PKA - Private Key Access
Technical White Paper

Overview

In this white paper we describe the use of VASCO's "Patent Pending" Private Key Access (PKA™) concept.
PKA is a concept in which the Private Key, of a Public and Private Key-pair, is stored in an encrypted way on the hard disk of a user's PC. Any key on a hard disk that has to be encrypted can be protected this way.
PKA-enabled applications use a Digipass® 300, 300C, 600 or 700, Vasco's authentication devices, and a small item of client plug-in software. The Digipass will produce a dynamic password each time the Private Key needs to be used. As such it uses the strong authentication capabilities of the Digipass family.

Strong authentication is based on two-factor security:

- Something the user knows: the PIN code to activate the Digipass
- Something the user possesses: the Digipass itself

To generate a dynamic password you need both factors, otherwise, the Private Key can not be accessed, even if it is stored on the PC's hard disk.

Please refer to "VASCO Digipass Family of Authentication Devices - Technical White Paper" for more detailed information on strong two-factor authentication devices.

Fig 1. From left to right the PKA capable devices: Digipass 300, Digipass 300C, Digipass 600, Digipass 700 and Digipass 800.

Problem Description

Today "Public Key Infrastructure" (PKI) is greatly talked about. PKI is a concept in which, via the RSA encryption algorithm, asymmetric key-pairs are created and then distributed to the users. Such a key-pair consists of a "Private Key" and a "Public Key".
The owner of such a key-pair can distribute or publish his Public Key, but must guard his or her Private Key with great care.
Without going into further detail on the concept of PKI, it essentially means that anybody can encrypt a message with this Public Key and send it to the owner of the Private Key. The message could be, at worst intercepted but not decrypted since the interceptor doesn't possess the Private Key. Only the real addressee can decrypt and read the message.
Vice versa, the key-pair owner can encrypt a message with his Private Key and send it to an addressee who has his Public Key. If this addressee can decrypt the message he knows that he holds a message sent by the right person.

It is obvious that the security level of such a system relies on the way this Private Key is safeguarded.

Today, there are two common safe storage techniques:

- The first one is to store it on a smart card that can be inserted into the PC, anytime the Private Key is needed.
- The second method is to store the Private Key on the PC's hard disk and use a static password or pass phrase to decrypt it. Since static passwords are easy to compromise and can be quickly broken into and is not fail-safe. PKA is the answer to this lack of security because it will replace the static password with a dynamic password generated by a Digipass.

Throughout this white paper VASCO provides you with security guidelines concerning the use of a Private key at a higher level.

Concept

The following illustrations visualize the PKA concept. Fig. 2 shows how the Digipass generates the dynamic password. In Fig. 3 you see how the dynamic password is verified by the application running on the client's PC.

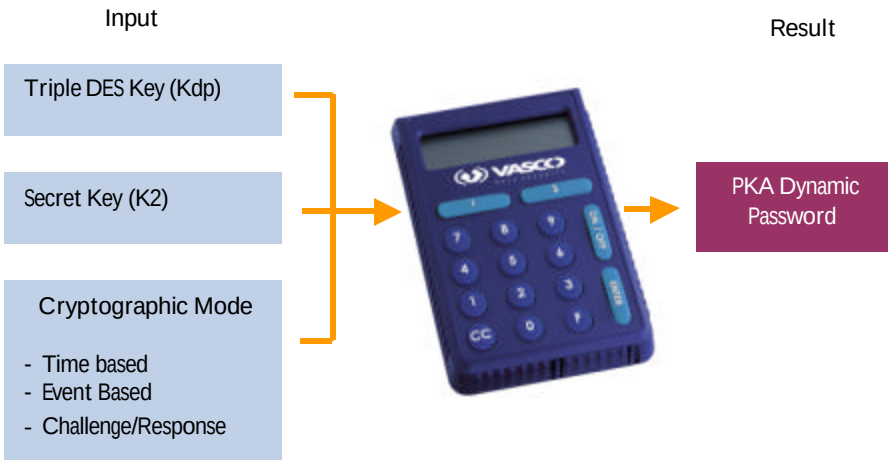

Fig 2. The initialization parameters for PKA in a Digipass 700.

