  
That algorithm is used at procedure of session encryption keys and session originality control agreement. Java applet encrypts information on bank's public encryption key. Application Server decrypts information on bank's private encryption key.

- GOST 28147-89 — symmetric cryptoalgorithm.

That algorithm is used by Java applets and Application Server for encryption of transferred data.

Secured Network Protocol work requires bank encryption keys pair.

Bank encryption keys pair is kept by Application Server. Private key is used by Application Server during interaction with Java applets. Public key is downloaded to the client from auxiliary Web-server ( `public_gs1` file) through browser's HTTPS and is used by Java applet as one of secured interaction parameters.

## Permanent and session keys

All keys used in iBank 2 UA system divided into two main groups: session keys and permanent keys.

Among session keys are:

- Session keys of data encryption. Cryptoalgorithm — GOST 28147-89;
- Session keys of authentication code forming. Algorithm — GOST 28147-89.

Among permanent keys are:

- Private and public clients' EDS keys. EDS algorithm — GOST 34.310-95;
- Private and public operationist and bank administrators' EDS keys. EDS algorithm — GOST 34.310-95;
- Private and public encryption keys of bank. Cryptoalgorithm — RSA with 1024 bits module. Used in session keys agreement level.

**Session keys of data encryption and authentication code forming** of 32 bytes size. Those keys are generated by Java applet for every new transaction.

SecureRandom cryptographic generator of pseudo-random numbers is used for generation of session keys. Bytes array (bytes are taken from SeedGenerator program transmitter of pseudo-random numbers) is used as a starting vector.

**Clients' EDS keys.** Client may have any number of EDS keys. Several EDS keys can be active at the same time. Grouping of EDS keys is provided: 1-st signature, 2-nd signature etc. Up to 8 groups can be adjusted by bank's administrator to the client.

Client generates EDS keys with the use of Java applet **Registrar**. SecureRandom cryptographic generator of pseudo-random numbers is used for generation of EDS keys. Bytes array (bytes are taken from SeedGenerator program transmitter of pseudo-random numbers) is used as a starting vector and also bytes array formed by biomedical sensor of random numbers (that array picks randomness from 1024 coordinate dimensions of mouse pointer and current time in milliseconds at the events rise moments, generated by mouse pointer movements).

After client's EDS keys generation, the public EDS key is transferred from Java applet **Registrar** through secured connection to AS of bank and there public EDS key is preliminarily registered. AS returns preliminarily registered EDS public key ID to Java applet **Registrar** through secured connection.

When the password was entered at Java applet **Registrar** the client's private EDS key is enciphered (GOST 28147-89) by hash function (GOST 34.311-95) from password entered by the client and key ID. After that client's private EDS key is saved on floppy disk in repository file. Each client's private EDS key has its own name (named by client) for future work with the key. In bank client signs a Operation Agreement and attests paper Certificate of public EDS key.

All public EDS keys are kept in iBank 2 UA system DBS of bank in clients' EDS keys Certificates form.

Client's private EDS key is used for clients' authentication and for client's EDS forming under financial documents and other client's outgoing orders.

Client's public EDS key is used by bank for client's authentication and for client's EDS verifying under financial document. Client's EDS verifying is carried out by AS when client signs document. Client's EDS verifying is also carried out by Gateway during documents unloading to automated bank system (ABS)

**EDS keys of operationists and bank administrators.** Bank operationist and administrator may have any number of EDS keys. Generation of EDS keys of bank employees is carried out by Java applet **Bank employee's registrar**.

Private EDS keys of bank employees are kept on floppy disk in repository file. Public EDS keys of bank employees are kept on iBank 2 UA DBS.

**Bank encryption keys** are generated by bank administrator and can be changed by bank at any time (it requires AS restart).

iBank 2 UA system has programme that generates bank encryption keys and saves keys in file `%ibank_home%\conf\gslkeystore` (`%ibank_home%` – directory where AS installed). Private bank encryption key is used by AS during interaction with Java applets.

