CERTIFICATE

HUNGUARD Informatics and IT R&D and General Service Provider Ltd. as a certification authority assigned by the assignment document No. 002/2006 of the Minister of the Ministry of Informatics and Communication (MIC) of the Republic of Hungary
based on the MIC Decree 9/2005 (VII.21.)

certifies

that the

"A2-Polysys CryptoSigno Interop JAVA API for qualified electronic signature" signature creation/verification application development kit

version 2.0.0
developed and distributed by

polysys®

electronic signature product

with functionality written down in Annex 1
and with conditions regarding the IT environment and secure usage listed in Annex 2

passes the requirements

for the development of secure applications meeting the applicable standards for generating and verifying qualified electronic signatures according to the Act XXXV of 2001 of the Republic of Hungary on electronic signature.

This certificate has been issued on the basis of the certification report No HUNG-TJ-29-2006.
Produced on commission of Polysys Ltd.
Certificate registration number: **HUNG-T-29/2006.**
Date of the certificate: 23.02.2006.
Validity period of the certificate: 23.02.2009.
Annexes: attributes, conditions, requirements and other features on eleven pages.

L.S.

Endrődi Zsolt
certification director

dr. Szabó István
managing director

Annex 1

Overview of the main features of A2-Polysys CryptoSigno Interop JAVA API for qualified electronic signatures v2.0.0

A2-Polysys CryptoSigno Interop JAVA API is a platform independent development kit (library) for creating qualified electronic signatures v2.0.0 ("**A2-API**") that supports the following functions for relying applications made by Java technology:

- generation and verification of qualified or advanced electronic signature;
- encryption and decryption;
- building and validation of certification path;
- checking of certificate revocation lists;
- identification, authentication and authorisation control;

in order that applications are able to provide effective and standard PKI services.

The A2-API has two distinct operational modes:

- strict mode, which is suitable for generation of qualified electronic signatures,
- normal operational mode, which is applicable for generation of advanced electronic signatures.

Both modes demand the following software configuration requirements:

- Operation system: Linux, Solaris, Unix, Java Desktop System, Windows,
- JRE 1.5 Java Runtime Environment or JRE1.4.2 and (if SSCD or SCDev are used) a PKCS#11 cryptographic service provider module,
- PKCS#11 driver.

Both modes demand the following hardware configuration requirements:

- CPU: 400 MHz or higher,
- RAM: 256 Mbyte or more,
- Disk capacity: 20 Mbyte or more,
- PKCS#11 token.

A2-API is an extended and re-developed version of the earlier certified product „A1-Polysys CryptoSigno JAVA API for qualified electronic signatures v1.1.0" ("**A1-API**") (registration number: Hung-T-25/2004) electronic signature application development kit.

A2-API is compatible downwards with A1-API meaning that A2-API is capable of generation and verification of signatures that are acceptable by A1-API.

As a result of the extension and re-development A2-API

- implements electronic signature generation and verification functions involving full scale of standard XAdES formats (XAdES-BES, EPES, T, C, X, X-L, A) and providing interoperability (according to the „Recommendation of Ministry of Informatics and Communication of the Republic of Hungary for the technical specification of electronic signatures applicable in public administration – Date: 22.11.2005.”),
- requests and checks time-stamps according to RFC 3161 (TSP, X.509 Internet Public Key Infrastructure Time-Stamp Protocol) during the generation and verification of signatures,
- is capable of making OCSP requests and checking OCSP responses according to RFC 2560 (OCSP, X.509 Internet Public Key Infrastructure Online Certificate Status Protocol) during the generation and verification of signatures.

Annex 2
Secure usage terms

In the following paragraphs those terms are summarised that should be fulfilled in order to contribute to the security of electronic signature applications developed with A2-API. Apart from the general quality assurance (design, test, documentation etc.) development requirements these conditions intend to guarantee the controlled security level of signature-specific elements of the product.

Mandatory rules for developers

The term below formulates a mandatory rule (condition) for developers producing signature applications with A2-API pointing out how they should use this product when incorporating it into their own developments.

1. A2-API is delivered to its users on CD or PenDrive, or it can be distributed on-line, downloaded by Java WebStart technology. Before putting into use, users should make a backup copy so that the original can serve as a master copy.

The primary users of A2-API are the developers producing applications with the development kit. At the same time, the functions of A2-API will be realised basically in the applications developed with it. Therefore users of the relying applications will appear as secondary users (who possibly will be divided into user and administrator roles).
Likewise, on the one hand the A2-API IT environment is the development environment of user application of the development kit, on the other hand it is the operational environment of the application developed with it. The following assumptions refer to the concept of user and IT environment in this double meaning.

