ИКАО

Doc 9303

Машиносчитываемые проездные документы

Издание восьмое, 2021

Часть 11. Механизмы защиты МСПД

Утверждено и опубликовано с санкции Генерального секретаря

Международная организация гражданской авиации

ИКАО

Doc 9303

Машиносчитываемые проездные документы

Издание восьмое, 2021

Часть 11. Механизмы защиты МСПД

Утверждено и опубликовано с санкции Генерального секретаря

Международная организация гражданской авиации

Опубликовано отдельными изданиями на русском, английском,
арабском, испанском, китайском и французском языках
МЕЖДУНАРОДНОЙ ОРГАНИЗАЦИЕЙ ГРАЖДАНСКОЙ АВИАЦИИ.
999 Robert-Bourassa Boulevard, Montréal, Quebec, Canada H3C 5H7

Загрузить и получить дополнительную информацию можно на сайте
www.icao.int/Security/FAL/TRIP.

Doc 9303. ***Машиносчитываемые проездные документы***
Часть 11. ***Механизмы защиты МСПД***
Номер заказа: 9303P11
ISBN 978-92-9265-499-3 (бумажная копия)
ISBN 978-92-9275-568-3 (электронная копия)

ПОПРАВКИ

Об издании поправок регулярно сообщается в дополнениях к Каталогу *Продукции и услуг ИКАО;* Каталог и дополнения размещены на веб-сайте ИКАО www.icao.int. Ниже приведена таблица для регистрации поправок и исправлений.

РЕГИСТРАЦИЯ ПОПРАВОК И ИСПРАВЛЕНИЙ

ПОПРАВКИ		
№	Дата	Кем внесено
1	14/6/24	ИКАО

ИСПРАВЛЕНИЯ		
№	Дата	Кем внесено

Употребляемые обозначения и изложение материала в данном издании не означают выражения со стороны ИКАО какого бы то ни было мнения относительно правового статуса страны, территории, города или района, или их властей, или относительно делимитации их границ.

ОГЛАВЛЕНИЕ

Страница

1. СФЕРА ПРИМЕНЕНИЯ

Часть 11 документа Doc 9303 содержит спецификации, позволяющие государствам и поставщикам реализовать элементы криптографической защиты электронных машиносчитываемых проездных документов (электронные МСПД) для обеспечения доступа к бесконтактной интегральной схеме (ИС). В данной части содержатся криптографические протоколы для:

- предотвращения скимминга данных с бесконтактной ИС;
- предотвращения перехвата обмена информацией между бесконтактной ИС и считывающим устройством;
- обеспечения аутентификации данных, хранящихся на бесконтактной ИС, на основе использования инфраструктуры открытых ключей (PKI), описанной в части 12;
- обеспечения аутентификации самой бесконтактной ИС.

В восьмом издании документа Doc 9303 содержатся спецификации, касающиеся факультативных приложений "Зарегистрированные данные о поездках, зарегистрированные данные о визах и дополнительная биометрическая идентификация" (известных как приложения LDS2), которые представляют собой факультативное расширение информации, содержащейся в электронных МСПД. В настоящую часть документа Doc 9303 включены необходимые протоколы расширенного контроля доступа, призванные обеспечить защиту записи и считывание данных соответствующих приложений LDS2. Эти протоколы контроля доступа могут также использоваться для защиты дополнительных биометрических данных приложения электронного МСПД.

Аутентификация данных, хранящихся на бесконтактной ИС, является основным элементом защиты, позволяющим использовать ИС для ручной и/или автоматизированной проверки. Таким образом, указанный элемент является ОБЯЗАТЕЛЬНЫМ.

Внедрение протокола для предотвращения скимминга данных, хранящихся на бесконтактной ИС, и предотвращения перехвата при обмене информацией между ИС и терминальным устройством носит ОБЯЗАТЕЛЬНЫЙ характер.

Внедрение других протоколов является ФАКУЛЬТАТИВНЫМ, что позволяет государству или организации выдачи определиться с необходимым набором элементов защиты в соответствии с национальными нормативными положениями/требованиями.

Данная часть рассматривается совместно со следующими частями документа Doc 9303:

- часть 1 *"Введение"*;
- часть 10 *"Логическая структура данных (LDS) для хранения биометрических и других данных на бесконтактной интегральной схеме (ИС)"*;
- часть 12 *"Инфраструктура открытых ключей для МСПД"*.

2. ДОПУЩЕНИЯ И ОБОЗНАЧЕНИЯ

Предполагается, что читатель знаком с концепциями и механизмами, предоставляемыми криптографией с открытым ключом и инфраструктурами открытых ключей.

Хотя использование техники криптографии с открытым ключом усложняет введение электронных МСПД, такая техника полезна тем, что она предоставляет в распоряжение пунктов пограничного контроля дополнительное средство установления подлинности электронного МСПД. Предполагается, что ее использование не является единственной мерой установления аутентичности и НЕ СЛЕДУЕТ полагаться не нее как на единственный определяющий фактор.

В случае невозможности использования данных с бесконтактной ИС, например в результате отзыва сертификата или недействительной верификации подписи, или если бесконтактная ИС была умышленно оставлена пустой (см. раздел 4.5.4 части 10 документа Doc 9303), электронный МСПД вовсе не обязательно становится недействительным. В таком случае принимающее государство МОЖЕТ полагаться на другие элементы защиты документа в целях валидации.

