Дескриптор защищенной ячейки

Защищенная ячейка определяется структурой в памяти электронного ключа, называемой ***дескриптором***. Дескриптор состоит из полей, описывающих тип данных, которые хранятся в защищенной ячейке, ее свойства, состояние, пароли активации/деактивации и на выполнение операций с данными.
Обращение к защищенной ячейке / аппаратному алгоритму производится по ***числовому имени***. Числовое имя – это идентификатор длиной 2 байта, который хранится в специальной таблице числовых имен ячеек и алгоритмов (Algorithm Root Table, ART). Числовое имя позволяет идентифицировать ячейку вне зависимости от того, какую область памяти она занимает, поскольку «физически» ячейки могут располагаться в памяти в произвольном порядке.

Смещение от начала дескриптора	Длинаполя в байтах	Обозначение поля	Описание поля
00h	1	rs_LoFlags	Младший байт флагов, см. Определения **nsafl_xxx**
01h	1	rs_algo	Код типа алгоритма (см. определения **rs_algo_XXXX**)
02h	2	ReservedForEven	Зарезервировано для выравнивания
04h	4	rs_HiFlags	Дополнительные флаги, см. определения **nsafh_xxx**
08h	4	rs_klen	Длина данных ячейки или определителя алгоритма в байтах (**rs_K[]**). Нужно учитывать необходимость оставить 16 свободных байт в конце доступной памяти. Для алгоритмов SHA256 равно 0. Для алгоритмов с загружаемым кодом соответствует размеру структуры **TgrdLoadableCodeData**
0C	4	rs_blen	Размер блока данных для аппаратного алгоритма. Для алгоритма SHA256 равно 0. Для защищенных ячеек и алгоритмов с загружаемым кодом содержимое поля игнорируется
10	8	rs_hash	Зарезервировано. Поле должно быть заполнено нулями
18	4	rs_ActivatePwd	Пароль активации (если установлен флаг nsafl_ActivationSrv)
1C	4	rs_DeactivatePwd	Пароль деактивации (если установлен флаг nsafl_DeactivationSrv)
20	4	rs_ReadPwd	Пароль для чтения полей rs_GP, rs_ErrorCounter, rs_K[] с помощью функции GrdPI_Read (если установлен флаг nsafh_ReadPwd)
24	4	rs_UpdatePwd	Пароль для изменения полей rs_GP, rs_ErrorCounter, rs_K[] с помощью функции GrdPI_Update (если установлен флаг nsafh_UpdateSrv)
28	6	rs_BirthTime	Дата и время автоматической активации ячейки (если установлен флаг nsafh_BirthTime). Дата и время хранятся в структуре TGrdTime. Размер поля равен sizeof(TGrdTime)
2E	6	rs_DeadTime	Дата и время автоматической деактивации ячейки (если установлен флаг nsafh_DeadTime). Дата и время хранятся в структуре TGrdTime. Размер поля равен sizeof(TGrdTime)
34	8	rs_Lifetime	Время, в течение которого ячейка будет оставаться активной после 1-го обращения к ней или вызова алгоритма (если установлен флаг nsafh_Lifetime) Дата и время хранятся в структуре TGrdLifeTime. Размер поля равен sizeof(TGrdLifeTime)
3C	8	rs_FlipTime	Если указан флаг nsafh_FlipTime - алгоритм меняется каждые rs_DaysGap дней, начиная с даты rs_ChangeFlipTimeStart Размер поля равен sizeof(TgrdFlipTime). Для алгоритмов с загружаемым кодом и SHA256 не используется
44	4	rs_GP	Обратный счетчик. Если установлен флаг nsafl_GP_dec, счетчик уменьшается на 1 при каждом вызове алгоритма через функцию GrdTransform(). При использовании других функций (например, GrdHash() или GrdCrypt() на больших объемах данных) счетчик может уменьшаться быстрее, согласно количеству вызовов GrdTransform() внутри вызывающей функции. По достижении счетчиком нулевого значения, алгоритм переходит в деактивированное состояние и перестает преобразовывать данные. При дальнейших обращениях возвращается код ошибки GrdE_GPis0. Значение счетчика можно увеличить, только записав весь дескриптор заново Если установлен флаг nsafl_GP, алгоритм становится зависимым от значения счетчика. См. описание флага. Если ни один из этих флагов не установлен, значение игнорируется.
48	4	rs_ErrorCounter	Допустимое количество ошибок ввода паролей (если установлен хотя бы один из флагов: nsafl_ActivationSrv, nsafl_DeactivationSrv, nsafl_UpdateSrv). Используется только 1 байт этого поля. По достижении счетчиком нуля защищенная ячейка переходит в состояние с неизменяемым статусом. При вводе правильного пароля значение счетчика не восстанавливается
4C	rs_klen	rs_K[]	Данные защищенной ячейки или определитель алгоритма длиной rs_klen. Для алгоритмов с загружаемым кодом в этом поле хранится структура TgrdLoadableCodeData

