

РУКОВОДСТВО РАЗРАБОТЧИКА
описание функций API для ключей серии
SenseLock EL

www.senselock.ru
lock@senselock.ru

Оглавление

Описание функций API.....	3
S4Enum().....	4
S4Open().....	6
S4OpenEx().....	7
S4Close().....	9
S4Control().....	10
S4CreateDir().....	13
S4CreateDirEx().....	15
S4ChangeDir().....	18
S4EraseDir().....	19
S4VerifyPin().....	20
S4ChangePin().....	21
S4WriteFile().....	22
PS4WriteFile().....	26
S4Execute().....	29
S4ExecuteEx().....	30
Коды ошибок.....	31

Описание функций API

В комплекте разработчика для ключа SenseLock EL присутствует описание функций и библиотеки для различных языков программирования, таких как C, Delphi, Java и т.д. Если к используемому вами языку библиотека отсутствует, то вы можете использовать динамическую библиотеку dll. Некоторые функции API используются только на стадии проектирования защиты, в то время как другие – для доступа к ключу из программы, находящейся у пользователя.

Никогда не используйте функции, помеченные как "Только для разработчика" в пользовательском приложении!

Таблица 8-1. Назначение функций API

Функция API	Описание	Область использования
S4Enum	Создание списка всех подключенных устройств SenseLock EL	Разработчик, пользователь
S4Open	Подключение к устройству используя указанный индекс.	Разработчик, пользователь
S4OpenEx	Подключение к устройству используя указанный режим.	Разработчик, пользователь
S4Close	Закрытие устройства	Разработчик, пользователь
S4Control	Передача управляющего кода устройству, например для управления LED.	Разработчик, пользователь
S4CreateDir	Создание новой директории в текущей, рекомендуется использовать функцию S4CreateDirEx	Только для разработчика
S4CreateDirEx	Создание новой директории в текущей.	Только для разработчика
S4ChangeDir	Сменить директорию.	Разработчик, пользователь
S4EraseDir	Удалить все файлы и поддиректории в указанной директории.	Только для разработчика
S4VerifyPin	Проверка PIN кода	Разработчик, пользователь
S4ChangePin	Изменение PIN кода	Только для разработчика ¹
S4WriteFile	Создание и обновление файла, создание ключей RSA,	Только для разработчика
S4Execute	Исполнение VM модуля	Только для пользователя
S4ExecuteEx	Исполнение VM и XA модулей	Только для пользователя
PS4WriteFile	Идентична функции S4WriteFile, может работать с файлами напрямую	Только для разработчика

Описание функций API

¹ При необходимости эта функция может быть использована в приложении пользователя для изменения PIN кода пользователя.

S4Enum()

Создание списка всех подключенных устройств SenseLock EL. Информация помещается в структуру S4CtxList.

```
DWORD WINAPI S4Enum(
    SENSE4_CONTEXT *pS4CtxList,
    DWORD *pdwSize);
```

Параметры:

pS4CtxList [out] Указатель на массив структур SENSE4_CONTEXT
pdwSize [in, out] Размер структуры SENSE4_CONTEXT в байтах на входе, размер полученных данных на выходе. Если *pS4CtxList* = NULL или указан размер меньше требуемого, функция вернет ошибку и сохранит в переменную *pdwSize* размер необходимого объема памяти.

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Вызов этой функции должен предшествовать вызову любой другой функции SenseLock EL API. Информация о каждом подключенном ключе SenseLock EL помещается в структуру SENSE4_CONTEXT:

```
typedef struct {
    DWORD dwIndex;
    DWORD dwVersion;
    S4HANDLE hLock;
    BYTE reserve[12];
    BYTE bAttr[MAX_ATR_LEN];
    BYTE bID[MAX_ID_LEN];
    DWORD dwAttrLen;
} SENSE4_CONTEXT, *PSENSE4_CONTEXT;
```

Описание параметров структуры SENSE4_CONTEXT:

<i>dwIndex</i>	Индекс устройства, начинается с 0.
<i>dwVersion</i>	Версия микропрограммы ключа SenseLock EL: <ul style="list-style-type: none"> • SENSE4_CARD_TYPE_V2_00 - версия 2.0 • SENSE4_CARD_TYPE_V2_01 - версия 2.1 • SENSE4_CARD_TYPE_V2_02 - версия 2.2 • SENSE4_CARD_TYPE_V2_30 - версия 2.3
<i>hLock</i>	Дескриптор ключа.
<i>reserve</i>	Зарезервировано системой.
<i>bAttr</i>	Информация ATR, зарезервировано системой.
<i>bID</i>	Идентификатор ключа. Содержит Уникальный Серийный Номер Устройства (HUSN). Если в ключе присутствует ATR файл, то <i>bID</i> будет содержать установленный разработчиком ID (содержимое ATR файла). Более детальную информацию о файле ATR вы можете узнать из описания функции S4CreateDirEx().
<i>dwAttrLen</i>	Длина ATR, зарезервировано системой.

Если вы не знаете необходимый размер для структуры SENSE4_CONTEXT всех подключенных устройств SenseLock EL, то можете установить параметр *pS4CtxList* равным NULL, тогда при выходе из функции параметр *pdwSize* будет содержать размер необходимой памяти.

Чтобы узнать количество подключенных устройств SenseLock EL, можно использовать значение, полученное в переменную *pdwSize*:

```
num = *pdwSize / sizeof(SENSE4_CONTEXT);
```

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

S4Open()

Подключение к электронному ключу SenseLock EL. Подключение должно предшествовать любой попытке доступа к ключу.

```
DWORD WINAPI S4Open (SENSE4_CONTEXT *pS4Ctx);
```

Параметры:

pS4Ctx [in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL.

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Для подключения к устройству SenseLock EL используйте структуру SENSE4_CONTEXT, полученную при вызове функции S4Enum().

По умолчанию подключение к ключу идет в режиме разделяемого доступа (не доступно при использовании HID режима). Более подробно о режимах разделяемого и эксклюзивного доступа можно узнать в описании функции S4OpenEx().

Если в системе присутствует несколько электронных ключей SenseLock EL, то перед использованием функций, необходимо обязательно удостовериться, что было произведено подключение к правильному ключу. В противном случае, подключение к чужому ключу и его использование может привести к некорректной работе как вашего, так и другого защищенного приложения.