2.1 Требования к чипам электронных МСПД и терминалам

В настоящей части документа Doc 9303 содержатся требования к внедрению чипов электронных МСПД (что тоже самое, что ИС) и терминалов (или систем проверки). В то время как чипы электронных МСПД должны удовлетворять этим требованиям согласно терминологии, описанной в части 1 документа Doc 9303, требования к терминалам должны интерпретироваться как рекомендации, т. е. интероперабельность чипа электронного МСПД и терминала гарантируется только в том случае, когда терминал удовлетворяет этим требованиям, в противном случае никакого взаимодействия с чипом электронного МСПД не произойдет либо поведение чипа электронного МСПД будет непредсказуемым. В принципе чипу электронного МСПД нет необходимости навязывать терминалам какие-либо требования, если только напрямую не затрагиваются аспекты защиты чипа электронного МСПД.

2.2 Обозначения

Для обозначения криптографических примитивов независимым от алгоритмов способом используются следующие обозначения:

- шифрование открытого текста S с помощью симметричного ключа K: **E**(K, S);
- дешифрование зашифрованного текста C с помощью симметричного ключа K: **D**(K, C);
- операция расчета хэша применительно к сообщению *m* обозначается как **H***(m)*.
- расчет кода аутентификации сообщения с помощью симметричного ключа K применительно к сообщению M: **MAC**(K,M);
- согласование ключей на основе пар асимметричных ключей (SK, PK) и (SK', PK') и параметров домена D: **KA**(SK,PK',D) / **KA**(SK',PK,D);
- выработка ключей из совместно используемого секретного S: **KDF**(S);
- подписание сообщения *m* с помощью закрытого ключа SK_{IFD} обозначается как: *s* = **Sign**(SK_{IFD}, *m*);

- проверка итоговой подписи *s* с помощью открытого ключа PK_{IFD} и сообщение *m:* = **Verify**(PK_{IFD}, *s*, *m*);

- расчет сжатого представления открытого ключа *PK*: **Comp**(*PK*).

3. ЗАЩИТА ЭЛЕКТРОННЫХ ДАННЫХ

Помимо пассивной аутентификации цифровыми подписями и контроля доступа к чипу государства или организации выдачи МОГУТ применять дополнительные средства защиты, используя более сложные способы защиты бесконтактной ИС и ее данных.

Получение доступа к электронному МСПД включает следующие этапы:

1. Получение доступа к бесконтактной ИС электронного МСПД (раздел 4).

2. Аутентификация данных (раздел 5).

3. Аутентификация чипа (раздел 6).

4. Дополнительные механизмы контроля доступа (раздел 7).

5. Считывание данных (см. часть 10 документа Doc 9303).

Для различных этапов существуют различные протоколы. Точная конфигурация электронного МСПД определяется государством или организацией выдачи. Приводимые в таблице 1 варианты могут быть надлежащим образом объединены в целях обеспечения дополнительной защиты в зависимости от потребностей выдающих органов.

Описание процедур проверки для различных конфигураций электронных МСПД приводятся в добавлении J.

Таблица 1. Защита электронных данных (резюме)

Метод	***Бесконтактная ИС***	***Система проверки***	***Преимущества***	***Примечание***
БАЗОВЫЙ МЕТОД ЗАЩИТЫ				
Пассивная аутентификация (раздел 5.1)	m	m	Доказывает, что содержание SO_D и LDS является подлинным и не изменено	Не предотвращает точное копирование или подмену ИС. Не предотвращает несанкционированный доступ. Не предотвращает скимминг
УСОВЕРШЕНСТВОВАННЫЕ МЕТОДЫ ЗАЩИТЫ				
Сравнение обычной МСЗ (OCR-B) и МСЗ (LDS), основанной на ИС	n/a	o	Доказывает, что содержание бесконтактной ИС и физического электронного МСПД соответствуют друг другу	Вносит (незначительную) сложность. Не предотвращает точное копирование бесконтактной ИС и обычного документа

Метод	*Бесконтактная ИС*	*Система проверки*	*Преимущества*	*Примечание*
Активная аутентификация (раздел 6.1)	o	o	Предотвращает копирование SO_D и доказывает, что он считан с аутентичной бесконтактной ИС. Доказывает, что бесконтактная ИС не подменена	Не предотвращает несанкционированный доступ. Вносит сложность. Для LDS2 аутентификация чипа является ОБЯЗАТЕЛЬНОЙ.
Аутентификация чипа (раздел 6.2)	o/c	o		
Базовый контроль доступа (BAC) (раздел 4.3)	Дс (см. также раздел 4.1)	m (см. также раздел 4.1)	Предотвращает скимминг и злоупотребление. Предотвращает перехват обмена сообщениями между электронным МСПД и системой проверки (при использовании для установки зашифрованного канала передачи)	Не предотвращает точное копирование или подмену ИС (требует также копирования обычного документа). Вносит сложность. Электронный МСПД поддерживает по крайней мере один BAC или PACE. Для LDS2 PACE является ОБЯЗАТЕЛЬНЫМ. PACE обеспечивает лучшую защиту от перехвата по сравнению с BAC. См. также добавление A
Установление соединения с аутентификацией паролем (PACE) (раздел 4.4)	r/c (см. также раздел 4.1)	m (см. также раздел 4.1)		
Аутентификация терминала (раздел 7.1)	o/c	o	Предотвращает несанкционированный доступ к конфиденциальным данным. Предотвращает скимминг конфиденциальных данных	Требует дополнительного управления ключами. Не предотвращает точное копирование или подмену ИС (требует также копирования обычного документа). Вносит сложность. Для LDS2 аутентификация терминала является ОБЯЗАТЕЛЬНОЙ
Шифрование данных (раздел 7.2)	o	o	Защищает дополнительные биометрические параметры. Не требует использования процессора ИС	Требует сложного управления ключами дешифровки. Не предотвращает точное копирование или подмену ИС. Вносит сложность
m – ОБЯЗАТЕЛЬНЫЙ ЭЛЕМЕНТ; r – РЕКОМЕНДУЕМЫЙ; o – ФАКУЛЬТАТИВНЫЙ; c – УСЛОВНО ОБЯЗАТЕЛЬНЫЙ; n/a – неприменимо.				