Поле **rs_LoFlags** содержит младший байт флагов, определяющих свойства защищенной ячейки. Возможна установка следующих флагов (приведенные ниже названия флагов используются в Guardant API):

Имя флага	Значение	Комментарий
nsafl_ID	1	Уникальность алгоритма по ID ключа. При одинаковых определителях алгоритмы в разных ключах кодируют данные по-разному. Флаг работает только для симметричных алгоритмов (AES128 и GSII64), остальными типами алгоритмов не используется
nsafl_GP_dec	2	Уменьшать счетчик GP при каждой обращении к алгоритму. По достижении счетчиком GP 0, алгоритм/ ячейка автоматически деактивируется и при дальнейших обращениях возвращается код ошибки GrdE_GPis0. Для алгоритмов SHA256 флаг игнорируется
nsafl_GP	4	Зависимость преобразования данных от значения счетчика GP. В современных ключах не используется
nsafl_ST_III	8	В современных ключах флаг установлен
nsafl_ActivationSrv	16	Сервис активации доступен
nsafl_DeactivationSrv	32	Сервис деактивации доступен
nsafl_UpdateSrv	64	Сервис изменения данных ячеек rs_K[] по паролю доступен (функция GrdPI_Update поддерживается)
nsafl_InactiveFlag	128	Признак, что в данный момент алгоритм/ячейка деактивирован. Операции GrdTransform, GrdPI_Read, GrdPI_Update недоступны.

Возможность применения флагов свойств в алгоритмах и защищенных ячейках:

Имя флага	AES128, GSII64	ECC160	SHA256	Загружаемыйкод	Ячейка с данными
nsafl_ID	+	-	-	-	-
nsafl_GP_dec	+	+	+	+	-
nsafl_GP	-	-	-	-	-
nsafl_ST_III	+	+	+	+	-
nsafl_ActivationSrv	+	+	+	+	+
nsafl_DeactivationSrv	+	+	+	+	+
nsafl_UpdateSrv	+	+	+	+	+
nsafl_InactiveFlag	+	+	+	+	+
nsafh_ReadSrv	+	+	+	+	+
nsafh_ReadPwd	+	+	+	+	+
nsafh_BirthTime	+	+	+	+	-
nsafh_DeadTime	+	+	+	+	-
nsafh_LifeTime	+	+	+	+	-
nsafh_FlipTime	+	-	-	-	-

Поле **rs_algo** содержит код типа защищенной ячейки.
а) Для Guardant Sign/Time/Net доступны следующие коды типов данных защищенных ячеек:

Имя кода	Значение	Комментарий
	0-4	Зарезервировано

rs_algo_GSII64	5	Симметричный алгоритм кодирования данных.Секретный ключ размером 128 или 256 бит
rs_algo_HASH64	6	Вычисление 64-битного хэша.Секретный ключ размером 128 или 256 бит
rs_algo_RND64	7	Генерация 64-битного случайного числа
rs_algo_PI	8	Защищенная ячейка данных
rs_algo_GSII64_ENCRYPT	10	Аналогичен rs_algo_GSII64, но возможно только зашифрование
rs_algo_GSII64_DECRYPT	11	Аналогичен rs_algo_GSII64, но возможно только расшифрование
rs_algo_ECC160	12	Цифровая подпись по алгоритму ECC 160 бит.Размер определителя при этом 20 байт
rs_algo_AES128	13	AES-128 (режимы шифрования поддерживаются)
rs_algo_SHA256	15	Алгоритм хэширования SHA256. Определитель не содержит данных в поле rs_K[]

б) Для Guardant Code/Code Time допустимы следующие значения:

Имя кода	**Значение**	**Комментарий**
rs_algo_PI	8	Защищенная ячейка, содержащая данные
rs_algo_ECC160	12	Цифровая подпись по алгоритму ECC 160 бит. Размер определителя при этом 20 байт
rs_algo_AES128	13	Симметричное шифрование по AES с длиной ключа 128 бит (режимы шифрования поддерживаются)
rs_algo_LoadableCode	14	Алгоритм, содержащий загружаемый код
rs_algo_SHA256	15	Алгоритм хэширования SHA256. Определительне содержит данных в поле rs_K[]
rs_algo_AES128 Encode	16	Симметричное шифрование по AES с длиной ключа 128 бит. Данные могут только зашифровываться. Возможны режимы шифрования только ECB и CBC
rs_algo_AES128 Decode	17	Симметричное шифрование по AES с длиной ключа 128 бит. Данные могут только расшифровываться. Возможны режимы шифрования только ECB и CBC