Public bank encryption key is kept in file `%ibank_home%\webapps\ROOT\public_gsl` on bank server. That key is downloaded by Java applet during initialization through browser HTTPS from bank auxiliary web-server and is used for interaction with bank AS.

## Глава 4

# Security tenets upon working with documents

### Transition of the document from the client to the bank

Client (corporate or private) can create, edit and save documents with the help of client's AWMs. When saving a document, the following issues are verified:

- Presents of client's rights on work with given document type;
- Conformity of client's Essential Elements in document to current Essential Elements;
- Accounts belonging to the client;
- Accuracy of document field filling.

A document is signed on client's side by cryptographic library that is built in client's AWM. Document signing is available in all client's AWMs except **Web-Banking** and **WAP-Banking** (those AWMs have no EDS mechanism of financial documents). During signing document is presented in xml format. All fields of the document presented in XML-description of given document type are signed. Time of bank AS is used during client's EDS forming. All client's AWMs are developed on Java except **Mobile-Banking**, **Web-Banking** и **WAP-Banking**. EDS is formed by client's AWM. It's almost impossible to substitute signed data in virtual Java-machine memory.

**Mobile-Banking** is developed on C# and requires presence of Microsoft .NET Compact Framework on pocket computer. Virtual Java-machine is presented at Pocket PC 2003 and Windows Mobile 5.0.

If it is required several EDS under the document for bank examination (for corporate clients), employees of corporate client (owners of corresponding EDS) sign document with the use of AWMs. When signing a document, the following issues are verified:

- Client's certificate of public EDS key belonging to signing client;
- Validity of client's public EDS certificate;
- Presents of rights on signing of given document type in certificate of client's public EDS key;
- Absence of other signatures of the same group under the document;
- Presents of client's rights on work with given document type;

- Conformity of client's Essential Elements in document to current Essential Elements;
- Accounts belonging to the client.

When a document gathers needed number of signatures, it is whether unloaded to corresponding bank account system or got to examination of bank employee.

## Uploading of the document from iBank 2 UA system to the bank's accounting system

Initial document with client's EDS is always kept on iBank 2 UA system DBS. Document unloading to bank account system is carries out by Gateway. A document gets **New** status when it's created and saved. Until document has no needed number of signatures it has **Signed** status (client can get info about the date and person who signed document). When document has all needed signatures it has **Delivered** status. After that the Gateway starts work on the bank side.

During document unloading the Gateway verifies:

- Document belonging to the given client;
- Presents of client's rights on work with given document type;
- Conformity of client's Essential Elements in document to current Essential Elements;
- Accounts belonging to the client;
- Presents of client's rights on work with accounts in document;
- Certificate of client's public EDS key belonging to the given client;
- Validity of client's public EDS certificate;
- Correctness of all EDS under the document;
- Conformity of EDS number and groups under the document to those stated in client's rights on work with given document type.

If all verifications were successful, the Gateway formats and unloads document into bank account system in coordinated format in that transaction. Additional mechanisms of information security can be built in the Gateway at the instance of bank.

In practice almost all integrations of iBank 2 UA system to the Gateways has external encryption information security methods: starting with certified DSTSIP to public CSP.

It should be taken into account upon concordance with the bank about unloading documents from iBank 2 UA system into bank account systems that forming of EDS by the Gateway is a resource-intensive process. At the same time EDS under the document will be verified in bank account system and that produces high load on server of account system. Great flow of documents (50-70 documents per second) produces critical load for the Gateway and account system.

## Changing document status by bank employee

Bank employee in AWM (Operationist) examines and prints documents, adds comments. Operationist can review each document (with the purpose of correctness of document filling examination) and verify EDS correctness. After document has been verified bank employee is available to change status of the document.

## Receiving by the client of the bank report

Bank sends reports and letters to the client. Among reports are:

- Account statements for any period;
- Turnover balance sheet for any period;
- Budgeting reports;
- Exchange rates.