Assumptions for the A2-API IT environment

1. Authorised users are trusted to accomplish duties assigned to them.
2. A2-API is installed and configured properly.
3. For the normal operational mode of A2-API (generation of advanced electronic signatures) the IT environment contains one or more cryptographic module(s) which meet(s) at least the requirements of FIPS 140 Level 1 and is (are) applicable for the implementation of the following operations: generation of RSA key pairs (at least 1024 key size), generation of 256-bit AES key, generation and verification of digital signature (with RSA algorithm), encryption and decryption (with AES algorithm), secure hash generation (with SHA-1 algorithm), generation of random numbers.
4. For the strict operational mode of A2-API (generation of qualified electronic signatures) the IT environment contains:
   - a certified SSCD device supporting PKCS#11 interface registered by the National Communication Authority (NCA) that is capable of implementing the following operations: generation of RSA key pairs (at least 1024 key size), generation of digital signature (with RSA algorithm), decoding AES key encrypted by RSA;
   - one or more cryptographic module that meet(s) at least the FIPS 140 Level 1 requirements and are (is) capable of implementing the following operations: verification of digital signatures (with RSA algorithm), generation of 256-bit AES keys, encryption of the generated AES keys with RSA (with at least 1024-bit key size), encryption and decryption (with AES algorithm), secure hash generation (with SHA-1 algorithm).

5. The assumed attack potential against A2-API is low.
6. The IT environment provides for the physical protection, so A2-API software elements are protected from the unauthorised physical access.
7. The application calling the A2-API functionality provides for the certificate and certificate revocation list information for the development kit.
8. The IT environment provides for system time with acceptable accuracy.
9. The hardware configuration is expected to meet the following additional requirements:
    - CPU: 400 MHz or higher,
    - RAM: 256 Mbyte or more,
    - Disc capacity: 20 Mbyte or more.
10. Sun Java JRE 1.5 Java Runtime Environment is available for the operating system (Linux, Solaris, Unix, Java Desktop System, Windows) and the latter is capable of running it. If JRE v1.5 is not available on the platform used, then it is replaceable by the combination of JRE 1.4.2 and PKCS#11 cryptographic service provider module.
11. In case of strict operational mode (for generation of qualified electronic signatures) if the environment is not considered as secure, for the signature application produced with A2-API development kit such a PKCS#11 driver or other hardware/software components should be used that are capable of building a trusted path with the SSCD, thus assuring the confidentiality of the signatory's authentication data (PIN) transferred to the SSCD and the integrity of protocol data and of the hash made on the data to be signed.
12. The IT environment contains such an SSCD/SCD interface software that is capable of providing the changing of authentication data based on knowledge (PIN), during the changing process is capable of requesting the new PIN twice, or this functionality must be included in the application produced with the use of A2-API development kit.

Assumptions for the secure usage of A2-API

The following conditions (user responsibilities) apply to the secure usage of the product:

1. Users should provide the acceptable accuracy of system time. System time should be set accurately, and this accuracy must be maintained by periodic checks and synchronisation.
2. In strict operational mode users should provide for the SSCD physical protection. Users should do their best against the theft of the SSCD, against its physical damage, and they should avoid the locking of the SSCD after entering incorrect PINs several times consecutively.
3. In strict operational mode users should keep their authentication data (PIN, password or passphrase) secret that is necessary for SSCD access. The authentication data should not be noted electronically or on paper in such a way that can make the passwords accessible or understandable by other individuals.
4. In normal operational mode users should provide for the physical protection of the SCD or PKCS#12 key storage (e.g. file, PEN drive etc.). Users should do their best against the theft of the SCD, and they should avoid the locking of the SSCD after entering incorrect PINs several times consecutively and against the copying of the content of PKCS#12 key storage.
5. In normal operational mode users should keep their authentication data (PIN, password or passphrase) secret that is necessary for SCD or PKCS#12 access. The authentication data should not be noted electronically or on paper in such a way that can make the passwords accessible or understandable by other individuals.

Assumptions for the applications developed with the usage of A2-API

These conditions are strongly recommended to be followed by application developers so that the application itself will be adequately secure.