Примечание. В разделе 4 приводится подробная информация об отвечающих требованиям конфигурациях бесконтактных ИС в условиях реализации базового контроля доступа и установления соединения с аутентификацией паролем.

Применение усовершенствованных методов защиты, перечисленных в таблице 1, не затрагивает проблему обеспечения соответствия требованиям ИКАО.

4. ДОСТУП К БЕСКОНТАКТНОЙ ИС

Включение бесконтактной ИС в документ без контроля доступа к электронному МСПД создает две новые возможности для злоумышленных действий:

- хранящиеся на бесконтактной ИС данные можно считать с помощью электронного устройства без разрешения на считывание документа (скимминг);

- обмен нешифрованными данными между бесконтактной ИС и считывающим устройством может быть перехвачен с расстояния в несколько метров.

Несмотря на наличие возможных мер физической защиты от скимминга (например, экранирование обложки паспортной книжки металлической сеткой), они не решают проблемы перехвата. В этой связи предполагается, что государствам и организациям выдачи СЛЕДУЕТ внедрить механизм контроля доступа к чипу, т. е. механизм контроля доступа, фактически требующий, чтобы владелец электронного МСПД знал о том, что хранящиеся на бесконтактной ИС данные считываются безопасным способом. Такой механизм базового контроля доступа предотвращает скимминг, а также перехват.

Бесконтактная ИС, защищенная механизмом контроля доступа к чипу, отказывает в предоставлении доступа к своему содержанию, если система проверки не может доказать, что ей разрешен доступ к бесконтактной ИС. Это доказательство предоставляется по криптографическому протоколу, в соответствии с которым система проверки доказывает знание информации, извлекаемой из физического документа.

Система проверки ДОЛЖНА быть обеспечена этой информацией до считывания бесконтактной ИС. Данная информация снимается оптически/визуально с электронного МСПД (например, с МСЗ). Проверяющий ДОЛЖЕН также иметь возможность ввести эту информацию в систему проверки вручную в случае невозможности машинного считывания информации.

Предположение о том, что информацию из физического документа невозможно получить с нераскрытого документа (например, поскольку она извлекается из оптически считываемой МСЗ), позволяет допускать, что электронный МСПД сознательно предоставлен для проверки. Ввиду шифрования канала перехват передаваемых сообщений требует значительных усилий.

В настоящем разделе определяются два механизма контроля доступа к чипу:

- базовый контроль доступа (BAC, раздел 4.3), который основан исключительно на симметричной криптографии;

- установление соединения с аутентификацией паролем (PACE, раздел 4.4), при котором применяется асимметричная криптография для предоставления сеансовых ключей с более высокой степенью энтропии.

Дополнительная информация о криптостойкости сеансовых ключей приводится также в добавлении A.

4.1 Конфигурации, соответствующие требованиям

Требованиям данной спецификации отвечают следующие конфигурации:

- чипы электронных МСПД, в которых применяется только базовый контроль доступа (BAC);
- чипы электронных МСПД, в которых применяются установление соединения с аутентификацией паролем (PACE) *и* BAC;
- используются чипы электронных МСПД, в которых применяется только PACE.

Безопасность, обеспечиваемая базовым контролем доступа, ограничена структурой протокола, как указано в добавлении A. Предполагается, что увеличение мощности компьютеров с течением времени позволит проводить атаки на BAC, которые можно будет успешно осуществлять при наличии умеренных финансовых средств и в приемлемые сроки. Поэтому был согласован постепенный переход от BAC к PACE.

Был установлен следующий переходный период:

- С 1 января 2027 года в качестве чипов электронных МСПД ДОЛЖНЫ использоваться чипы, в которых применяется PACE, а чипы электронных МСПД, в которых применяется только BAC, упраздняются. Все электронные МСПД, в которых применяется только BAC и которые были выпущены до 1 января 2027 года, остаются пригодными для использования в течение всего срока их действия.

- С 1 января 2028 года BAC упраздняется, и чипы электронных МСПД ДОЛЖНЫ применять только PACE. Все электронные МСПД, в которых применяются PACE и BAC и которые были выпущены до 1 января 2028 года, остаются пригодными для использования в течение всего срока их действия.

Системы проверки, соответствующие требованиям, ДОЛЖНЫ поддерживать все удовлетворяющие требованиям конфигурации электронных МСПД. Если какой-либо электронный МСПД поддерживает как PACE, так и BAC, то система проверки ИСПОЛЬЗУЕТ либо BAC, либо PACE, но не оба механизма в рамках одного и того же сеанса.

Примечание 1. В предыдущих изданиях документа Doc 9303 допускалось использование чипов электронных МСПД, в которых не применяется контроль доступа к чипу ("открытые электронные МСПД"). Из восьмого издания это положение исключено. Тем не менее, системы проверки, отвечающие требованиям, ДОЛЖНЫ поддерживать электронные МСПД без контроля доступа к чипу.

Примечание 2. Для доступа к приложениям LDS2 ИС ДОЛЖНА требовать проведения PACE.

4.2 Процедура доступа к чипу

Процедура доступа к чипу, предназначенная для аутентификации системы проверки, включает следующие этапы.

1. **Считывание файла EF.CardAccess** **(ОБЯЗАТЕЛЬНЫЙ)**

Если PACE поддерживается электронным МСПД, то чип электронного МСПД ДОЛЖЕН предоставить параметры, которые будут использоваться для PACE в файле EF.CardAccess.