Предлагается несколько рекомендаций:

Во-первых, возможно создание индивидуального ID для всей серии ваших ключей и таким образом вы с легкостью сможете отличить "свой" ключ от "чужого". Это достаточно эффективный метод. 8-ми байтовый ID ключа устанавливается посредством внесения изменений в ATR идентификатор при формировании корневого каталога. Когда вы используете функцию S4Enum() для перечисления подключенных к системе ключей, ID каждого из них содержится в структуре SENSE4_CONTEXT. Более детально функциональные особенности установки произвольного ID рассмотрены в описании функции S4CreateDirEx().

Во-вторых, по договоренности с разработчиком в ключи может быть прописан "ID Разработчика". Он никак не зависит от идентификатора, задаваемого при помощи ATR файла, и может быть получен при помощи функции S4Control().

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

S4OpenEx()

Подключение к электронному ключу SenseLock EL. Подключение должно предшествовать любой попытке доступа к ключу. Эта функция является расширенной версией функции S4Open(). Более детальную информацию Вы можете получить ниже, в разделе “Комментарий”.

```
DWORD WINAPI S4OpenEx(
    SENSE4_CONTEXT *pS4Ctx
    S4OPENINFO *pS4OpenInfo);
```

Параметры:

pS4Ctx [in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL

pS4OpenInfo [in] Указатель на структуру S4OPENINFO, описывающую метод открытия ключа

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Основным отличием функции S4OpenEx() от функции S4Open() является возможность включения режима совместного использования ключа несколькими приложениями. Если при использовании функции S4Open() вы можете использовать ключ только тем приложением или модулем, которое произвело операцию открытия, то в случае использования расширенной функции S4OpenEx() возможно использовать ключ несколькими модулями и приложениями.

Описание структуры S4OPENINFO:

```
typedef struct _S4OPENINFO {
    DWORD dwS4OpenInfoSize;
    DWORD dwShareMode;
} S4OPENINFO;
```

dwS4OpenInfoSize	Должен быть равен sizeof(S4OPENINFO)
dwShareMode	Режим подключения ключа, поддерживается 2 режима: <ul style="list-style-type: none"> • S4_EXCLUSIVE_MODE – эксклюзивный режим • S4_SHARE_MODE – общий режим

Для получения более подробной информации смотрите описание функции S4Open().

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

Пример использования:

```
/* Демонстрация работы с ключом SenseLock EL в эксклюзивном режиме.*/
...
SENSE4_CONTEXT s4ctx = {0};
```

```
S4OPENINFO oinf = {0};
DWORD ret = 0;
/* Здесь необходимо использовать функции для перечисления
ключей... */
/* Теперь открываем ключ в эксклюзивном режиме... */
oinf.dwS4OpenInfoSize = sizeof(S4OPENINFO );
oinf.dwShareMode = S4_EXCLUSIVE_MODE;
ret = S4OpenEx(&s4ctx, &oinf);
...
```


S4Close()

Закрывает соединение с ключом.

```
DWORD WINAPI S4Close (SENSE4_CONTEXT *pS4Ctx);
```

Параметры:

pS4Ctx [in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL.

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Эта функция закрывает соединение с ключом, открытое посредством использования функций S4Open() или S4OpenEx(). Важно учитывать, что функция S4Close() не очищает статус безопасности ключа SenseLock EL. Если Вам нужно закрыть ключ и очистить статус безопасности, используйте функцию S4Control() с параметром S4_RESET_DEVICE перед использованием S4Close().

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

S4Control()

Передача в ключ кода управления.

```

DWORD WINAPI S4Control(
    SENSE4_CONTEXT *pS4Ctx,
    DWORD dwCtlCode,
    VOID *pInBuff,
    DWORD dwInBuffLen,
    VOID *pOutBuff,
    DWORD dwOutBuffLen,
    DWORD *pdwBytesReturned);

```

Параметры:

<i>pS4Ctx</i>	[in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL
<i>dwCtlCode</i>	[in] Код управления, значения и описание смотрите в описании
<i>pInBuff</i>	[in] Указатель на буфер, содержащий дополнительные данные, передаваемые в ключ с кодом управления
<i>dwInBuffLen</i>	[in] Размер передаваемых в буфере <i>pInBuff</i> данных
<i>pOutBuff</i>	[out] Указатель на буфер, который будет содержать данные после выполнения функции
<i>dwOutBuffLen</i>	[in] Размер буфера <i>pOutBuff</i>
<i>pdwBytesReturned</i>	[out] Указатель на DWORD, который после выполнения функции будет содержать размер полученных данных

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Ключ SenseLock EL поддерживает следующие коды управления:

- S4_LED_UP – включить светодиод
Вход: нет параметров
Выход: нет параметров
- S4_LED_DOWN – выключить светодиод
Вход: нет параметров
Выход: нет параметров
- S4_LED_WINK – помигать светодиодом
Вход: 1 байт, указывающий на количество переключений состояния светодиода за 1 секунду времени.
Выход: нет параметров
- S4_GET_DEVICE_TYPE – получить тип подключенного ключа
Вход: нет параметров
Выход: 1 байт, показывающий тип подключенного ключа: обычный ключ, сетевой мастер ключ или сетевой клиент ключ.

- S4_GET_SERIAL_NUMBER – получить уникальный серийный номер ключа (HUSN)
Вход: нет параметров
Выход: 8 байт HUSN
- S4_GET_VM_TYPE – получить тип используемой виртуальной машины. Этот код управления не используется.
Вход: нет параметров
Выход: нет параметров
- S4_GET_DEVICE_USABLE_SPACE – получить количество доступной памяти в ключе.
Вход: нет параметров
Выход: 2 байта, количество свободного в ключе места
- S4_SET_DEVICE_ID – установить произвольный ID (идентичен установке ATR файла). Данная функция требует авторизацию уровня разработчика.
Вход: 8 байт ID
Выход: нет параметров
- S4_RESET_DEVICE – сброс ключа. Также сбрасываются все статусы безопасности.
Вход: нет параметров
Выход: нет параметров
- S4_DF_AVAILABLE_SPACE – получить количество доступной памяти в текущем каталоге.
Вход: нет параметров
Выход: 2 байта, количество свободного места в текущем каталоге
- S4_EF_INFO – получить атрибуты файла в текущем каталоге
Вход: ID файла, атрибуты которого запрашиваются
Выход: Структура EFINFO. Более подробное описание структуры смотрите ниже в таблице.
- S4_SET_USB_MODE – установить стандартный режим работы ключа с использованием драйверов.
Вход: нет параметров
Выход: нет параметров
- S4_SET_HID_MODE – установить режим работы ключа в HID (без драйверов). Внимание! Этот режим поддерживается операционной системой Windows 98 и выше.
Вход: нет параметров
Выход: нет параметров
- S4_GET_CUSTOMER_NAME – Получить ID разработчика.
Вход: нет параметров
Выход: 4 байта ID разработчика

