Hardware algorithm services

Flags controlling the algorithms services are located in **Available services** group. The use of services significantly expands the functionality of hardware algorithms.

The services allow for:

- Setting the algorithm status (active/inactive) and its further control from within the application or using the remote update feature
- Accessing the algorithm descriptor contents and updating it without affecting (overwriting) the remaining memory (compare to hardware locks ideology).

Such possibilities became available thanks to ***protected items technology*** with hardware algorithms being a special case of such technology.

If **Activation** service is on in the algorithm properties, this algorithm can be made active by accessing it with a special Guardant API function from within the application or by executing the update procedure.

After activation you can execute all preprogrammed actions with the algorithm: perform encryption, read algorithm descriptor contents, update specific areas of descriptor, deactivate algorithm.

Examples of use

1. *Remote activation of an algorithm responsible for operation of additional application modules after payment from the end-user is received.*
2. *Activation of an additional algorithm on a special event for complicating the protected application logic.*
3. *Activation of a new and deactivation of the old algorithm upon release of a new version of application. Whereas all algorithms should be previously constructed, and their properties and services should be predefined.*

GrdPI_Activate function serves for activating an algorithm from within the application.

Setting the **Activation** flag turns on the service and enables the following options (see information below for more details):

- **Activation password** entry field
- **Unified/random password** entry field
- **Set inactive status** flag

Also another option common for all services appears:

- **Password entry attempts** field

After turning on **Activation** service set the activation password, define its type and the number of password entry attempts.

If the application protection scheme presumes that the algorithm must be initially deactivated, **Set inactive status** flag is required. The inactive algorithm will appear in palegray font in Mask Editor.
If **Deactivation** service is on in the algorithm properties, this algorithm can be made inactive by accessing it from within the application using special Guardant API function or by performing the update procedure.

After deactivation the algorithm can be only activated (if activation service was previously turned on in the algorithm properties). Other actions programmed for this algorithm are disabled at this stage.

GrdPI_Deactivate function serves for deactivating an algorithm from within the application.

Setting **Deactivation** flag turns on the service and enables the following options (see information below for more details):

- **Deactivation password** entry field
- **Unified/random password** entry field

Also another option common for all services appears:

- **Password entry attempts** field
  If it has not been already enabled after turning on another service

After turning on **Deactivation** service set the deactivation password, define its type and the number of password entry attempts.
If **Data reading** service is on in the algorithm properties, you will be able to get information on the determinant contents of this algorithm by accessing it from within the application using a special Guardant API function.

If **Password-protected reading** is on during this session, you will need to enter the correct password to execute the reading command.

Use ***GrdPI_Read*** function for accessing hardware algorithm contents from within the application.

Setting **Data reading** flag turns on the service and enables **Password-protected reading** option at the same time.

Setting flag **Password-protected reading** turns on the service and enables the following options (see information below for more details):

- **Password for data reading**entry field
- **Unified/random password**entry field

Also another option common for all services appears:

- **Password entry attempts** field
  If it has not been already enabled after turning on another service

After turning on **Data reading** service turn on **Password-protected reading** service, set the password for data reading, define its type and the number of password entry attempts.
If **Data update** service is on in the algorithm properties, the determinant contents of such algorithms can be changed by accessing it from within the application using a special Guardant API function.

Use ***GrdPI_Update*** function to write new data into algorithm descriptor from within the application.

Setting flag **Data update** turns on the service and enables the following options (see information below for more details):

- **Password for updating data** entry field
- **Unified/random password** entry field

Also another option common for all services appears:

- **Password entry attempts** field
  If it has not been already enabled after turning on another service

After turning on **Data update** set the password for updating data, define its type and the number of password entry attempts.