Если имеется файл EF.CardAccess, то система проверки СЧИТЫВАЕТ этот файл (см. раздел 9.2.11) для определения параметров (т. е. симметричные шифры, алгоритмы согласования ключей, параметры доменов и отображения), поддерживаемых чипом электронного МСПД. Система проверки может выбрать любой из этих параметров.

Если файл EF.CardAccess отсутствует или не содержит параметров для PACE, то система проверки ДОЛЖНА попытаться считать электронный МСПД с базовым контролем доступа (перейти к этапу 4).

2. **Считывание файла EF.DIR** **(ФАКУЛЬТАТИВНЫЙ)**

Система проверки МОЖЕТ считывать файл EF.DIR (если имеется) для извлечения перечня приложений, имеющихся на чипе электронного МСПД.

3. **PACE** **(УСЛОВНО ОБЯЗАТЕЛЬНЫЙ)**

Данный этап является РЕКОМЕНДУЕМЫМ, если PACE поддерживается чипом электронного МСПД. Если предполагается обеспечение доступа к приложениям LDS2, то этот этап является ОБЯЗАТЕЛЬНЫМ.

- Системе проверки СЛЕДУЕТ извлечь ключ K_{π} из МСЗ. Вместо МСЗ она МОЖЕТ использовать CAN, если CAN известен системе проверки.

- Чип электронного МСПД ПРИНИМАЕТ МСЗ в качестве паролей для PACE. Вместо МСЗ он МОЖЕТ дополнительно принять CAN.

- Система проверки и чип электронного МСПД взаимно аутентифицируются, используя K_{π}, и вырабатывают сеансовые ключи KS_{Enc} и KS_{MAC}. ИСПОЛЬЗУЕТСЯ протокол PACE, описанный в разделе 4.4.

В случае успешного завершения операции чип электронного МСПД выполняет следующее:

- Он НАЧИНАЕТ безопасный обмен сообщениями.

- Он ПРЕДОСТАВЛЯЕТ доступ к менее конфиденциальным данным (например, файлы EF.DG1, EF.DG2, EF.DG14, EF.DG15 и т. д. приложения электронного МСПД и объект защиты документа. Определение термина "конфиденциальные данные" приводится в части 1 документа Doc 9303).

- Он ОГРАНИЧИВАЕТ права доступа в целях обеспечения безопасного обмена сообщениями.

Система проверки ДОЛЖНА верифицировать аутентичность содержания файла EF.CardAccess с использованием файлов EF.DG14 или EF.CardSecurity и файла EF.DIR (если имеется и считан) с использованием файла EF.CardSecurity.

Примечание. Если на чипе электронного МСПД отсутствует приложение LDS2, то в файле EF.CardSecurity может отсутствовать защищенная копия файла EF.DIR.

4. **Базовый контроль доступа** **(УСЛОВНО ОБЯЗАТЕЛЬНЫЙ)**

Этот этап является ОБЯЗАТЕЛЬНЫМ, если контроль доступа к чипу предусмотрен в чипе электронного МСПД в качестве обязательного условия, а механизм PACE не был использован. Если PACE был успешно выполнен или если в электронном МСПД контроль доступа к чипу не является обязательным, этот этап пропускается.

Приложение электронного МСПД ДОЛЖНО выбираться до выполнения базового контроля доступа.

- Системе проверки СЛЕДУЕТ извлечь из МСЗ базовые ключи доступа к документу (K_{Enc} и K_{MAC}).

- Система проверки и чип электронного МСПД взаимно аутентифицируются, используя базовые ключи доступа к документу, и вырабатывают сеансовые ключи KS_{Enc} и KS_{MAC}.

В случае успешного завершения операции чип электронного МСПД выполняет следующее:

- Он НАЧИНАЕТ безопасный обмен сообщениями.

- Он ПРЕДОСТАВЛЯЕТ доступ к менее конфиденциальным данным (например, файлы EF.DG1, EF.DG2, EF.DG14, EF.DG15 и т. д. приложения электронного МСПД и объект защиты документа).

- Он ОГРАНИЧИВАЕТ права доступа в целях обеспечения безопасного обмена сообщениями.

Примечание. В результате реализации процедуры обеспечения доступа к чипу текущая DF может быть мастер-файлом (если использовалась PACE) или приложением электронного МСПД (если использовался BAC).

4.3 Базовый контроль доступа

4.3.1 Спецификация протокола

Аутентификация и установление ключей обеспечиваются трехпроходным запросно-ответным протоколом в соответствии с механизмом установления ключей 6 стандарта [ИСО/МЭК 11770-2] с использованием 3DES [FIPS 46-3] как блочного шифра. Вычисляется криптографическая контрольная сумма согласно MAC-алгоритму 3 [ИСО/МЭК 9797-1] и добавляется к шифртекстам. ДОЛЖНЫ использоваться режимы работы, описываемые в разделе 4.3.3. Размер обмениваемых одноразовых идентификаторов (nonce) ДОЛЖЕН составлять 8 байтов, а обмениваемого ключевого материала – 16 байтов. IFD (т. е. система проверки) и бесконтактная ИС НЕ ДОЛЖНЫ использовать отличительные идентификаторы в качестве одноразовых идентификаторов.

IFD и ИС конкретно выполняют следующие этапы:

1) IFD запрашивает RND.IC, посылая команду GET CHALLENGE. ИС генерирует и отвечает одноразовым идентификатором RND.IC.

2) IFD выполняет следующие операции:

   a) генерирует одноразовый идентификатор RND.IFD и ключевой материал K.IFD;

   b) генерирует конкатенацию S = RND.IFD || RND.IC || K.IFD;

   c) вычисляет криптограмму $E_{IFD} = \mathbf{E}(K_{Enc}, S)$;

   d) вычисляет контрольное число $M_{IFD} = \mathbf{MAC}(K_{MAC}, E_{IFD})$;

   e) посылает команду EXTERNAL AUTHENTICATE с функцией взаимной аутентификации, используя данные $E_{IFD} || M_{IFD}$.