- `S4_GET_MANUFACTURE_DATE` – Получить дату создания ключа.
Вход: нет параметров
Выход: Структура `S4_MANUFACTURE_DATE`. Более подробное описание структуры смотрите ниже в таблице.
- `S4_GET_CURRENT_TIME` – Получить из ключа текущее время. Только для модели Senselock EL-RTC
Вход: нет параметров
Выход: Указатель на “tm” структуру. Полностью соответствует стандартной структуре “tm” языка Си. Подробное описание смотрите в документации MSDN.

Структура `EFINFO`:

```
typedef struct{
    WORD EfID;
    BYTE EfType;
    WORD EfSize;
} EFINFO, *PEFINFO;
```

<i>EfID</i>	ID файла
<i>EfType</i>	Тип файла
<i>EfSize</i>	Размер файла

Структура `S4_MANUFACTURE_DATE`:

```
typedef struct {
    WORD wYear;
    BYTE byMonth;
    BYTE byDay;
} S4_MANUFACTURE_DATE;
```

<i>wYear</i>	Год
<i>byMonth</i>	Месяц 1-12
<i>byDate</i>	Число 1-31

Структура `tm`:

```
struct tm {
    int tm_sec; /* секунды - [0,59] */
    int tm_min; /* минуты - [0,59] */
    int tm_hour; /* часы - [0,23] */
    int tm_mday; /* дни - [1,31] */
    int tm_mon; /* месяцы - [0,11] */
    int tm_year; /* года начиная с 1900 */
    int tm_wday; /* дни недели начиная с Воскресенья -
[0,6] */
    int tm_yday; /* день в году, начиная с 1 Января -
[0,365] */
    int tm_isdst; /* флаг перехода на летнее время */
};
```

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

S4CreateDir()

Создаёт новый каталог в ключе. Если создание каталога прошло успешно, то он становится текущим. Рекомендуется использовать расширенную версию функции - S4CreateDirEx().

```
DWORD WINAPI S4CreateDir(
    SENSE4_CONTEXT *pS4Ctx,
    LPCSTR lpszDirID,
    DWORD dwDirSize,
    DWORD dwFlags);
```

Параметры:

- | | |
|------------------|--|
| <i>pS4Ctx</i> | [in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL. |
| <i>lpszDirID</i> | [in] ID каталога. В API ключа SenseLock EL, ID файлов и каталогов всегда передается в виде строки. Например, строка "1234" означает 0x1234. Пожалуйста, не забывайте также, что при указании ID необходимо использовать все 4 символа. То есть нельзя передавать "12" как ID файла или каталога. Необходимо дополнить его двумя нулями вначале "0012". Также, существуют зарезервированные системные ID, которые использовать нельзя, их список смотрите в разделе "Комментарии". |
| <i>dwDirSize</i> | [in] Размер создаваемого каталога. Этот размер не может превышать размер каталога, в котором создаётся новый каталог. Также при создании каталога, файловая система резервирует некоторое количество свободного пространства. Обычные ключи резервируют 134 байта, а сетевые ключи 988 байт. Для примера, если нам необходимо создать каталог размером 1024 байта, то после его создания будет доступно для использования (1024-134)=990 байт для обычных ключей и (1024-988)=36 байт для сетевых ключей. В дополнение к этому, заголовок каталога занимает определенное место (42 байта). Таким образом, при создании каталога длиной 1024 байта, в родительском каталоге свободное пространство уменьшится на (1024+42)=1066 байт. |
| <i>dwFlags</i> | [in] Флаги и опции для создания каталога. <ul style="list-style-type: none"> • S4_CREATE_ROOT_DIR – Создать корневой каталог. При использовании этого флага, необходимо установить ID в "\", а параметр <i>dwDirSize</i> равным 0. • S4_CREATE_SUB_DIR – Создать поддиректорию. Работает только на обычных ключах (не сетевых!). • S4_CREATE_MODULE – Создать подмодуль. Предназначено только для сетевой версии ключей SenseLock. |

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Эта функция НЕ ПОЗВОЛЯЕТ устанавливать произвольный ATR файл и тем самым модифицировать ID ключа. Для того чтобы управлять ATR файлом, Вам необходимо использовать функцию `S4CreateDirEx()`.

После создания каталога PIN код разработчика и PIN код пользователя для созданного каталога устанавливаются в значения по умолчанию.

PIN код разработчика: "123456781234567812345678"

PIN код пользователя : "12345678"

Вам НЕОБХОДИМО изменить PIN код разработчика на более секретное значение!

В сетевой версии ключа SenseLock EL, все PIN коды также принимают значение по умолчанию после создания нового модуля.

Следующие ID зарезервированы операционной системой ключа и не могут быть использованы для создания каталогов и файлов:

0x0000, 0x0015, 0x0016, 0x0018, 0x001e, 0x3f00, 0x3f01, 0x3f02, 0x3f03, 0x3f04.