Поле **rs_HiFlags** содержит 4 байта флагов, определяющих свойства защищенной ячейки. Возможна установка следующих флагов (приведенные названия флагов используются в Guardant API):

Имя флага	**Значение**	**Комментарий**
nsafh_ReadSrv	1	Сервис чтения данных ячеек rs_K[] доступен (функция GrdPI_Read поддерживается)
nsafh_ReadPwd	2	Чтение осуществляется по паролю rs_ReadPwd
nsafh_BirthTime	4	Включен режим активации в указанное время (хранится в поле rs_BirthTime)
nsafh_DeadTime	8	Включен режим деактивации в указанное время (хранится в поле rs_DeadTime)
nsafh_LifeTime	16	Включен режим деактивации через указанное время после первого обращения к ячейке (оставшееся до деактивации время хранится в ячейке rs_LifeTime) Одновременное использование с флагами nsafh_DeadTime и nsafh_BirthTime не допускается!
nsafh_FlipTime	32	Включен режим автоматического изменения определителя каждые rs_DaysGap дней, начиная с даты rs_ChangeFlipTimeStart. Можно комбинировать с флагами nsafh_DeadTime, nsafh_BirthTime и nsafh_LifeTime. Этот режим поддерживается только для симметричных алгоритмов (AES128 и GSII64).

Поле **rs_klen** содержит размер данных **rs_K[]**, хранящихся в защищенной ячейке (секретного ключа алгоритма) в байтах.
Поле **rs_blen** содержит минимальный размер блока данных для аппаратного алгоритма. Допустимые значения:

Тип аппаратного алгоритма	rs_klen - длина секретного ключа, байт	rs_blen - минимальная длина блока данных, байт
GSII64	GrdADS_GSII64=16/32	GrdARS_GSII64=8
HASH64	GrdADS_HASH64=16/32	GrdARS_HASH64=8
RND64	GrdADS_RAND64=16/32	GrdARS_RAND64=8
AES128	GrdADS_AES128=16	GrdARS_AES128=16
ECC160	GrdADS_ECC160=20	GrdARS_ECC160=20
SHA256	-	GrdARS_HASH_SHA256=0
Загружаемый код	sizeof(TGrdLoadableCodeData)	Не используется

Для алгоритмов GSII64, HASH64 и RND64 при ошибочном значении длины ключа, его длина устанавливается равной 16 байт.
Поле **rs_hash** зарезервировано для будущего использования.
Поле **rs_ActivatePwd** содержит 4-байтовый пароль активации защищенной ячейки, в том случае, если разрешен сервис активации установкой флага **nsafl_ActivationSrv**.
Поле **rs_DeactivatePwd** содержит 4-байтовый пароль деактивации защищенной ячейки, в том случае, если разрешен сервис деактивации установкой флага **nsafl_DeactivationSrv**.
Поле **rs_ReadPwd** содержит 4-байтовый пароль чтения данных защищенной ячейки, в том случае, если разрешен сервис чтения данных по паролю установкой флага **nsafh_ReadPwd**.
Поле **rs_UpdatePwd** содержит 4-байтовый пароль на обновление данных защищенной ячейки в том случае, если разрешен сервис обновления данных по паролю установкой флага **nsafh_UpdateSrv**.
Поле **rs_BirthTime** содержит дату и время автоматической активации алгоритма. Если не установлен флаг **nsafh_BirthTime**, значение поля игнорируется. Дата и время хранятся в структуре **TGrdTime**:

Смещение	Длина	Название	Комментарий
0	BYTE	BSeconds	Секунды от 0 до 59
1	BYTE	BMinute	Минуты от 0 до 59
2	BYTE	BHour	Часы от 0 до 23
3	BYTE	BDay	Дни от 1 до 31
4	BYTE	BMonth	Месяцы от 1 до 12
5	BYTE	BYear	Годы от 0 до 99, начиная с 2000 года