3) ИС выполняет следующие операции:

   a) проверяет контрольную сумму M_{IFD} криптограммы $E_{IFD;}$

   b) расшифровывает криптограмму $E_{IFD;}$

c) извлекает RND.IC из S и проверяет, выдало ли IFD правильное значение;

d) генерирует ключевой материал K.IC;

e) генерирует конкатенацию R = RND.IC || RND.IFD || K.IC;

f) вычисляет криптограмму E_{IC} = **E**(K_{Enc}, R);

g) вычисляет контрольное число M_{IC} = **MAC**(K_{MAC}, E_{IC});

h) посылает ответ с использованием данных E_{IC} || M_{IC}.

4) IFD выполняет следующие операции:

a) проверяет контрольную сумму M_{IC} криптограммы $E_{IC;}$

b) расшифровывает криптограмму $E_{IC;}$

c) извлекает RND.IFD из R и проверяет, выдала ли ИС правильное значение.

5) Устройства IFD и ИС устанавливают сеансовые ключи KS_{Enc} и KS_{MAC}, используя механизм выработки ключей, описанный в разделах 9.7.1 и 9.7.4, с числами (K.IC xor K.IFD) в качестве совместно используемых секретных ключей.

4.3.2 Процесс проверки

Когда электронный МСПД с механизмом базового контроля доступа предоставляется проверочной системе, оптически или визуально считываемая информация используется для выработки базовых ключей доступа к документу (K_{Enc} и K_{MAC}) с целью получения доступа к бесконтактной ИС и установления защищенного канала для обмена данными между бесконтактной ИС электронного МСПД и системой проверки.

Бесконтактная ИС электронных МСПД, поддерживающая базовый контроль доступа, после установления защищенного канала ДОЛЖНА давать на неаутентифицированные попытки считывания, т. е. попытки считывания, предпринимаемые без системы безопасного обмена сообщениями (включая выбор (защищенных) файлов в LDS), ответ "Статус защиты неудовлетворителен" (0x6982). Если ИС получает открытую команду SELECT (ВЫБОР), т. е. без применения режима безопасного обмена сообщениями, по защищенному каналу, то ИС ПРЕРЫВАЕТ защищенный канал. Если открытая команда SELECT посылается до установления защищенного канала или после его прерывания, то ИС МОЖЕТ дать оба ответа – 0x6982 и 0x9000, т. е. они отвечают требованиям ИКАО.

Для аутентификации системы проверки ДОЛЖНЫ быть выполнены следующие этапы:

1) Система проверки считывает "информацию MCЗ". Эта информация состоит из конкатенации номера документа, даты рождения и даты истечения срока действия, включая соответствующие контрольные цифры, как описывается в частях 4, 5 или 6 документа Doc 9303 применительно соответственно к документам размера ПД3, ПД1 и ПД2, в машиносчитываемой зоне, используя считыватель знаков в формате OCR-B. В качестве альтернативы нужная информация может впечатываться; в этом случае она ВПЕЧАТЫВАЕТСЯ в том виде, в каком фигурирует в MCЗ. 16 наиболее значимых байтов алгоритма хэширования (SHA-1) этой "информации MCЗ" используются в качестве начального заполнения генератора ключей с целью установить базовые ключи доступа к документу, используя механизм выработки ключей, описываемый в разделе 9.7.2.

2) Система проверки и бесконтактная ИС электронного МСПД взаимно аутентифицируются и устанавливают сеансовые ключи. ДОЛЖЕН использоваться протокол аутентификации и выработки ключей, описанный выше.

3) После успешного выполнения протокола аутентификации IFD и ИС вычисляют сеансовые ключи KS_{Enc} и KS_{MAC}, используя механизм выработки ключей, описанный в разделах 9.7.1 и 9.7.4, с (K.IC xor K.IFD) совместно используемых секретных ключей. Все последующие передачи ДОЛЖНЫ защищаться методом безопасного обмена сообщениями, который описывается в разделе 9.8.

4.3.3 Криптографические спецификации

4.3.3.1 Шифрование команд Challenge (Запрос) и Response (Ответ)

Для вычисления E_{IFD} и E_{IC} ИСПОЛЬЗУЕТСЯ двухключевой 3DES в режиме CBC с нулевым вектором инициализации IV (т. е. 0x00 00 00 00 00 00 00 00) в соответствии со стандартом [ИСО/МЭК 11568-2]. При выполнении команды EXTERNAL AUTHENTICATE заполнение для вводимых данных не используется.

4.3.3.2 Аутентификация команд Challenge и Response

Криптографические контрольные суммы M_{IFD} и M_{IC} ВЫЧИСЛЯЮТСЯ с использованием MAC алгоритма 3 стандарта [ИСО/МЭК 9797-1] с блочным шифром DES (нулевой IV (8 байтов)) и метода заполнения 2 стандарта [ИСО/МЭК 9797-1]. Длина MAC ДОЛЖНА быть 8 байтов.

4.3.4 Протокольный блок данных приложения

Базовый контроль доступа осуществляется с использованием команд GET CHALLENGE и EXTERNAL AUTHENTICATE с функцией взаимной аутентификации. Эти команды ШИФРУЮТСЯ, как указано в стандарте [ИСО/МЭК 7816-4].

