PkBox – Technical Overview

Ver. 1.0.7

Index

1. Introduction

PkBox is a powerful and flexible security server designed to manage all Digital Signature operations (both Massive and Remote) according to law, encryption (symmetric and asymmetric) and Strong Authentication to support applications that must deal with logical data security.

PkBox is useful in all the situations where it is necessary to handle large volumes of transactions with maximum flexibility, reliability and scalability of the implemented solution.

In order to ensure an excellent solution to the problems of data security, Intesi Group introduced in PkBox (one of PkSuite products) several features and characteristics that make it an extremely scalable and flexible solution, in terms of number of documents to protect, processing of large documents and management of large amounts of users (keys and certificates).

This document will cover only the analysis of high level functional characteristics and doesn't address topics related to programming applications trough APIs made available by the product, all this information is available in other set of documents. It is worthwhile to mention that the offered features have been implemented at a very high level of abstraction that hides most of the implementative complexity of the offered features. This approach reduces to a minimum the learning time of the system and it doesn't require specific technical skills from developers (a developer that uses PkBox interfaces not necessary is an expert of security issues). Most of the operating characteristics of the offered features are managed via configuration parameters.

The source code of the applications that use PkBox functionalities is independent from particular configuration set at server level. In this way, changes to the server configuration do not require changes to the application code already developed, ensuring high reusability of the developments already made.

Leveraging the ability of PkBox to simultaneously manage different configurations (called Environment), multiple applications can be configured to perform encryption operations with different operational modes, without any preliminary setting during development and without any operation on the source code.

This aspect is of great importance in the economy of the common development and integration operations because the same application can be proposed in different configurations and can

handle different types of encryption operations without having to modify the application source code.

For further details on PkSuite APIs please see the specific technical documentation available on the website www.pksuite.it

2. PkBox features

Depending on the product configuration, PkBox offers different features. The product is available in three different functional versions: **Advanced, Enterprise and EMV**.

Please refer to PkBox technical spec for all details regarding the availability of specific features for each product configuration.

Features offered by PkBox are:

- Digital Signature / Digital Signature Verification
- Electronic Signature / Electronic Signature Verification
- Time Stamps Request / Time Stamps Check
- Strong Authentication
- Asymmetric Encryption
- Symmetric Encryption
- EMV commands
- Utilities.

Features offered by PkBox are accessible through programming interfaces such as:

- Web Services
- Java Remote API
- .Net Remote API

3. Digital Signature

Digital Signature is a digital authentication system that ensures non-repudiation and integrity of data to which was affixed signature. The digital signature process is based on encryption with public and private asymmetric keys (or Public Key Infrastructure - PKI).

Qualified Digital Signature has a legal value, as it identifies in a reliable and trustworthy way the person who signed the document, and for this reason is compared to the signature that is normally affixed to the traditional paper documents.

PkBox implements the digital signature features and the corresponding signature verification, including the certificates verification processes (certification path of the Certification Authority that issued the certificate, time validity, verification of profiles and validity of the certificate).

PkBox also supports the calculation and verification of single, multiple and parallel signatures and countersignatures.

Digital Signature formats supported are:

- RSA pkcs#7
- ISO 3200 PDF signature
- XMLDSIG
- ETSI CAdES (CAdES-BES, CAdES-T)
- ETSI PAdES (PAdES-Basic, PAdES-BES, PAdES-T)
- ETSI XAdES (XAdES-BES, XAdES-T, XAdES-C, XAdES-X-1, , XAdES-X-2, XAdES-X-L, XAdES-A).

Digital Signature algorithm supported is:

- RSA (512, 1024, 2048, 4096).

Hash calculation algorithms supported are:

- MD2 and MD5
- SHA-1, SHA-256 and SHA-512
- RIPEMD-128 and RIPEMD-160

4. Time Stamps

Time stamp is a computing evidence that allows to make enforceable against third parties the time reference. Time stamp of an electronic document is a digital signature (in addition to the one of the subscriber), issued by a trusted third party (typically the Certification Authority/Time Stamp Authority) to which is associated the document hash (or the digital signature hash of the document, if available) and the information related to a certain date and time.