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

S4CreateDirEx()

Создаёт новый каталог в ключе. Если создание каталога прошло успешно, то он становится текущим. Данная команда рекомендована к использованию вместо S4CreateDir().

```
DWORD WINAPI S4CreateDirEx(
    SENSE4_CONTEXT *pS4Ctx,
    LPCSTR lpszDirID,
    DWORD dwDirSize,
    DWORD dwFlags,
    S4CREATEDIRINFO *pCreateDirInfo);
```

Параметры:

<i>pS4Ctx</i>	[in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL.
<i>lpszDirID</i>	[in] ID каталога. В API ключа SenseLock EL, ID файлов и каталогов всегда передается в виде строки. Например, строка "1234" означает 0x1234. Пожалуйста, не забывайте также, что при указании ID необходимо использовать все 4 символа. То есть нельзя передавать "12" как ID файла или каталога. Необходимо дополнить его двумя нулями вначале "0012". Так же, существуют зарезервированные системные ID, которые использовать нельзя, их список смотрите в разделе "Комментарии".
<i>dwDirSize</i>	[in] Размер создаваемого каталога. Этот размер не может превышать размер каталога, в котором создаётся новый каталог. Также при создании каталога, файловая система резервирует некоторое количество свободного пространства. Обычные ключи резервируют 134 байта, а сетевые ключи 988 байт. Для примера, если нам необходимо создать каталог размером 1024 байта, то после его создания будет доступно для использования (1024-134)=990 байт для обычных ключей и (1024-988)=36 байт для сетевых ключей. Дополнительно, заголовок каталога занимает определенное место (42 байта). Таким образом при создании каталога длиной 1024 байта, в родительском каталоге свободное пространство уменьшится на (1024+42)=1066 байт.
<i>dwFlags</i>	[in] Флаги и опции для создания каталога. S4_CREATE_ROOT_DIR – Создать корневой каталог. При использовании этого флага, необходимо установить ID в "\", а параметр <i>dwDirSize</i> равным 0. <ul style="list-style-type: none"> • S4_CREATE_SUB_DIR – Создать поддиректорию. Работает только на обычных ключах (не сетевых!). • S4_CREATE_MODULE – Создать подмодуль. Предназначено только для сетевой версии ключей SenseLock.
<i>pCreateDirInfo</i>	[in] Указатель на структуру S4CREATEDIRINFO. Вы должны заполнить эту структуру, если <i>dwFlags</i> установлен в значение S4_CREATE_ROOT_DIR.

Возвращаемые значения:

Возвращает `S4_SUCCESS` если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

При использовании этой функции API будет автоматически создан ATR файл, который может быть использован для установки произвольного ID ключа (данные для нового ID содержатся в значении `szArt[8]` структуры `pCreateDirInfo`).

ATR файл – это специальный тип, содержащий только 8 байт данных. Если этот файл создан и существует, то при использовании функции `S4Enum()` для перечисления подключенных ключей, `bID` значение структуры `SENSE4_CONTEXT` будет иметь значение 8 байт ATR файла. Если же Вы хотите получить реальные 8 байт ID ключа (HUSN), Вам необходимо использовать функцию `S4Control()` с параметром `S4_GET_SERIAL_NUMBER`.

Использование ATR файла позволит с лёгкостью различать ключи нескольких разработчиков. Более детальную информацию смотрите в комментариях к функции `S4Enum()`.

После создания каталога PIN код разработчика и PIN код пользователя для созданного каталога устанавливаются в значения по умолчанию.

PIN код разработчика: “123456781234567812345678”

PIN код пользователя : “12345678”

Вам НЕОБХОДИМО изменить PIN код разработчика на более секретное значение! Для более детальной информации пожалуйста обратитесь к главе 1.3.1

В сетевой версии ключа SenseLock EL, все PIN коды также принимают значение по умолчанию после создания нового модуля.

Следующие ID зарезервированы операционной системой ключа и не могут быть использованы для создания каталогов и файлов:

0x0000, 0x0015, 0x0016, 0x0018, 0x001e, 0x3f00, 0x3f01, 0x3f02, 0x3f03, 0x3f04.

Описание структуры S4CREATEDIRINFO

```
typedef struct _S4CREATEDIRINFO {
    DWORD dwS4CreateDirInfoSize;
    BYTE szAtr[8];
} S4CREATEDIRINFO;
```

<code>dwS4CreateDirInfoSize</code>	Размер структуры должен быть равен: <code>sizeof(S4CREATEDIRINFO)</code> .
<code>szAtr</code>	8 байт ATR файла.

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

Пример использования:

```
/* Пример создания корневого каталога с установкой
произвольного ID ключа. */
...
SENSE4_CONTEXT s4ctx = {0};
S4CREATEDIRINFO rinf = {0};
DWORD ret = 0;
```



```
/* Здесь открываем ключ и идентифицируем с PIN кодом
разработчика. Далее удаляем корневой каталог, если он
существует... */
/* Создаём новый корневой каталог... */
rinf.dwS4CreateDirInfoSize = sizeof(S4CREATEDIRINFO);
memcpy(rinf.szAtr, "SenseLock EL", 8);
ret = S4CreateDirEx(&s4ctx, "\\", 0, S4_CREATE_ROOT_DIR,
&rinf);
...
```

S4ChangeDir()

Выбрать новый рабочий каталог.

```
DWORD WINAPI S4ChangeDir(  
    SENSE4_CONTEXT *pS4Ctx,  
    LPCSTR lpszPath);
```

Параметры:

<i>pS4Ctx</i>	[in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL.
<i>lpszPath</i>	[in] Путь к новому рабочему каталогу. Есть два варианта указания этого пути: <ul style="list-style-type: none">• Абсолютный путь, базирующийся от корневого каталога. Для примера "\\0001\0012", "\", "\\0011".• Относительный путь, базирующийся от текущего рабочего каталога. Для примера "0001\0012","00e4".

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Если вы сменили рабочий каталог на уровень выше того каталога, в котором вы находились, вся авторизация будет утеряна. Вам необходимо будет авторизовать доступ снова. Повторная авторизация также требуется в случае повторного выбора основного каталога.

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

S4EraseDir()

Удаляет все содержимое текущего каталога, а также сбрасывает все атрибуты авторизации.

```
DWORD WINAPI S4EraseDir(  
    SENSE4_CONTEXT *pS4Ctx,  
    LPCSTR lpszPath);
```

Параметры:

pS4Ctx [in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL.
lpszPatch [in] Зарезервировано. Должно быть равно NULL.

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Если текущий каталог не является корневым, то эта функция удалит все файлы и каталоги, находящиеся в нем. При этом текущий каталог не будет удалён. PIN код разработчика и PIN код пользователя этого каталога будут переведены в значение по умолчанию.

Если текущий каталог является корневым, то эта функция удалит его и ключ приобретёт статус “чистый”.