4.3.4.1 Команда GET CHALLENGE

Команда		
CLA		В зависимости от контекста
INS	0x84	GET CHALLENGE
P1/P2	0x0000	—
Данные		*Отсутствуют*
Ответ		
Данные	Случайный одноразовый идентификатор	
Байты состояния	0x9000	*Нормальная обработка* Случайный одноразовый идентификатор успешно сгенерирован и передан
	Прочее	*Ошибка, зависящая от операционной системы* Случайный одноразовый идентификатор передать невозможно

4.3.4.2 Команда EXTERNAL AUTHENTICATE

Команда			
CLA		В зависимости от контекста	
INS	0x82	EXTERNAL AUTHENTICATE	
P1/P2	0x0000	—	
Данные		Данные команды E_{IFD} || M_{IFD}	ОБЯЗАТЕЛЬНЫЕ
Ответ			
Данные		Данные ответа E_{IC} || M_{IC}	ОБЯЗАТЕЛЬНЫЕ
Байты состояния	0x9000	*Обычная обработка* Протокол выполнен успешно	
	Прочее	*Ошибка, зависящая от операционной системы* Сбой протокола	

4.4 Установление соединения с аутентификацией паролем

PACE представляет собой протокол согласования ключей Диффи-Хеллмана с аутентификацией паролем, который обеспечивает безопасный обмен сообщениями и основанную на пароле аутентификацию чипа электронного МСПД и системы проверки (т. е. чип электронного МСПД и система проверка совместно используют один и тот же пароль π).

PACE обеспечивает безопасный обмен сообщениями между чипом электронного МСПД и системой проверки на основе слабых (коротких) паролей. Контекст защиты установлен в мастер-файле. Указанный протокол позволяет чипу электронного МСПД убедиться в том, что система проверки имеет разрешение на доступ к хранящимся данным и обладает следующими характеристиками:

- предоставляются криптостойкие сеансовые ключи независимо от стойкости пароля;
- энтропия пароля(ей), используемых для аутентификации системы проверки, может быть очень низкой (например, 6 цифр обычно являются достаточными).

Механизм PACE использует ключи K_π, установленные из паролей с помощью функции выработки ключей $\mathbf{KDF_\pi}$ (см. раздел 9.7.3). Для глобально интероперабельных машиносчитываемых проездных документов имеются следующие два пароля и соответствующие ключи:

- MRZ: ключ K_π , определяемый как $K_\pi = \mathbf{KDF_\pi}$(MRZ), является ОБЯЗАТЕЛЬНЫМ. Его извлекают из машиносчитываемой зоны (MRZ) аналогично базовому контролю доступа, т. е. этот ключ определяют, используя номер документа, дату рождения и дату истечения срока действия.
- CAN: ключ K_π, определяемый как $K_\pi = \mathbf{KDF_\pi}$(CAN), является ФАКУЛЬТАТИВНЫМ. Его извлекают из номера доступа к карточке (CAN). CAN представляет собой номер, напечатанный на документе, и ДОЛЖЕН выбираться случайным или псевдослучайным образом (например,

используя криптографически стойкую псевдослучайную функцию). Конкретная информация, касающаяся определения поля CAN, содержится в частях 4, 5 и 6 документа Doc 9303.

Примечание. В отличие от MCЗ (номер документа, дата рождения, дата истечения срока действия) преимуществом CAN является то, что его можно легко напечатать вручную.

PACE поддерживает различные отображения в рамках протокола выполнения:

- *отображение общего типа* на основе стандарта согласования ключей Диффи-Хеллмана;
- *интегрированное отображение* на основе прямого отображения элемента поля в криптографической группе;
- *отображение для аутентификации* чипа расширяет возможности отображения общего типа и интегрирует аутентификацию чипа в протокол PACE.

Если чип поддерживает отображение для аутентификации чипа, то по крайней мере один из указанных двух видов отображения (общего типа или интегрированное) и аутентификация чипа ДОЛЖНЫ также поддерживаться чипом. Это означает, что для систем проверки, поддерживающих PACE, ОБЯЗАТЕЛЬНОЙ является только поддержка отображения общего типа и интегрированного отображения. Поддержка отображения для аутентификации чипа является ФАКУЛЬТАТИВНОЙ.

4.4.1 Спецификация протокола

Система проверки считывает параметры для механизма PACE, поддерживаемого чипом электронного МСПД, из файла EF.CardAccess (см. раздел 9.2.11) и выбирает подлежащие использованию параметры, после чего выполняется протокол.

ИСПОЛЬЗУЮТСЯ следующие команды:

- Команда READ BINARY, как указано в части 10 документа Doc 9303.
- Команда MSE:Set AT (команда MANAGE SECURITY ENVIRONMENT (УПРАВЛЕНИЕ СРЕДСТВАМИ ЗАЩИТЫ) с функцией установления шаблона аутентификации), как это указано в разделе 4.4.4.1.
- Система проверки и чип электронного МСПД ВЫПОЛНЯЮТ следующие этапы, используя последовательность команд GENERAL AUTHENTICATE, как это указано в разделе 4.4.4.2:

  1) Чип электронного МСПД случайным образом и единообразно выбирает одноразовый идентификатор s, зашифровывает его как $z = \mathbf{E}(K_\pi, s)$, где $K_\pi = \mathbf{KDF}_\pi(\pi)$ выводится из совместно используемого пароля π, и передает этот шифротекст z системе проверки.

  2) Система проверки восстанавливает открытый текст $s = \mathbf{D}(K_\pi, z)$ с помощью совместно используемого пароля π.