When the Digipass is initialized, the special secret key (K2) as well as the normal key "Kdp" is generated by the initialization software. Once the secret key is generated, it will be programmed into the Digipass. This secret key (K2) is the primary element for PKA use. It is this K2 that will be regenerated by the PKA verification software on the client PC, to encrypt and decrypt the Private Key of the RSA key-pair.
In order to verify the dynamic password on the PC-side, there are a few things to be done.
First the Digipass secrets have to be imported into the application's database. In Digipass-enabled applications you need to use a Digipass import file (. DPX format) in order to store the necessary Digipass secret keys into the database system of the application. The application software that imports the initialized Digipasses into its database in an encrypted format would then use this imported file.
The total cost of ownership for the application has been kept as low as possible (read: preventing the shipment of encrypted files on floppy disks). Instead of using an imported file you can use a simple activation code. This activation code will be enough to activate the usage of a unique Digipass for PKA usage into your client PC application. The first time you use the application, the software will ask you to enter this activation code and will then ask you twice for a PKA dynamic password. This "double" dynamic password is used to check whether you have received the right activation code for this Digipass. If you have received the right activation code, the application software will immediately encrypt your Private Key. From that point on you will need a PKA enabled Digipass to decrypt your Private Key.
The next section illustrates how the dynamic password - protecting the Private Key - is verified on the client PC.

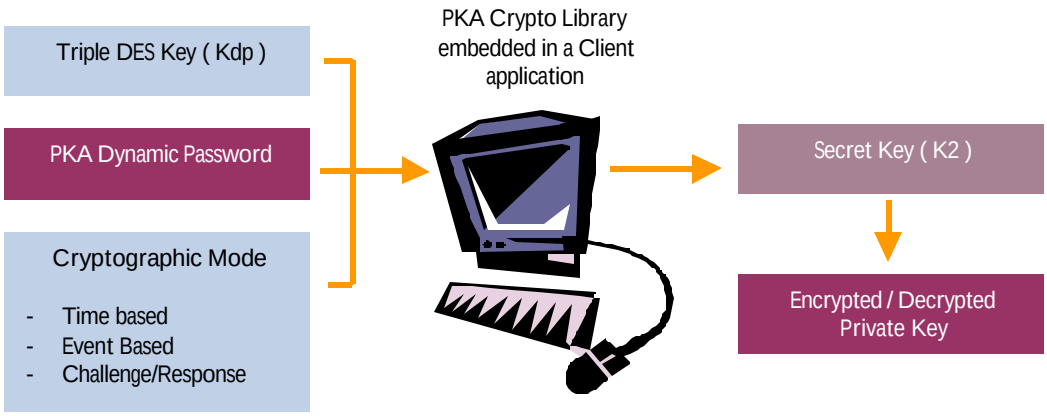

Fig 3. The verification process of a PKA dynamic password on a client PC.

When comparing Fig 2 with Fig 3 you will see that the two fields switched position between the input and the result side.
In Fig 2 "PKA Dynamic Password" has become an input field, because it needs to be verified. "The Secret Key" (K2) became the result of the calculations performed by the PKA library integrated into the application.
In fact it appears as if the secret key (K2) has been exported out of the Digipass into the application. As mentioned before the secret key (K2) is the most important element of PKA. This means that the encryption and decryption of your locally stored Private Key can be performed based on a secret key that is never stored on your hard disk.

We have now reached the point where a locally stored Private Key (or any other piece of stored information) can be encrypted and decrypted by means of a dynamic password.

System components

- Following members of the Digipass Family support PKA:
  - Digipass 300
  - Digipass 300C
  - Digipass 600
  - Digipass 700
- VASCO's Private Key Access Toolkit.
  - This toolkit contains all procedures, written in native C source code, needed to convert an existing or new application into an application that uses dynamic passwords, together with encryption protection for your private data stored on your system.
  - The toolkit is an integration tool for the client PC application. A member of the Digipass Family uses it to initialize the application with a PKA dynamic password or to verify the PKA dynamically generated passwords.

For more information

VASCO U.S. Headquarters at:
1.800.238.2726 or e-mail your information requests to
info_usa@vasco.com

VASCO Europe Headquarters at:
+32 2.456.98.10 or e-mail your information requests to
info_europe@vasco.com

VASCO France at:
+33 5 57.19.11.00 or e-mail your information requests to
info_france@vasco.com

Or visit our corporate web site on: http://www.vasco.com

About VASCO

OVERVIEW:
VASCO Data Security, a US corporation, helps organizations rapidly and effectively secure their e-business and e-commerce applications and services. The company's family of Digipass®, and SnareWorks™ products offers end-to-end security through PKI, strong user authentication, true Single Sign-On, Web access control, enterprise management, and encryption solutions that also sharply reduce the time and effort required to deploy and manage them. The company has more than 100 employees in headquarters and development centers in the US, Europe and Asia.