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

S4VerifyPin()

Авторизация доступа (уровень разработчика/уровень пользователя) к текущему каталогу ключа посредством PIN кода.

```
DWORD WINAPI S4VerifyPin (
    SENSE4_CONTEXT *pS4Ctx,
    BYTE *pbPin,
    DWORD dwPinLen,
    DWORD dwPinType);
```

Параметры:

<i>pS4Ctx</i>	[in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL.
<i>pbPin</i>	[in] Указатель на PIN код.
<i>dwPinLen</i>	[in] Длина PIN кода. PIN код разработчика – 24 символа. PIN код пользователя – 8 символов.
<i>dwPinType</i>	[in] Тип верификации: <ul style="list-style-type: none"> • S4_DEV_PIN – верификация PIN кода разработчика • S4_USER_PIN – верификация PIN кода пользователя

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Ключ SenseLock EL будет заблокирован после нескольких неудачных попыток авторизоваться с PIN кодом разработчика. Если ключ заблокирован – разблокировать его уже невозможно. Чтобы избежать блокирования – следует знать и учитывать “алгоритм блокировки”. Если есть ошибка в первых 16 символах PIN кода разработчика – блокирование ключа не будет осуществляться. Если ошибка в последних 8-ми символах – ключ будет заблокирован. В случае блокирования ключа будет полезным знать количество оставшихся попыток. В случае ошибки функция будет возвращать результат такого вида: 0x63C[x], где X-оставшееся количество попыток авторизации. Данное значение может быть от 0 до F. Например, если функция вернула 0x63CA, то это означает, что осталось 0x0A (10) попыток.

Независимо от количества некорректных попыток использовать пользовательский PIN код, ключ заблокирован не будет.

Все PIN коды эффективны только для текущего рабочего каталога. Не забывайте, что у каждого каталога есть свой пользовательский PIN код и свой PIN код разработчика.

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

S4ChangePin()

Поменять PIN код разработчика или пользователя для текущего рабочего каталога.

```

DWORD WINAPI S4ChangePin (
    SENSE4_CONTEXT *pS4Ctx,
    BYTE *pbOldPin,
    DWORD dwOldPinLen,
    BYTE *pbNewPin,
    DWORD dwNewPinLen,
    DWORD dwPinType);

```

Параметры:

<i>pS4Ctx</i>	[in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL.
<i>pbOldPin</i>	[in] Указатель на старый PIN код. Если <i>dwPinType</i> установлен в значение S4_AUTHEN_PIN, этот параметр должен быть равен NULL.
<i>dwOldPinLen</i>	[in] Длина старого PIN кода. PIN код разработчика равен 24 символам. PIN код пользователя равен 8 символам.
<i>pbNewPin</i>	[in] Указатель на новый PIN.
<i>dwNewPinLen</i>	[in] Длина нового PIN кода. PIN код разработчика равен 24 символам. PIN код пользователя равен 8 символам.
<i>dwPinType</i>	[in] Тип изменяемого PIN кода: <ul style="list-style-type: none"> • S4_DEV_PIN – PIN код разработчика • S4_USER_PIN – PIN код пользователя • S4_AUTHEN_PIN – PIN код сетевого ключа

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Данная функция автоматически осуществляет проверку старого PIN кода перед его сменой. Действуют все правила авторизации, описанные для функции S4VerifyPin(). Более подробную информацию вы можете получить в разделе “Комментарий” функции S4VerifyPin().

После смены PIN кода, произойдёт автоматическая авторизация ваших прав в соответствии с типом изменяемого кода. Смена пользовательского PIN кода возможна только при авторизации пользовательским кодом. Смена PIN кода разработчика возможна только с авторизацией кодом разработчика.

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

S4WriteFile()

Записать либо обновить файл в ключе. Производит действие в текущем рабочем каталоге.

```
DWORD WINAPI S4WriteFile (
    SENSE4_CONTEXT *pS4Ctx,
    LPCSTR lpzFileID,
    DWORD dwOffset,
    LPVOID lpBuffer,
    DWORD dwBufferSize,
    DWORD dwFileSize,
    DWORD *pdwBytesWritten,
    DWORD dwFlags,
    BYTE bFileType);
```

Параметры:

<i>pS4Ctx</i>	[in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL.
<i>lpzFileID</i>	[in] ID файла. Есть несколько исключений касательно использования этого параметра, если флаг <i>dwFlags</i> имеет одно из следующих значений: <ul style="list-style-type: none"> • S4_KEY_GEN_RSA_FILE Вам необходимо указать 8 символьный ID. То есть ID для публичного ключа и ID для приватного ключа RSA • S4_SET_LICENCES Должно быть равным NULL.
<i>dwOffset</i>	[in] Смещение в файле. Если флаг <i>dwFlags</i> установлен в одно из перечисленных ниже значений, этот параметр должен быть равен 0 : <ul style="list-style-type: none"> • S4_KEY_GEN_RSA_FILE • S4_SET_LICENCES • S4_RSA_PUBLIC_FILE • S4_RSA_PRIVATE_FILE
<i>lpBuffer</i>	[in] Указатель на буфер с данными, записываемыми в ключ. Пожалуйста ознакомьтесь с типами данных и их форматом в одной из таблиц раздела “Комментарий”. Если флаг <i>dwFlags</i> имеет одно из перечисленных ниже значений, этот параметр должен быть равен NULL: <ul style="list-style-type: none"> • S4_KEY_GEN_RSA_FILE • S4_SET_LICENCES Если флаг <i>dwFlags</i> имеет одно из перечисленных ниже значений, то данные должны находиться в строго определённом формате. Более подробно о формате данных Вы можете ознакомиться в разделе “Комментарий”. <ul style="list-style-type: none"> • S4_RSA_PUBLIC_FILE • S4_RSA_PRIVATE_KEY
<i>dwBufferSize</i>	[in] Размер данных, записываемых в ключ. Если флаг <i>dwFlags</i> имеет значение: <ul style="list-style-type: none"> • S4_KEY_GEN_RSA_FILE <i>dwBufferSize</i> = 0

	<ul style="list-style-type: none"> • S4_SET_LICENCES <i>dwBufferSize</i> = 0 • S4_RSA_PUBLIC_FILE <i>dwBufferSize</i> = sizeof(S4_RSA_PUBLIC_KEY) • S4_RSA_PRIVATE_FILE <i>dwBufferSize</i> = sizeof(S4_RSA_PRIVATE_KEY)
<i>dwFileSize</i>	[in] Размер файла, который будет создан в ключе. Если <i>dwFlags</i> имеет значение: <ul style="list-style-type: none"> • S4_CREATE_NEW <i>dwFileSize</i> равен размеру создаваемого файла. Пожалуйста не забудьте что размер публичного ключа RSA равен 136 байт, а размер приватного ключа RSA равен 330. <ul style="list-style-type: none"> • S4_SET_LICENCES <i>dwFileSize</i> означает количество лицензий и должно иметь значение от 1 до 64
<i>pdwByteWritten</i>	[out] Размер данных, которые были записаны (в байтах).
<i>dwFlags</i>	[in] Флаг дополнительных опций. Для более детальной информации, пожалуйста, смотрите раздел “Комментарий”.
<i>bFileType</i>	[in] Тип записываемого файла. Для более детальной информации, пожалуйста, смотрите раздел “Комментарий”.