The purpose of a time stamp is to ensure with certainty the "existence" of a document (digitally signed or not) at a certain date.

PkBox supports access to timestamping services of Time Stamp Authorities through the protocols and formats specified in RFC 3161 spec.

The product manages the affixing and recognition of time stamps to signed documents in accordance with RFC 3161 spec, to generic documents in accordance with RFC 5544 spec and in detached mode via TimeStampToken or TimeStampResp.

Time stamps digital signature algorithm supported is:

- RSA (512, 1024, 2048, 4096).

Hash calculation algorithms supported are:

- MD2 and MD5
- SHA-1, SHA-256 and SHA-512
- RIPEMD-128 and RIPEMD-160.

5. Strong Authentication

Authentication is a process by which a system verifies the identity of another person, user or software, which requires access to resources or services. While accessing services available on the Web is important for the user to use simple and secure methods to define identity, and for the services supplier to identify with certainty and reliability the identity of people who require access.

For the most critical services or with a high value of the processed information, such as financial services or services that manage sensitive personal information, is extremely important that the users' virtual identity is uniquely associated to the "physical" one.

In many cases, the authentication process may be repeated several times a day: this implies that, in addition to the safety requirements needed to recognize the user in a reliable manner, the process must require simple mechanisms to ensure a high usability for users.

A good authentication process should provide flexible mechanisms to ensure the recognition of users and the access to the provided services trough different devices (such as PC, smartphone, tablet, ...).

A strong authentication mechanism aims to raise the level of reliability of the recognition process of entities that require access, using different multi-factor authentication.

Typically, the authentication factors used are:

- knowledge of a secret: a password or PIN
- possession (or demonstration of a possession): a safety device (smartcard or token), a credit card or a mobile phone
- a biometric feature: a fingerprint or in general a unique feature of the human body.

The multi-factor authentication systems typically are based on two separate factors among those listed before. A two-factor authentication is more secure than a single factor mechanism (such as a single password) as the "loss" of one of the authentication parameters is not sufficient for an attempt of fraudulent access.

The most common mechanisms of two-factor authentication are based on "the knowledge of a secret" (typically a password) and on "possession". The use of ATM is an example of two-factor

authentication: the ATM card is the object that represents the "possession" and the PIN is "the knowledge of the secret."

PkBox supprts the following auhentication mechanisms:

- via digital certificates
- via token OTP (token hardware, virtual or via SMS)
- via PIN cards

In the case of authentication through digital certificates, PkBox implements a mechanism of challenge/response signed using X.509 digital certificates. With this mechanism it is possible to identify very strongly both users towards the provider of the service, both services towards users who require access. In this case the first authentication factor is the signature device that contains keys and its digital certificate, the second factor is PIN to accede the device.

In the case of authentication via OTP tokens or PIN cards, PkBox implements a verification mechanism of dynamic passwords. In case of OTP authentication, PkBox internally implements the calculation and verification of OTP values through standard algorithms such as OATH Time Based (TOTP) and Event Based (HOTP). Moreover, PkBox integrates OTP validation systems of other vendors available on the market, offering an homogeneous and transparent integration application interface OTP vendor-independent. In this case the first authentication factor is the OTP device or the PIN card, the second factor is represented by a password associated with the user or OTP token.

OTP authentication providers are:

- Time4ID SMS and Mobile Token OTP (TOTP e HOTP algorithms)
- Time4ID OATH (TOTP e HOTP algorithms)
- Vasco Vacman controller
- Vasco IdentiKey Server
- RSA SecureID Authentication Engine
- RSA SecureID ACE Server
- Radius

6. Asymmetric Encryption

Asymmetric encryption, also known as public and private key encryption, is a cryptographic process in which each entity is associated with a pair of keys:

- the public key is used to encrypt a document "adressed" to the entity that owns the corresponding private key;
- the private key is used to decrypt a document encrypted with the public key.

The main feature of asymmetric key algorithms is that a document or a message encrypted with the public key can be decrypted only with the corresponding private key: the public key is used only to encrypt, the private key is used only to decrypt. An entity that needs to send an encrypted document/message to a recipient must therefore be in possession of the public key of the recipient. The recipient uses his private key to decrypt the encrypted document/message.