Reports are formed dynamically on the base of accounting transaction information in iBank 2 UA system DBS. In standard version dynamically formed reports are not signed by bank EDS, but transferred with the use of cryptographic information security. In custom versions it is possible to use bank EDS when transferring documents from bank to clients.

## Verification of the client's public key certificate validity

Verification of the client's public key certificate validity is carried out every time before certificate using for EDS verifying under document/applied request. Verification of the client's public key certificate validity is carried out by two services:

- iBank 2 UA AS when documents are being signed by clients and during client's authentication;
- The Gateway upon unloading client's documents into bank account system.

Before using of public EDS key certificate following issues are verified:

- Certificate belonging to subject;
- Certificate validity;
- Certificate current status;
- Certificate area of application;
- Public EDS key correctness;
- User's rights, who signed certificate;
- Developer EDS under certificate.

## Event and error logs

iBank 2 UA system has built-in client's EDS mechanism under financial documents for conflict resolution. The necessary material for conflicts settlement is client's Electronic Digital Signature signed documents, that are kept in bank, on iBank 2 UA DBS.

Logging mechanism is designed for full recovery of clients operations and events in iBank 2 UA system. Event and error logs are not the demonstrative base in conflicts settlement, but they help to recover course of events.

Application Server saves individual files for each client's AWM in %iBank\_home%\logs subdirectory. Errors and events are noticed in separate files. Hence the name of logs formed in the following way: **<user's AWM service name > + accentuation sign " \_ " + event** (for event logs) or **error** (for error logs).

iBank 2 UA has following event and error logs:

№	File name	Description
1	access.log	Access to iBank 2 UA built-in Web-server
2	activemq.log	ActiveMQ module (message posting by SMS – Banking)
3	admin_event(error).log	<b>Bank administrator</b> module <sup>1</sup>
4	autoclient_event(error).log	<b>Corporate autoclient</b> module
5	dbcreator.log	Utility for SQL-scripts creation for chosen DBS type, that form iBank 2 UA DB
6	handy_event(error).log	<b>Mobile — Banking for corporate clients</b> module
7	ibank_event(error).log	<b>Internet — Banking for corporate clients</b> module
8	ipfilter_event.log	IP-filtering mechanism
9	ip_filter_filling.log	IP-filter table loading into DB
10	isida_event(error).log	<b>Operationist</b> for corporate clients module
11	jerry.log	Application Server
12	load_res.log	Software resources loading for users' applets in iBank 2 UA DB
13	load_struct_corporate.log	Structured payments loading for corporate clients
14	load_struct_private.log	Structured payments loading for private clients
15	make_dist.log	Forming of client's distributives (PC — Banking, Corporate Autoclient)
16	make_dist_ticker.log	Ticker for corporate clients distributive forming
17	mfo.log	MFO reference import into iBank 2 UA DB
18	msinker_event(error).log	<b>PC — Banking. Financial Control Centre</b> module
19	multiclient_event(error).log	<b>Internet – Banking. Financial Control Centre</b> module
20	pegasus_ event(error).log	<b>SMS – Banking</b> module
21	pibank_event(error).log	<b>Internet — Banking for private clients</b> module
22	pisida_event(error).log	<b>Operationist</b> of private clients module
23	psinker_event(error).log	<b>PC — Banking for private clients</b> module
24	rates.log	Exchange rates import and bank rate of conversion in iBank 2 UA DB
25	registry_event(error).log	<b>Registrar</b> module
26	sinker_event(error).log	<b>PC — Banking for corporate clients</b> module
27	sirena_event(error).log	<b>Phone — Banking</b> module
28	swift.log	SWIFT reference import in iBank 2 UA DB
29	ticker_event(error).log	<b>Ticker for corporate clients</b> module
30	wap_event(error).log	<b>WAP — Banking for corporate clients</b> module and <b>WAP — Banking for private clients</b> module

<sup>1</sup>AWM System administrator types messages to console only

Those logs don't contain information about errors, appeared at client's side. Logs of client's side noted by applications installed on client's side.