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Возможные значения параметра *dwFlags*:

S4_CREATE_NEW	Создать новый файл. По умолчанию атрибуты нового создаваемого файла будут установлены в значение “чтение/запись”.
S4_FILE_EXECUTE_ONLY	Используется вместе с S4_CREATE_NEW и создаёт файл с атрибутом “только выполнение”. Используется только для исполняемых файлов. Файл с таким атрибутом не может быть прочитан или дописан. Он может быть только исполнен.
S4_CREATE_PEDDING_FILE	Для Senselock EL с версией микропрограммы 2.3 и выше, этот флаг не используется.
S4_UPDATE_FILE	Обновить существующий файл.
S4_KEY_GEN_RSA_FILE	Создать пару ключей RSA (публичный и приватный ключ). Данный параметр может быть использован вместе с S4_CREATE_NEW.
S4_SET_LICENCES	Установить номер лицензии для сетевой версии ключа.

Флаги “чтение/запись” и “только выполнение” обозначают что в режиме “чтение/запись” файл может быть прочитан или записан внутренними SES функциями `_read()` и `_write()`. В случае флага “только выполнение”, данные операции над файлом невозможны.

Типы файлов *bFileType*:

S4_EXE_FILE	Исполняемый файл виртуальной машины VM51, созданный компиляторами Keil или SDCC. Действия: Записывает данные из <i>lpBuffer</i> в файл ключа SenseLock.
S4_XA_EXE_FILE	Исполняемый файл пользовательского уровня в XA формате, созданный компилятором RKit. Действия: Записывает данные из <i>lpBuffer</i> в файл ключа SenseLock.
S4_DATA_FILE	Файл данных. Действия: Записывает данные из <i>lpBuffer</i> в файл ключа SenseLock.
S4_RSA_PUBLIC_FILE	Публичный ключ RSA * Действия: Записывает данные из <i>lpbuffer</i> в файл объекта после преобразования. Более детальную информацию смотрите в комментариях после таблицы.
S4_RSA_PRIVATE_FILE	Приватный ключ RSA. Действия: Записывает данные из <i>lpbuffer</i> в файл объекта после преобразования. Более детальную информацию смотрите в комментариях после таблицы.
0	Если <i>dwFlags</i> равен S4_KEY_GEN_RSA_FILE или S4_SET_LICENCES, то <i>bFileType</i> должен быть равен 0

* Когда записывается файл публичного или приватного RSA ключа, API не пишет данные в ключ непосредственно в бинарном формате, данные предварительно конвертируются в необходимый формат. Структура публичного и приватного ключа (S4_RSA_PUBLIC_KEY и S4_RSA_PRIVATE_KEY) объявлена в файле-заголовке. Остальные типы файлов передаются в ключ без конвертирования, в бинарном формате.

Внимание! Вы не можете удалять созданные в ключе файлы обычными средствами (функции удаления файла не предусмотрены API). Если вы хотите удалить файл и освободить занимаемое им место – вам необходимо удалить каталог со всеми файлами в нём, а потом восстановить нужные.

Следующие ID зарезервированы операционной системой ключа и не могут быть использованы пользователем для создания каталогов и файлов:

0x0000, 0x0015, 0x0016, 0x0018, 0x001e, 0x3f00, 0x3f01, 0x3f02, 0x3f03, 0x3f04.

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

Пример использования:

```
/* Этот пример демонстрирует создание файла данных размером 1 килобайт и запись в него некоторой информации. */
```

.....

```
SENSE4_CONTEXT s4ctx = {0};
unsigned char buff[] = "This is a test message!";
DWORD dwRet = 0;
DWORD dwBytesWritten = 0;
```



```
/* Отрываем ключ и проверяем PIN код разработчика... */

/* Теперь создаём новый файл данных и записываем его. */
dwRet = S4WriteFile(
    &s4ctx,
    "d001",          /* ID нового файла. */
    0,              /* Смещение. */
    buff,           /* Данные для записи. */
    sizeof(buff),  /* Размер данных. */
    1024,          /* Размер создаваемого в ключе файла,
                   устанавливаем его в значение 1024 байт.
                   */
    &dwBytesWritten,
    S4_CREATE_NEW, /* Флаг создания нового файла. */
    S4_DATA_FILE); /* Тип файла. */

if (dwRet != S4_SUCCESS)

{
/* Обрабатываем ошибки здесь... */
}
.....
```

PS4WriteFile()

Записать либо обновить файл в ключе. Производит действие в текущем рабочем каталоге. Эта функция работает напрямую с файловой системой и не требует предварительных операций по созданию буфера с данными в памяти. Вы можете указывать просто путь к файлам в формате .HEX или .BIN. Функция PS4WriteFile() произведет необходимое преобразование файла и вызовет функцию S4WriteFile() для записи данных в ключ. Данная функция может вызываться только из DLL библиотеки. Заголовок функции объявлен в файле “psense4.h”.

```

DWORD WINAPI PS4WriteFile (
    SENSE4_CONTEXT *pS4Ctx,
    LPCSTR lpszFileID,
    LPCSTR lpszPCFilePath,
    DWORD *dwFileSize,
    DWORD dwFlags,
    DWORD dwFileType,
    DWORD *pdwBytesWritten);

```

Параметры:

<i>pS4Ctx</i>	[in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL.
<i>lpszFileID</i>	[in] ID файла.
<i>lpszPCFilePath</i>	[in] Путь к файлу, содержащему данные, предназначенные для записи в ключ. Информацию о форматах файлов вы можете получить в разделе “Комментарий”. Если Вы просто хотите создать файл без записи в него каких-либо данных, этот параметр необходимо установить в значение NULL.
<i>dwFileSize</i>	[in, out] Размер файла, который будет создан в ключе. Если вы хотите чтобы размер файла в ключе совпадал с размером файла, содержащего данные – то установить этот параметр в 0. Если вы также хотите обновить информацию в уже существующем файле – этот параметр установите равным 0.
<i>dwFlags</i>	[in] Флаг дополнительных опций. Для более детальной информации смотрите раздел “Комментарий”.
<i>bFileType</i>	[in] Тип записываемого файла. Для более детальной информации смотрите раздел “Комментарий”.
<i>pdwByteWritten</i>	[out] Размер данных, которые были записаны в ключ (в байтах).

