

GrdCodeInit

Функция(метод) **GrdCodeInit** инициализирует пароль перед проведением быстрого взаимнообратного преобразования данных. Функция(или метод) **GrdCodeInit** были предназначены для работы с устаревшими ключами **Guardant Stealth**. Функция(или метод) реализованы исключительно в целях совместимости и использование их в современных приложениях не рекомендуется.

C

```
int GRD_API GrdCodeInit(
    HANDLE hGrd,
    DWORD dwCnvType,
    DWORD dwAddr,
    void *pKeyBuf
);
```

<i>hGrd</i>	хэндл, через который будет выполнена данная операция						
<i>dwCnvType</i>	метод быстрого взаимнообратного преобразования. Задается одним из флагов GrdAT_XXX <table><tr><td>GrdAT_Algo0</td><td>Базовый метод. Кодирование выполняется блоками по 32 байта. Этот метод лучше всего использовать для преобразования нестроковых данных. Важная особенность метода: если закодировать большой объем памяти (или файл) участками, то декодирование нужно будет обязательно делать точно теми же участками, или же размер участков должен быть кратен 32 байтам. Функции GrdEncode и GrdDecode не изменяют пароль.</td></tr><tr><td>GrdAT_AlgoSCII</td><td>Символьный метод. Кодирование выполняется блоками по 32 байта. Этот метод хорош для кодирования строковых данных. Например, если вы хотите закодировать название поля базы данных, указанное в исходном тексте программы, то в некоторых языках программирования будет проблематично создать строки с несимвольными значениями. Символьный метод кодирует строки так, чтобы в них не было неотображаемых символов. Функции GrdEncode и GrdDecode не изменяют пароль.</td></tr><tr><td>GrdAT_AlgoFile</td><td>Файловый метод. Кодирование выполняется блоками произвольной длины, поэтому такой метод подходит для кодирования файлов. Кодирование этим методом файла блоками меньше 32 байт неэффективно. Функции GrdEncode и GrdDecode в этом методе преобразования изменяют пароль, поэтому последовательность использования функции GrdDecode должна совпадать с последовательностью использования функции GrdEncode .</td></tr></table>	GrdAT_Algo0	Базовый метод. Кодирование выполняется блоками по 32 байта. Этот метод лучше всего использовать для преобразования нестроковых данных. Важная особенность метода: если закодировать большой объем памяти (или файл) участками, то декодирование нужно будет обязательно делать точно теми же участками, или же размер участков должен быть кратен 32 байтам. Функции GrdEncode и GrdDecode не изменяют пароль.	GrdAT_AlgoSCII	Символьный метод. Кодирование выполняется блоками по 32 байта. Этот метод хорош для кодирования строковых данных. Например, если вы хотите закодировать название поля базы данных, указанное в исходном тексте программы, то в некоторых языках программирования будет проблематично создать строки с несимвольными значениями. Символьный метод кодирует строки так, чтобы в них не было неотображаемых символов. Функции GrdEncode и GrdDecode не изменяют пароль.	GrdAT_AlgoFile	Файловый метод. Кодирование выполняется блоками произвольной длины, поэтому такой метод подходит для кодирования файлов. Кодирование этим методом файла блоками меньше 32 байт неэффективно. Функции GrdEncode и GrdDecode в этом методе преобразования изменяют пароль, поэтому последовательность использования функции GrdDecode должна совпадать с последовательностью использования функции GrdEncode .
GrdAT_Algo0	Базовый метод. Кодирование выполняется блоками по 32 байта. Этот метод лучше всего использовать для преобразования нестроковых данных. Важная особенность метода: если закодировать большой объем памяти (или файл) участками, то декодирование нужно будет обязательно делать точно теми же участками, или же размер участков должен быть кратен 32 байтам. Функции GrdEncode и GrdDecode не изменяют пароль.						
GrdAT_AlgoSCII	Символьный метод. Кодирование выполняется блоками по 32 байта. Этот метод хорош для кодирования строковых данных. Например, если вы хотите закодировать название поля базы данных, указанное в исходном тексте программы, то в некоторых языках программирования будет проблематично создать строки с несимвольными значениями. Символьный метод кодирует строки так, чтобы в них не было неотображаемых символов. Функции GrdEncode и GrdDecode не изменяют пароль.						
GrdAT_AlgoFile	Файловый метод. Кодирование выполняется блоками произвольной длины, поэтому такой метод подходит для кодирования файлов. Кодирование этим методом файла блоками меньше 32 байт неэффективно. Функции GrdEncode и GrdDecode в этом методе преобразования изменяют пароль, поэтому последовательность использования функции GrdDecode должна совпадать с последовательностью использования функции GrdEncode .						
<i>dwAddr</i>	порядковый номер аппаратного алгоритма, который будет использован для преобразования пароля.						
<i>pKeyBuf</i>	буфер, содержащий 32-байтовый пароль для преобразования.						