This mechanism requires the distribution of the sole public key (an information that doesn't need to be kept in a protected form) between the various parties wishing to exchange documents in a secure (encrypted) way, avoiding the exchange of secret information typical of the mechanisms based on symmetric key (that works with a single public key for both encryption and to decryption). Each user has its own pair of keys: private key is kept secret and stored in a safe place/form, the public key is shared with the various parties in a free manner: via e-mail, published in lists or directories accessible to a community of users.

The mechanism of asymmetric encryption also helps to protect (encrypt) a single document for N different recipients using the public key of each recipient. In this way you can share a single document in a protected way avoiding to create an encrypted copy for each recipient.

The encryption process of a document involves the generation of a symmetric key used to encrypt the document, the symmetric key is encrypted N times using the public key of each of the recipients of the document. This approach offers two very interesting plus:

- to use a symmetric key encryption algorithm to encrypt the document provides more performances than an asymmetric key algorithm
- use an asymmetric key encryption algorithm to protect the symmetric key guarantees the mechanisms of simplicity and security described above.

PkBox supports the following formats of encrypted documents:

- CMS Enveloped Data - RFC 3852
- PGP Encrypted Data - RFC 2440 (PGP format)

The asymmetric encryption algorithm supported is:

- RSA (512, 1024, 2048, 4096).

Symmetric encryption algorithms supported are:

- DES and triple DES
- RC2 and RC5
- AES 128, AES 192 and AES 256.

7. Symmetric Encryption

Symmetric encryption, also called secret key cryptography, is a cryptographic process in which each entity shares the same key value to exchange documents or data in encrypted form.

The security of a symmetric key cryptographic system is based on the following key features:

- the secrecy of the keys used by users;
- the goodness of the algorithm of generation and use of keys;
- the length of the used keys.

Symmetric encryption uses simpler and faster mechanisms and algorithms than the asymmetric encryption, and for that reason is "preferred" for encryption operations that require high performance, but requires that before both share a key between the various stakeholders. It's clear that the method of exchanging keys represents the critical point of system security.

The exchange mechanisms of symmetric keys can be different. In some areas, the keys are exchanged manually in several segments (also called components), each piece of the key is known to a different entity (a Key Manager). It's also possible to exchange in a simple and secure manner a symmetric key using asymmetric encryption: the public key would be exchanged in unsecured form, the symmetric key would be exchanged in encrypted form through the mechanism of public and private keys.

PkBox supports the following encryption formats:

- ECB
- CBC
- CBC with padding (PKCS # 5 or PKCS # 7)

Symmetric encryption algorithms supported are:

- DES and triple DES
- RC2 and RC5
- AES 128, AES 192 and AES 256.

8. EMV Commands

PkBox offers a specific set of cryptographic security services to manage electronic payment systems. The cryptographic services like Europay Mastercard Visa (hereinafter referred EMV) offered by PkBox are useful to manage the processes of emission and the authorization of electronic payment instruments such as debit cards (ie ATM cards) or credit cards.

The cryptographic EMV features were classified according to the different categories of operations performed. Each set of functions is organized in a specific service.

The division of functions in services, allows to raise the level of security offered by PkBox EMV solution. Cryptographic commands used by application services enabled to accede to single installation are sharply separated. Thanks to the division into separate modules is prevented, for example, to applications that manage the authorization processes to use commands necessary to the Issuing processes.

The security services offered by PkBox EMV are divided into the following categories:

Issuing

In the Issuing module are included all the cryptographic commands necessary to card issuance processes and PIN generation (eg PVV, CVV, CVV2, offset calculation, …).

EMV

The EMV module provides cryptographic commands for EMV services of "Data preparation" (cards customization), authentication and key derivation.

Authorization

Authorization module provides cryptographic controls for the management and validation of authorization messages (MAC calculation and verification, verification and transcoding PinBlock, validation IN/PVV, change PIN, CVV verification/CVV2, management and cryptograms 3D secure EMV).

Petrol

> Petrol module includes a set of commands for the management and validation of authorization messages for fuel cards (management Shell circuits, DVK, Routex, Uta and Lomo).