  3) Электронный МСПД и система проверки выполняют следующие этапы:

     a) Они обмениваются дополнительными данными, необходимыми для отображения одноразового идентификатора:

        i) в случае отображения общего типа чип электронного МСПД и система проверки обмениваются эфемерными открытыми ключами;

ii) в случае интегрированного отображения система проверки посылает дополнительный одноразовый идентификатор на чип электронного МСПД.

b) Они вычисляют параметры эфемерного домена D = **Map**(D_{IC},s,…), как это описано в разделе 4.4.3.3.

c) Они выполняют анонимное согласование ключей Диффи-Хеллмана (см. раздел 9.6) на основе эфемерных параметров домена и генерируют совместно используемый секретный ключ K = **KA**($SK_{DH,IC}$, $PK_{DH,IFD}$,D) = **KA**($SK_{DH,IFD}$,$PK_{DH,IC}$,D).

d) Во время согласования ключей по стандарту Диффи-Хеллмана ИС и система проверки ДОЛЖНЫ убедиться в том, что два открытых ключа $PK_{DH,IC}$ и $PK_{DH,IFD}$ отличаются.

e) Они вырабатывают сеансовые ключи KS_{MAC} = **KDF**$_{\text{MAC}}$(K) и KS_{Enc} = **KDF**$_{\text{Enc}}$(K), как это описано в разделе 9.7.1.

f) Они обмениваются аутентификационным маркерным изображением T_{IFD} = **MAC** (KS_{MAC},$PK_{DH,IC}$) и T_{IC} = **MAC**(KS_{MAC},$PK_{DH,IFD}$), как это описано в разделе 4.4.3.4.

4) В зависимости от условий чип электронного МСПД вычисляет данные аутентификации чипа CA_{IC}, зашифровывает их A_{IC} = **E***(KS_{Enc}, CA_{IC})* и посылает их на терминал (см. раздел 4.4.3.5.1). Терминал дешифрует A_{IC} и верифицирует аутентичность чипа, используя восстановленные данные аутентификации чипа CA_{IC} (см. раздел 4.4.3.5.2).

Упрощенная версия протокола приводится также на рис. 1.

ИС (чип)		***IFG (система проверки)***
Статические параметры домена D_{IC} Выбор случайного одноразового идентификатора s Вычисление z = **E**(K_{π},s)		
	— z →	
		Вычисление s = **D**(K_{π},z)
	← дополнительные данные для **Map** →	
D = **Map**(D_{IC},s,…) Выбор случайной пары эфемерных ключей ($SK_{DH,IC}$, $PK_{DH,IC}$,D)		D = **Map**(D_{IC},s,…) Выбор случайной пары эфемерных ключей ($SK_{DH,IFD}$,$PK_{DH,IFD}$,D)
	← $PK_{DH,IC}$, $PK_{DH,IFD}$ →	
Проверка $PK_{DH,IC} \neq PK_{DH,IFD}$ K = **KA**($SK_{DH,IC}$, $PK_{DH,IFD}$,D) Вычисление T_{IC} = **MAC**(KS_{MAC},$PK_{DH,IFD}$) [вычисление CA_{IC} и шифрование в виде A_{IC} = **E***(KS_{Enc}, CA_{IC})*.]		Проверка $PK_{DH,IC} \neq PK_{DH,IFD}$ K = **KA**($SK_{DH,IFD}$,$PK_{DH,IC}$,D) Вычисление T_{IFD} = **MAC**(KS_{MAC},$PK_{DH,IC}$)
	← T_{IC}, T_{IFD} → [— A_{IC} →]	
Верифицирование T_{IFD}		Верифицирование T_{IC} [дешифровка A_{IC} и аутентификация чипа]

Рис. 1. Установление соединения с аутентификацией паролем

4.4.2 Статус защиты

Чип электронного МСПД, который поддерживает PACE, ОТВЕЧАЕТ на неаутентифицированные попытки считывания (включая выбор (защищенных) файлов в LDS) сообщением "Статус защиты неудовлетворителен" (0x6982).

Примечание. Данная спецификация носит более ограничительный характер, чем соответствующая спецификация для электронных МСПД только с BAC.

Если PACE было успешно выполнено, то чип электронного МСПД верифицировал используемый пароль. Начинается безопасный обмен сообщениями с использованием выработанных сеансовых ключей KS_{MAC} и KS_{Enc}.

4.4.3 Криптографические спецификации

В настоящем разделе содержатся подробные криптографические данные спецификации.

Конкретные алгоритмы выбираются государством или организацией, выдающими МСПД. Система проверки ДОЛЖНА поддерживать все комбинации, описанные в нижеследующих подразделах, за исключением отображения для аутентификации чипа, которое является ФАКУЛЬТАТИВНЫМ. Чип электронного МСПД МОЖЕТ поддерживать несколько комбинаций алгоритмов.

Примечание. Некоторые алгоритмы не могут применяться в случае использования отображений для аутентификации чипа. По соображениям безопасности использование 3DES более не рекомендуется. Варианты алгоритмов DH, призванные сократить количество вариантов, которые будут применяться на терминалах, отсутствуют.

4.4.3.1 DH

Для PACE с DH ДОЛЖНЫ использоваться соответствующие алгоритмы и форматы, приведенные в разделе 9.6 и таблице 2.

Таблица 2. Алгоритмы и форматы для DH

OID	*Отображение*	*Симметр. шифр*	*Длина ключа*	*Безопасный обмен сообщениями*	*Аутент. маркерное изображение*
id-PACE-DH-GM-3DES-CBC-CBC	Общего типа	3DES	112	CBC/CBC	CBC
id-PACE-DH-GM-AES-CBC-CMAC-128	Общего типа	AES	128	CBC/CMAC	CMAC
id-PACE-DH-GM-AES-CBC-CMAC-192	Общего типа	AES	192	CBC/CMAC	CMAC
id-PACE-DH-GM-AES-CBC-CMAC-256	Общего типа	AES	256	CBC/CMAC	CMAC
id-PACE-DH-IM-3DES-CBC-CBC	Интегрированное	3DES	112	CBC/CBC	CBC
id-PACE-DH-IM-AES-CBC-CMAC-128	Интегрированное	AES	128	CBC/CMAC	CMAC
id-PACE-DH-IM-AES-CBC-CMAC-192	Интегрированное	AES	192	CBC/CMAC	CMAC
id-PACE-DH-IM-AES-CBC-CMAC-256	Интегрированное	AES	256	CBC/CMAC	CMAC

4.4.3.2 ECDH

Для PACE с ECDH ДОЛЖНЫ использоваться соответствующие алгоритмы и форматы, приведенные в разделе 9.6 и таблице 3.