1. The administrator guidance of the application should contain those points that are listed in the „Assumptions for the A2-API IT environment" section of this annex.
2. The guidance documents of the application should contain those points that are listed in the „Assumptions for the secure usage of A2-API" section of this annex.
3. It is advisable that applications divide the roles into user and administrator roles. In this case only the individual or process belonging to the administrator role can do the following activities:
   - management of top-level trusted authentication certificates,
   - setting the security attribute „CRL validity period (number of days) after distribution",
   - setting the security attribute „CRL validity period (number of days) after next distribution date",
   - setting the security attribute „accessibility of time-stamping service",
   - setting the security attribute „accessibility of OCSP service",
   - in case of OCSP service support the setting of the following security attributes:
     - OCSP response validity period (in minutes) after its generation,
     - OCSP response validity period (in minutes) after its distribution,
     - OCSP response validity period (in minutes) after the next distribution,
     - authorisation of OCSP refresh checks.
   - setting the security attribute „organisational signature policy",
   - default values of security attributes.

4. The installation procedures of applications should be worked out to include A2-API correct set-up.
5. The security attributes affecting the operation of A2-API should be set with the knowledge of the application user and according to their intent. The application should be implemented to have appropriate GUI interface for the users to set the security attributes. On GUI interfaces users should be informed of the vulnerabilities regarding the values of the security attributes that users intend to set. The setting of security attributes should not happen automatically, without the clear knowledge of users.
6. The application should not store the following security attributes persistently set for one execution (or the application should disregard them in the next execution):
   - authorisation of CRL refresh checks,
   - authorisation of disabling revocation checking.

   It is recommended that the application does not set values of security attributes listed above, instead the user on the appropriate interface for each execution responsibly and fully aware initiates the setting.
7. During application usage the revocation check should be set (see A2ApiControl class serviceSetBypassRevocationCheckEnabled method). Revocation check may be disabled only if justified, with the knowledge and intent of the application user. In the application user guidance users should be informed of the necessity of repeating the signature verification with an enabled state of the revocation check attribute, as soon as possible.
8. During application usage the revocation list (CRL) refresh checks should be set (see A2ApiControl class serviceSetCRLFreshnessCheckEnabled method). Refresh checks may be disabled only if justified, with the knowledge and intent of the application user. In the application user guidance users should be informed of the necessity of repeating the signature verification with an enabled state of the CRL refresh check attribute, as soon as possible.
9. If it is possible, during application usage, the system time should be obtained not from the clock of the IT environment but from a trusted time server (see A2ApiControl class, setTimeServers method).
10. During application usage the values of the security attributes CRLAfterThisUpdateLimit and CRLAfterNextUpdateLimit should be set to the minimum value possible (1 day and 0 day). Usage of greater value can cause that a revoked certificate may be approved based on old CRL (see A2ApiControl class, serviceSetCRLAfterThisUpdateLimit and serviceSetCRLAfterNextUpdateLimit methods).
11. During the application execution, the building and validation of certificate path of certificate belonging to identification/authentication should be accomplished as soon as possible (see A2ApiControl class, serviceCheckIAACertificate method).

Conditions for the secure usage of A2-API

1. For signature applications developed with A2-Polysys CryptoSigno Interop JAVA API development kit a PKCS#11 driver should be used that is capable of maintaining the integrity of the hash made on the data to be signed representation (DTBRS) and all protocol data.
2. For signature applications developed with A2-Polysys CryptoSigno Interop JAVA API development kit a PKCS#11 driver should be used that is capable of maintaining the confidentiality of the signatory's authentication data.
3. In the operational environment of the signature application using A2-Polysys CryptoSigno Interop JAVA API development kit technical and procedural measures should be followed in order to assure that non-trusted system and application processes, peripheral devices and communication channels not necessary for signature generation applications should not be able to interfere with the signature process.
4. For the signature application using A2-Polysys CryptoSigno Interop JAVA API development kit:
    a. either a PKCS#11 driver should be used that is capable of setting a timeout for the period which may pass from entering the signatory's authentication data to initiating the signature,
    b. either the application should meet this requirement by requesting the authentication data after initiating the signature process.
5. For the signature application using A2-Polysys CryptoSigno Interop JAVA API development kit:
    a. either a PKCS#11 driver should be used that is capable of cancelling the signature process after a timeout and only after re-requesting the authentication data it starts the process and informs the signatory of the necessity of the restart.
    b. either the application should meet this requirement by requesting the authentication data after initiating the signature process.
6. For the signature application using A2-Polysys CryptoSigno Interop JAVA API development kit the SSCD driver software should be installed, which provides for the changing of the authentication data based on knowledge.
7. The SSCD driver software installed for the signature application using A2-Polysys CryptoSigno Interop JAVA API development kit should provide for the capability of entering the new PIN twice during PIN change in order to avoid incorrect input.
8. In the operational environment of A2-Polysys CryptoSigno Interop JAVA API development library technical and procedural measures should be applied in order to provide for the following:
    a. viruses will not damage the signature application and other signature components used by it, and
    b. signature components accidentally infected by viruses will be recovered correctly.