Поле **rs_DeadTime** содержит дату и время автоматической деактивации алгоритма. Если не установлен **nsafh_DeadTime**, значение поля игнорируется. Дата и время хранятся в структуре **TgrdTime** (см. выше).
Поле **rs_LifeTime** содержит время, в течение которого алгоритм будет оставаться в активном состоянии после первого обращения к нему. Если не установлен флаг **nsafh_LifeTime**, значение поля игнорируется. Дата и время хранятся в структуре **TGrdLifeTime**:

Смещение	Длина	Название	Комментарий
0	TGrdTime	LifeTime	Срок жизни аппаратного алгоритма после первого обращения (в виде разницы дат)
6	BYTE	State	0 – обращений к алгоритму еще не было, **это исходное значение**. При предпродажном программировании ключа в поле rs_DeadTime рекомендуется помещать текущее время на момент программирования. 1 – алгоритм уже активировался, это значение устанавливается **автоматически** после первого обращения к алгоритму. В поле rs_DeadTime автоматически записывается время, когда алгоритм должен деактивироваться
7	BYTE	Reserved ForEven	Зарезервировано для выравнивания

Поле **rs_FlipTime** содержит время **rs_ChangeFlipTimeStart**, когда должна начаться автоматическая модификация определителя алгоритма и количество дней **rs_DaysGap**, составляющих период смены.
Если не установлен флаг **nsafh_FlipTime**, значение поля игнорируется. Дата и время хранятся в структуре **TGrdFlipTime**:

Смещение	Длина	Название	Комментарий
0	TGrdTime	rs_ChangeFlipTimeStart	Дата и время, с которого начинается отсчет
6	BYTE	rs_DaysGap	Алгоритм меняется каждые rs_DaysGap дней, начиная с даты rs_ChangeFlipTimeStart. Должно быть значение от 1 до 255
7	BYTE	ReservedForEven	Зарезервировано для выравнивания

Поле **rs_GP** содержит счетчик алгоритма. Если указан флаг **nsaf_GP_dec**, то это поле задает то количество раз, которое алгоритм может быть выполнен. По достижении счетчиком нулевого значения, такой алгоритм переходит в деактивированное состояние и перестает преобразовывать данные. При дальнейших обращениях возвращается код ошибки **GrdE_InactiveItem**. Значение счетчика можно увеличить, только записав весь дескриптор заново.
Поле **rs_ErrorCounter** содержит счетчик неудачных попыток подбора паролей на доступ к защищенной ячейке. При каждой неудачной попытке счетчик уменьшается на 1. По достижении счетчиком нуля защищенная ячейка переходит в состояние с неизменяемым статусом.
Поле **rs_K[]** содержит данные защищенной ячейки. В зависимости от типа ячейки здесь может храниться секретный ключ алгоритма, либо какие-нибудь иные данные. Длина данных обязательно должна соответствовать значению поля **rs_klen**. Данные защищенной ячейки можно считывать или изменять только с помощью специальных функций. Главное отличие защищенной ячейки от простых данных заключается в том, что ее данные можно менять, только указав специальный пароль. Этот пароль программируется разработчиком и хранится в дескрипторе защищенной ячейки. Такой механизм позволяет производить более безопасное изменение части памяти ключа, не затрагивающее остальную память.
В дескрипторах алгоритмов с загружаемым кодом в поле **rs_K[]** должна содержаться структура **TGrdLoadableCodeData**, описывающая свойства модуля загружаемого кода. Эта структура разделена на публичную часть, данные которой можно прочитать, если сервис чтения разрешен, и приватную часть, данные которой закрыты от чтения.

Длина, байт	Имя поля	Комментарий
Публичная часть (структура TGrdCodePublicData), данные доступны для чтения		
1	bLoadableCodeVersion	Версия загружаемого кода
1	bReserved0	Зарезервировано
1	bState	Состояние загружаемого кода (GrdCodeState_XXXXX)
1	bReserved	Зарезервировано
4	dwLoadingDate	Дата загрузки кода. Заполняется автоматически функцией GrdCodeLoad
Закрытая часть структуры, данные не доступны для чтения		
40	abLoadableCodePublicKey4VerifySign	Открытый ключ для проверки подписи кода
20	abLoadableCodePrivateKey4DecryptKey	Закрытый ключ для расшифрования ключа AES, на котором зашифрован загружаемый код
4	dwBegFlashAddr	Начальный адрес Flash-памяти, зарезервированной для загружаемого кода
4	dwEndFlashAddr	Конечный адрес Flash-памяти, зарезервированной для загружаемого кода
4	dwBegMemAddr	Начальный адрес RAM ключа, зарезервированной для загружаемого кода
4	dwEndMemAddr	Конечный адрес RAM ключа, зарезервированной для загружаемого кода