Устройство

Дескриптор

Каждый аппаратный алгоритм описывается дескриптором. Дескрипторы хранятся в памяти ключа в защищенных ячейках специального вида и защищены от чтения и модификации. Часть дескриптора описывает свойства алгоритма. Другая его часть представляет собой определитель аппаратного алгоритма. Он играет важнейшую роль в реализации конкретного алгоритма, будучи секретным ключом преобразования.
Детальную информацию об устройстве дескриптора см. в разделе **Дескриптор защищенной ячейки**.

Свойства и их использование

В процессе создания дескриптора участвует также и комбинация свойств аппаратного алгоритма. Свойства задаются специальными флагами, входящими в состав дескриптора. Задавая то или иное сочетание свойств, можно добиться от аппаратного алгоритма нужного «поведения». Свойства аппаратного алгоритма можно комбинировать в зависимости от необходимости.

Алгоритм с ограниченным числом запусков

Если в дескрипторе алгоритма установлен флаг **nsaf_GP_dec**, перед каждым выполнением этого алгоритма значение его счетчика (4-байтовое поле **km_ad_GP** дескриптора) будет уменьшаться на 1. Когда значение счетчика алгоритма достигнет нулевого значения, алгоритм автоматически деактивируется и перестает преобразовывать данные. Это прекрасный способ создания приложения, защищенного с ограничением количества запусков.
Для обратной активации алгоритма, счетчик выполнений которого достиг нуля, необходимо либо перепрограммировать ключ локально, либо записать новое значение в поле **km_ad_GP** дескриптора этого алгоритма, используя TRU.
Можно рассчитать примерное количество обращений к алгоритму за время предполагаемой жизни приложения (или его очередной версии) и ограничить количество выполнений алгоритма соответствующим числом. Это может служить одним из способов противодействия попыткам брутфорса алгоритма.

Активация и деактивация алгоритма

Если в определителе алгоритма заданы эти свойства, то появляется возможность в нужный момент активировать или деактивировать алгоритм, используя специальную функцию . Для активации и деактивации можно задавать пароли, которые также хранятся в дескрипторе алгоритма. При помощи активации и деактивации можно эффективно управлять конкретным набором дескрипторов алгоритмов, участвующих в работе системы защиты.
Более подробно см. **Активация/деактивация защищенных ячеек**.

Зависимость от ID

Флаг **nsaf_ID** устанавливает зависимость аппаратного алгоритма от идентификационного номера (ID) данного ключа. Это означает, что такой алгоритм в каждом ключе будет преобразовывать данные уникальным образом, даже если все значения его дескриптора будут одинаковыми во всех ключах.
Предупреждение
В случае если ключ с таким аппаратным алгоритмом будет случайно испорчен, его будет невозможно заменить ключом с таким же алгоритмом, т. к. не может существовать двух ключей с одинаковыми ID.

Ограничение алгоритма по времени работы

Электронные ключи Guardant с часами реального времени позволяют управлять активностью аппаратных алгоритмов при помощи таймера реального времени с автономным источником питания. Для этого в дескрипторе аппаратного алгоритма предусмотрены специальные поля, в которых хранятся ограничивающие значения времени.
Более подробно см. **Использование таймера для управления статусом аппаратных алгоритмов**.

Секретный ключ (определитель) алгоритма

Определитель аппаратного алгоритма – это набор байтов, записанных в особом поле дескриптора алгоритма и интерпретируемых микропрограммой как секретный ключ алгоритма.
Определители алгоритмов являются критическими данными, относящимися к защите приложения и всегда должны храниться в секрете. Доступ к ним может осуществлять только авторизованный персонал.
В процессе кодирования/декодирования или вычисления хэш-функции определитель никогда не покидает памяти контроллера. Дескриптор алгоритма, содержащий определитель должен быть защищен от чтения/записи аппаратными запретами.
Для генерации определителей лучше всего пользоваться надежными защищенными алгоритмами генерации случайных чисел, например, аппаратным алгоритмом типа RND64. Не рекомендуется использовать для этой цели встроенные функции языков программирования.
Определители полезно время от времени менять. Это очень хорошая и распространенная практика, повышающая защищенность системы.
Определители алгоритмов желательно менять при выпуске каждой новой версии защищенного приложения.