

Сервисы аппаратных алгоритмов

В группе **Доступные сервисы** расположены флаги, управляющие сервисами алгоритмов. Использование сервисов значительно расширяет функциональность аппаратных алгоритмов.

Сервисы позволяют:

- Задавать состояние алгоритма (активное/неактивное)и, в дальнейшем, управлять им из приложения или путемудаленного обновления
- Получать доступ к содержимому дескриптора алгоритмаи обновлять его, не затрагивая (не перезаписывая) остальную память – ср. с идеологией аппаратных запретов.

Такие возможности стали доступны благодаря **технологии защищенных ячеек**, частным случаем которых являются аппаратные алгоритмы.

Активация

Если в свойствах алгоритма включен сервис **Активация**, то этот алгоритм в определенный момент можно сделать активным, обратившись к нему из приложения с помощью специальной команды **Guardant API** или выполнив процедуру обновления.

После активации с алгоритмом можно выполнять все ранее запрограммированные для него действия: выполнять преобразования, читать содержимое дескриптора алгоритма, обновлять определенные участки дескриптора, деактивировать алгоритм.

Примеры использования

1. Удаленная активация алгоритма, который отвечает за работоспособность дополнительных модулей приложения, после получения оплаты от конечного пользователя.
2. Активация дополнительного алгоритма по определенному событию для усложнения логики работы программы.
3. Активация нового и деактивация старого алгоритма при выходе новой версии приложения. При этом все алгоритмы должны быть заранее созданы и их свойства и сервисы должны быть определены.

Активация алгоритма из приложения выполняется с помощью команды **GrdPI_Activate**.

Установка флага **Активация** включает одноименный сервис, при этом становятся доступными опции (подробнее о них см. ниже):

- Поле ввода **Пароли** активации
- Поле ввода **Постоянный/случайный** пароли
- Флаг **Установить неактивное состояние**

А также появляется опция, общая для всех сервисов:

- Поле **Допустимое количество ошибок** при наборе паролей

После включения сервиса **Активация** задайте пароль активации, определите его вид и допустимое число ошибочного ввода.

Если схема защиты приложения предполагает, что алгоритм должен быть деактивирован изначально, установите флаг **Установить неактивное состояние**. Неактивный алгоритм выделяется в **Редакторе образа** бледно-серым шрифтом.

Деактивация

Если в свойствах алгоритма включен сервис **Деактивация**, то алгоритм в определенный момент можно сделать неактивным, обратившись к нему из приложения с помощью специальной команды **Guardant API** или выполнив процедуру обновления.

После деактивации алгоритм можно только активировать (если сервис активации был предварительно включен в свойствах алгоритма), но нельзя выполнять остальные действия, запрограммированные для этого алгоритма.

Деактивация алгоритма из приложения выполняется с помощью команды **GrdPI_Deactivate**.

Установка флага **Деактивация** включает одноименный сервис, при этом становятся доступными опции (подробнее о них см. ниже):

- Поле ввода **Пароли** деактивации
- Поле ввода **Постоянный/случайный** пароль

А также появляется опция, общая для всех сервисов (*если она уже не появилась после включения другого сервиса*):

- Поле **Допустимое количество ошибок** при наборе паролей

После включения сервиса **Деактивация** задайте пароль деактивации, определите его вид и допустимое число ошибочного ввода.

Чтение по паролю

Если в свойствах алгоритма включен сервис **Чтение данных**, то можно получить информацию о содержимом определителя этого алгоритма, обратившись к нему из приложения с помощью специальной команды [Guardant API](#).

Если при этом включить сервис **Чтение по паролю**, то для выполнения команды чтения необходимо будет указать верный пароль.

Доступ к содержимому аппаратного алгоритма из приложения выполняется с помощью команды [GrdPI_Read](#).

Установка флага **Чтение данных** включает одноименный сервис, при этом становится доступной опция **Чтение по паролю**.

Установка флага **Чтение по паролю** включает одноименный сервис, при этом становятся доступными опции (подробнее см. далее):

- Поле ввода [Пароли](#) для чтения данных
- Поле ввода [Постоянный/случайный](#) пароль

А также появляется опция, общая для всех сервисов (*если она уже не появилась после включения другого сервиса*):

- Поле [Допустимое количество ошибок](#) при наборе паролей

После включения сервиса **Чтение данных** включите сервис **Чтение по паролю**, задайте пароль для чтения данных, определите его вид и допустимое число ошибочного ввода.

Обновление данных

Если в свойствах алгоритма включен сервис **Обновление данных**, то содержимое определителя такого алгоритма можно будет изменить, обратившись к нему из приложения с помощью специальной команды [Guardant API](#).

Запись новых данных в дескриптор алгоритма из приложения выполняется с помощью команды [GrdPI_Update](#).

Установка флага **Обновление данных** включает одноименный сервис, при этом становятся доступными опции (подробнее см. далее):

- Поле ввода [Пароли](#) для обновления данных
- Поле ввода [Постоянный/случайный](#) пароль

А также появляется опция, общая для всех сервисов (*если она уже не появилась после включения другого сервиса*):

- Поле [Допустимое количество ошибок](#) при наборе паролей

После включения сервиса **Обновление данных** задайте пароль для обновления, определите его вид и допустимое число попыток ошибочного ввода.