9. In the operational environment of A2-Polysys CryptoSigno Interop JAVA API development library technical and procedural measures should be applied in order to provide for the integrity protection of functional components of A2-Polysys CryptoSigno Interop JAVA API development library, preventing attackers from damaging it.
10. In the operational environment of A2-Polysys CryptoSigno Interop JAVA API development library technical and procedural measures should be applied in order to implement the A2-Polysys CryptoSigno Interop JAVA API development library and all of its components interacting with signature generation, signature verification processes in a secure area.

Annex 3
Product compliance requirements

Requirements

- Act XXXV of 2001 of the Republic of Hungary on electronic signature
- CEN/ISSS/E-Sign; CWA 14170:2004; Security requirements for signature creation applications
- CEN/ISSS/E-Sign; CWA 14171:2004; General guidelines for electronic signature verification
- ETSI TS 101 733 v1.5.1 (2003-12) Electronic Signature Formats
- ETSI TS 101 903 v1.2.2 (2004-04) XML Advanced Electronic Signatures (XAdES)
- PKE PP (Public Key-Enabled Application Family of Protection Profiles) with < Certification Path Validation (CPV) – Basic, PKI Signature Generation, PKI Signature Verification, PKI Encryption using Key Transfer Algorithms, PKI Decryption using Key Transfer Algorithms, Certificate Revocation List (CRL) Validation Online Certificate Status Protocol Client > at EAL <3> with augmentation /Version: 2.5, 31.10.2002/

Standards

- PKCS #1 RSA Cryptography Standard /RFC 2313/
- PKCS#11 Cryptographic Token Interface Standard
- PKCS#12 Personal Information Exchange Information Standard
- RFC 3161 TSP, X.509 Internet Public Key Infrastructure Time-Stamp Protocol
- RFC 2560 OCSP, X.509 Internet Public Key Infrastructure Online Certificate Status Protocol
- RFC 3280 Internet X.509 PKI Certificate and Certificate Revocation List (CRL) Profile
- RSA Rivest-Shamir-Adleman (public key cryptosystem) /ANSI X9.31/
- SHA-1 Secure Hash Algorithm /FIPS PUB 180-1/

Annex 4
Further details about the certification procedure

Developers' documents examined during certification

- Request for certification
- Questionnaire for request for certification
- Protection Profile: PKE PP /Public Key-Enabled Application Family of Protection Profiles) with < Certification Path Validation (CPV) – Basic, PKI Signature Generation, PKI Signature Verification, PKI Encryption using Key Transfer Algorithms, PKI Decryption using Key Transfer Algorithms, Certificate Revocation List (CRL) Validation > at EAL <3> with augmentation - Version 2.5 - 31.10.2002/
- Security target v2.0
- Functional specification v2.0
- High level design v2.0
- Test coverage and depth analysis v2.0
- Test documentation v2.0
- Configuration management v2.0
- Change control management v2.0
- Delivery procedures v2.0
- Supporting documents v2.0
- Guidance for developers v2.0 /a2-api-DOC-2_0_0.jar/
- Vulnerability assessment v2.0
- Annex v1.0

Documents independent from developers examined during certification

- Evaluation report on A2-Polysys CryptoSigno Interop JAVA API development kit for qualified electronic signatures (created by the evaluator division of Hunguard Ltd.)

Method of independent assessment checking the requirement compliance

The evaluation of A2-API has been done according to the Hungarian IT Security and Evaluation Scheme (Hungarian abbr.: MIBÉTS) methodology that is for the technological evaluation of information systems and products. For this purpose MIBÉTS acknowledges the concepts, principles and criteria of the Common Criteria (MSZ ISO/IEC 15408:2002).

Evaluation level

Moderate security level (EAL3+)

Documents concerning methodology used during evaluation

- MIBÉTS publication No 1: General model of National MIBÉTS scheme /v0.95, February 2005./,
- MIBÉTS publication No 2: Procedures of evaluation and certification /v0.95, February 2005./,
- MIBÉTS publication No 3: Evaluation methodology 1 – Evaluation methodology of the Security Target /v0.95, February 2005./,
- MIBÉTS publication No 3: Evaluation methodology 3 – Evaluation methodology of the moderate assurance level /v0.95, February 2005./.