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Возможные значения параметра *dwFlags*:

S4_CREATE_NEW	Создать новый файл. По умолчанию атрибуты нового создаваемого файла будут установлены в значение “чтение/запись”.
---------------	---

S4_FILE_EXECUTE_ONLY	Используется вместе с S4_CREATE_NEW и создаёт файл с атрибутом “только выполнение”. Используется только для исполняемых файлов. Файл с таким атрибутом не может быть прочитан или дописан. Он может быть только исполнен.
S4_CREATE_PEDDING_FILE	Для SenseLock EL с версией микропрограммы 2.3 и выше, этот флаг не используется.
S4_UPDATE_FILE	Обновить существующий файл.
S4_KEY_GEN_RSA_FILE	Создать пару ключей RSA (публичный и приватный ключ). Данный параметр может быть использован вместе с S4_CREATE_NEW.
S4_SET_LICENCES	Установить номер лицензии для сетевой версии ключа.

Флаги “чтение/запись” и “только выполнение” обозначают, что в режиме “чтение/запись” файл может быть прочитан или записан внутренними SES функциями `_read()` и `_write()`. В случае флага “только выполнение”, данные операции над файлом невозможны.

Типы файлов *bFileType*:

S4_EXE_FILE	Исполняемый файл виртуальной машины VM51, созданный компилятором Keil или SDCC. Действия: Записывает данные из <i>lpBuffer</i> в файл ключа SenseLock.
S4_XA_EXE_FILE	Исполняемый файл пользовательского уровня в XA формате, созданный компилятором RKit. Действия: Записывает данные из <i>lpBuffer</i> в файл ключа SenseLock.
S4_HEX_FILE	Исполняемый файл виртуальной машины VM51, созданный компилятором Keil или SDCC в HEX формате. Действия: Записывает данные из <i>lpBuffer</i> в файл ключа SenseLock после конвертирования.
S4_XA_HEX_FILE	Исполняемый файл пользовательского уровня в XA формате, созданный компилятором RKit в HEX формате. Действия: Записывает данные из <i>lpBuffer</i> в файл ключа SenseLock после конвертирования.
S4_DATA_FILE	Файл данных. Действия: Записывает данные из <i>lpBuffer</i> в файл ключа SenseLock.
S4_RSA_PUBLIC_FILE	Публичный ключ RSA. Действия: Записывает данные из <i>lpBuffer</i> в файл ключа SenseLock после конвертирования. Более детальную информацию смотрите в комментариях после таблицы.
S4_RSA_PRIVATE_FILE	Приватный ключ RSA. Действия: Записывает данные из <i>lpBuffer</i> в файл ключа SenseLock после конвертирования. Более детальную информацию смотрите в комментариях после таблицы.

Внимание! Вы не можете удалять созданные в ключе файлы обычными средствами (функции удаления файла не предусмотрены API). Если вы хотите удалить файл и освободить занимаемое им место – вам необходимо удалить каталог со всеми файлами в нём, а потом восстановить нужные.

Следующие ID зарезервированы операционной системой ключа и не могут быть использованы пользователем для создания каталогов и файлов:

0x0000, 0x0015, 0x0016, 0x0018, 0x001e, 0x3f00, 0x3f01, 0x3f02, 0x3f03, 0x3f04.

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

Пример использования:

```
/* Этот пример демонстрирует, как созданный исполняемый XA
файл в формате HEX записывается в ключ. Записываемый файл
находится в каталоге "c:\demo\demo1.hex" */
```

.....

```
SENSE4_CONTEXT s4ctx = {0};
```

```
DWORD dwRet = 0;
```

```
DWORD dwBytesWritten = 0;
```

```
DWORD dwFileSize = 0;
```

```
/* Отрываем ключ и проверяем PIN код разработчика... */
```

```
/* Теперь открываем наш новый исполняемый файл и записываем
его в ключ. */
```

```
dwRet = PS4WriteFile(
    &s4ctx,
    "E001",
    "c:\\demo\\demo1.hex", /* Путь до файла. */
    &dwFileSize,           /* 0, автоматически определить
                           размер файла. */
    S4_CREATE_NEW | S4_FILE_EXECUTE_ONLY,
                           /* Устанавливаем атрибут
                           "только выполнение". */
    S4_XA_HEX_FILE,      /* Тип файла. */
    &dwBytesWritten);
```

```
if (dwRet != S4_SUCCESS)
```

```
{
/* Обрабатываем ошибки здесь... */
}
```

.....

S4Execute()

Запустить на выполнение исполняемый файл в ключе. Позволяет запускать только файлы формата VM51. Производит действие с файлами в текущем рабочем каталоге.

```

DWORD WINAPI S4Execute (
    SENSE4_CONTEXT *pS4Ctx,
    LPCTSTR lpszFileID,
    LPVOID lpInBuffer,
    DWORD dwInBufferSize,
    LPVOID lpOutBuffer,
    DWORD dwOutBufferSize,
    DWORD *dwBytesReturned) ;

```

Параметры:

<i>pS4Ctx</i>	[in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL.
<i>lpszFileID</i>	[in] ID выполняемого файла.
<i>lpInBuffer</i>	[in] Буфер, содержащий данные, которые будут переданы исполняемому файлу.
<i>dwInBufferSize</i>	[in] Размер передаваемого буфера.
<i>lpOutBuffer</i>	[out] Буфер, в который будут переданы данные из ключа после выполнения программы.
<i>dwOutBufferSize</i>	[in] Размер буфера, принимающего данные из ключа.
<i>dwBytesReturned</i>	[out] Размер данных, полученных из ключа.