Возможные ошибки

GrdE_AlgoNotFound	Алгоритм с указанным номером не существует
GrdE_CRCErrorFunc	Ошибка CRC при выполнении алгоритма. Обычно возникает, если длина преобразуемого пароля не совпадает с длиной ответа алгоритма
GrdE_GPis0	Счетчик алгоритма достиг нулевого значения. Результат этого алгоритма больше нельзя получить
GrdE_InvalidCnvType	Указан неверный метод преобразования
	Набор ошибок Guardant API

Функция **GrdCodeInit** выполняет подготовительные действия перед проведением быстрого взаимнообратного преобразования данных. Этот способ преобразования удобно использовать для обработки больших объемов информации (килобайты и мегабайты).

Функция **GrdCodeInit** позволяет преобразовать пароль для его дальнейшего использования в функциях [GrdEncode](#) и [GrdDecode](#). Адрес буфера, содержащего преобразуемый пароль, задает параметр *pKeyBuf*. Длина пароля фиксирована и должна составлять 32 байта. Для преобразования пароля должен использоваться специальный аппаратный алгоритм Stealth I типа **Fast**; его порядковый номер задается в параметре *dwAddr*. В случае успешного выполнения функции по адресу, заданному в *pKeyBuf*, будет помещен преобразованный пароль.

Переменная *dwCnvType* задает метод быстрого взаимнообратного преобразования, который будет использоваться в операциях [GrdEncode](#) и [GrdDecode](#).

C#

```
public static GrdE GrdCodeInit(Handle grdHandle, GrdAT cnvType, uint addr, byte[] key)
```

grdHandle [in]

Тип: [Handle](#)

хэндл, через который будет выполнена данная операция.

cnvType [in]

Тип: GrdAT

Метод быстрого взаимнообратного преобразования. Задается одним из флагов GrdAT.

addr [in]

Тип: uint

Порядковый номер аппаратного алгоритма, который будет использован для преобразования пароля.

key [in]

Тип: byte []

Буфер, который содержит пароль для преобразования размером 32 байта.

Возможные ошибки

GrdE.AlgoNotFound	Алгоритм с указанным номером не существует
GrdE.CRCErrFunc	Ошибка CRC при выполнении алгоритма. Обычно возникает, если длина преобразуемого пароля не совпадает с длиной ответа алгоритма
GrdE.GPis0	Счетчик алгоритма достиг нулевого значения. Результат этого алгоритма больше нельзя получить
GrdE.InvalidCnvType	Указан неверный метод преобразования
	Набор ошибок Guardant API

Метод GrdCodeInit выполняет подготовительные действия перед проведением быстрого взаимнообратного преобразования данных. Этот способ преобразования удобно использовать для обработки больших объемов информации (килобайты и мегабайты).

Метод GrdCodeInit позволяет преобразовать пароль для его дальнейшего использования в функциях GrdEncode и GrdDecode. Адрес буфера, содержащего преобразуемый пароль, задает параметр key. Длина пароля фиксирована и должна составлять 32 байта. Для преобразования пароля должен использоваться специальный аппаратный алгоритм Stealth I типа Fast; его порядковый номер задается в параметре addr. В случае успешного выполнения метода по адресу, заданному в key, будет помещен преобразованный пароль.

Переменная cnvType задает метод быстрого взаимнообратного преобразования, который будет использоваться в операциях GrdEncode и GrdDecode.

Java

```
public static GrdE GrdCodeInit(Handle grdHandle, int cnvType, int addr, byte[] key)
```

grdHandle [in]

Тип: Handle

хэндл, через который будет выполнена данная операция.

cnvType [in]

Тип: int

Метод быстрого взаимнообратного преобразования. Задается одним из флагов GrdAT.

addr [in]

Тип: int

Порядковый номер аппаратного алгоритма, который будет использован для преобразования пароля.

key [in]

Тип: byte []

Буфер, который содержит пароль для преобразования размером 32 байта.

Возможные ошибки

GrdE.AlgoNotFound	Алгоритм с указанным номером не существует
GrdE.CRCErrFunc	Ошибка CRC при выполнении алгоритма. Обычно возникает, если длина преобразуемого пароля не совпадает с длиной ответа алгоритма
GrdE.GPis0	Счетчик алгоритма достиг нулевого значения. Результат этого алгоритма больше нельзя получить
GrdE.InvalidCnvType	Указан неверный метод преобразования
	Набор ошибок Guardant API

Метод **GrdCodeInit** выполняет подготовительные действия перед проведением быстрого взаимного преобразования данных. Этот способ преобразования удобно использовать для обработки больших объемов информации (килобайты и мегабайты).

Метод **GrdCodeInit** позволяет преобразовать пароль для его дальнейшего использования в функциях [GrdEncode](#) и [GrdDecode](#). Адрес буфера, содержащего преобразуемый пароль, задает параметр *key*. Длина пароля фиксирована и должна составлять 32 байта. Для преобразования пароля должен использоваться специальный аппаратный алгоритм Stealth I типа **Fast**; его порядковый номер задается в параметре *addr*. В случае успешного выполнения метода по адресу, заданному в *key*, будет помещен преобразованный пароль.

Переменная *cnvType* задает метод быстрого взаимного преобразования, который будет использоваться в операциях [GrdEncode](#) и [GrdDecode](#).