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Перед использованием этой функции необходимо произвести верификацию пользовательского PIN кода. Эта функция позволяет выполнять ТОЛЬКО исполняемые модули в режиме виртуальной VM51 машины. Для того чтобы выполнить исполняемый модуль в режиме ХА, необходимо использовать функцию S4ExecuteEx().

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

S4ExecuteEx()

Запустить на выполнение исполняемый файл в ключе. Это расширенная версия функции S4Execute(). Позволяет исполнять модули в формате VM51 и ХА. Производит действие с файлами в текущем рабочем каталоге.

```

DWORD WINAPI S4ExecuteEx (
    SENSE4_CONTEXT *pS4Ctx,
    LPCTSTR lpszFileID,
    DWORD dwFlag,
    LPVOID lpInBuffer,
    DWORD dwInBufferSize,
    LPVOID lpOutBuffer,
    DWORD dwOutBufferSize,
    DWORD *dwBytesReturned);

```

Параметры:

<i>pS4Ctx</i>	[in, out] Указатель на структуру SENSE4_CONTEXT подключаемого устройства SenseLock EL.
<i>lpszFileID</i>	[in] ID исполняемого модуля.
<i>dwFlag</i>	[in] Тип исполняемого модуля: <ul style="list-style-type: none"> • S4_VM_EXE – исполняемый модуль VM51 • S4_XA_EXE – исполняемый модуль ХА
<i>lpInBuffer</i>	[in] Буфер, содержащий данные, которые будут переданы исполняемому файлу.
<i>dwInBufferSize</i>	[in] Размер передаваемого буфера.
<i>lpOutBuffer</i>	[out] Буфер, в который будут переданы данные из ключа после выполнения программы.
<i>dwOutBufferSize</i>	[in] Размер буфера, принимающего данные из ключа.
<i>dwBytesReturned</i>	[out] Размер данных, полученных из ключа

Возвращаемые значения:

Возвращает S4_SUCCESS если функция отработала без ошибок или соответствующий код ошибки.

Комментарий:

Перед использованием этой функции необходимо произвести верификацию пользовательского PIN кода. За исключением возможности запускать на выполнение модулей в ХА режиме, эта функция полностью идентична S4Execute().

Требования:

Версия микропрограммы ключа SenseLock EL 2.x

Пример использования:

Коды ошибок

Название ошибки	Значение	Комментарий
S4_SUCCESS	0x00000000	Успешное выполнение
S4_UNPOWERED	0x00000001	Устройство выключено
S4_INVALID_PARAMETER	0x00000002	Ошибка параметра
S4_COMM_ERROR	0x00000003	Ошибка передачи данных, или нет ответа от ключа
S4_PROTOCOL_ERROR	0x00000004	Ошибка протокола передачи данных
S4_DEVICE_BUSY	0x00000005	Ключ занят
S4_KEY_REMOVED	0x00000006	Ключ отсоединён или не подключен
S4_INSUFFICIENT_BUFFER	0x00000011	Недостаточный размер буфера
S4_NO_LIST	0x00000012	Ключ не найден
S4_GENERAL_ERROR	0x00000013	Указывает на нехватку памяти
S4_UNSUPPORTED	0x00000014	Функция не поддерживается или файловая система не может быть создана.
S4_DEVICE_TYPE_MISMATCH	0x00000020	Тип ключа не верный
S4_FILE_TYPE_MISMATCH	0x00006981	Тип файла не верный
S4_FILE_SIZE_CROSS_7FFF	0x00000021	Только для версий v2.2 и ниже
S4_CURRENT_DF_ISNOT_MF	0x000000201	Только в версиях сетевого ключа.
S4_INAVAILABLE_MODULE_DF	0x000000202	Только в версиях сетевого ключа.
S4_FILE_SIZE_TOO_LARGE	0x000000203	Файл слишком длинный. Больше 0x7FFF байт.
S4_DF_SIZE	0x000000204	Размер указанного каталога не достаточный.
S4_DIRECTORY_EXIST	0x00006901	Каталог уже существует
S4_INSUFFICIENT_SECU_STATE	0x00006982	Ошибка статуса безопасности. Необходимо верифицировать PIN код.
S4_PIN_BLOCK	0x00006983	Ключ заблокирован.
S4_APPLICATION_TEMP_BLOCK	0x00006985	Приложение временно заблокировано.
S4_FILE_EXIST	0x00006a80	Файл уже существует
S4_DEVICE_UNSUPPORTED	0x00006a81	Не поддерживается этим ключом.
S4_FILE_NOT_FOUND	0x00006a82	Файл не найден
S4_INSUFFICIENT_SPACE	0x00006a84	Недостаточно свободного места
S4_OFFSET_BEYOND	0x00006B00	Указанное смещение превышает границу файла.
S4_CRYPTO_KEY_NOT_FOUND	0x00009403	Не найден криптографический ключ.
S4_APPLICATION_PERM_BLOCK	0x00009303	Каталог заблокирован.
S4_DATA_BUFFER_LENGTH_ERROR	0x00006700	Неправильный размер буфера данных.
S4_CODE_RANGE	0x00010000	Выход за пределы области кода. Переполнение стека.
S4_CODE_RESERVED_INST	0x00020000	Ошибка выполнения инструкции. Неизвестный опкод.

S4_CODE_RAM_RANGE	0x00040000	Ошибка указателя на внутреннюю память ключа.
S4_CODE_BIT_RANGE	0x00080000	Некорректное значение бита.
S4_CODE_SFR_RANGE	0x00100000	Некорректное значение SFR
S4_CODE_XRAM_RANGE	0x00200000	Ошибка указателя на внешнюю память ключа.
S4WF_INVALID_S4CONTEXT	0x00000101	Ошибка параметра S4Context. Функция PS4WriteFile().
S4WF_INVALID_FILE_ID	0x00000102	Ошибка ID файла. Функция PS4WriteFile().
S4WF_INVALID_PC_FILE	0x00000103	Ошибка файла (PC) Функция PS4WriteFile().
S4WF_INVALID_FLAGS	0x00000104	Ошибка флага. Функция PS4WriteFile().
S4WF_INVALID_FILE_SIZE	0x00000105	Ошибка размера файла. Функция PS4WriteFile().
S4WF_INVALID_FILE_TYPE	0x00000106	Ошибка типа файла. Функция PS4WriteFile().
S4_ERROR_UNKNOWN	0xffffffff	Неизвестная ошибка.