ИСПОЛЬЗУЮТСЯ только простые кривые с точками в несжатом формате. СЛЕДУЕТ использовать стандартизированные параметры домена, указанные в разделе 9.5.1.

Таблица 3. Алгоритмы и форматы для ECDH

OID	***Отображение***	***Симметр. шифр***	***Длина ключа***	***Безопасный обмен сообщениями***	***Аутент. маркерное изображение***
id-PACE-ECDH-GM-3DES-CBC-CBC	Общего типа	3DES	112	CBC/CBC	CBC
id-PACE-ECDH-GM-AES-CBC-CMAC-128	Общего типа	AES	128	CBC/CMAC	CMAC
id-PACE-ECDH-GM-AES-CBC-CMAC-192	Общего типа	AES	192	CBC/CMAC	CMAC
id-PACE-ECDH-GM-AES-CBC-CMAC-256	Общего типа	AES	256	CBC/CMAC	CMAC
id-PACE-ECDH-IM-3DES-CBC-CBC	Интегрированное	3DES	112	CBC/CBC	CBC
id-PACE-ECDH-IM-AES-CBC-CMAC-128	Интегрированное	AES	128	CBC/CMAC	CMAC
id-PACE-ECDH-IM-AES-CBC-CMAC-192	Интегрированное	AES	192	CBC/CMAC	CMAC
id-PACE-ECDH-IM-AES-CBC-CMAC-256	Интегрированное	AES	256	CBC/CMAC	CMAC
id-PACE-ECDH-CAM-AES-CBC-CMAC-128	Аутентификация чипа	AES	128	CBC/CMAC	CMAC
id-PACE-ECDH-CAM-AES-CBC-CMAC-192		AES	192	CBC/CMAC	CMAC
id-PACE-ECDH-CAM-AES-CBC-CMAC-256		AES	256	CBC/CMAC	CMAC

4.4.3.3 Шифрование и одноразовые идентификаторы отображений

Чип электронного МСПД случайным и единообразным образом ВЫБИРАЕТ одноразовый идентификатор *s* в качестве строки двоичных битов длиной *l*, где *l* является величиной, кратной размеру блока в битах соответствующего блочного шифра **E**(), выбранного чипом электронного МСПД.

- Одноразовый идентификатор s ШИФРУЕТСЯ в режиме CBC в соответствии со стандартом [ИСО/МЭК 10116] с использованием ключа $K_\pi = KDF_\pi(\pi)$, полученного из пароля π и вектора IV в виде строки со всеми нулями.

- Одноразовый идентификатор s ПРЕОБРАЗУЕТСЯ в произвольный генератор с использованием присущей для данного алгоритма функции отображения **Map**.

- В случае интегрированного отображения случайным и единообразным образом ВЫБИРАЕТСЯ дополнительный одноразовый идентификатор *t* в качестве строки двоичных битов длиной *k* и посылается открытым способом. В этом случае *k* является размером ключа в битах соответствующего блочного шифра **E**(), а *l* ЯВЛЯЕТСЯ наименьшей величиной, кратной размеру блока **E**(), причем *l* ≥ *k*.

Для внесения отображения одноразового идентификатора *s* или одноразовых идентификаторов *s*, *t* в криптографическую группу ИСПОЛЬЗУЕТСЯ один из следующих типов отображения:

- *отображение общего типа* (раздел 4.4.3.3.1);
- *интегрированное отображение* (раздел 4.4.3.3.2);
- *отображение для аутентификации чипа* (раздел 4.4.3.3.3).

4.4.3.3.1 Отображение общего типа

ECDH

Функция **Map**:G → Ĝ определяется как Ĝ = s×G+H, где H выбирается в <G>, причем $\log_G H$ неизвестен. Точка H ВЫЧИСЛЯЕТСЯ с помощью анонимного согласования ключей Диффи-Хеллмана [TR-03111] в виде $H = \mathbf{KA}(SK_{Map,IC}, PK_{Map,IFD}, D_{IC}) = \mathbf{KA}(SK_{Map,IFD}, PK_{Map,IC}, D_{IC})$.

Примечание. Алгоритм согласования ключей ECKA предотвращает атаки на небольшие подгруппы за счет использования умножения на соответствующий сомножитель.

DH

Функция **Map**:g → ĝ определяется как $\hat{g} = g^s \times h$, где h выбирается в <g>, причем $\log_g h$ неизвестен. Элемент группы h ВЫЧИСЛЯЕТСЯ с помощью анонимного согласования ключей Диффи-Хеллмана в виде $h = \mathbf{KA}(SK_{Map,IC}, PK_{Map,IFD}, D_{IC}) = \mathbf{KA}(SK_{Map,IFD}, PK_{Map,IC}, D_{IC})$.

Примечание. Для предотвращения атак на небольшие подгруппы ДОЛЖЕН использоваться метод валидации открытых ключей, описанный в [RFC 2631].

4.4.3.3.2 Интегрированное отображение

ECDH

Функция **Map**:G → Ĝ определяется как $\hat{G} = f_G(\mathbf{R}_p(s,t))$, где $\mathbf{R}_p()$ является псевдослучайной функцией, которая формирует отображение октетных строк в элементах GF(p), а $f_G()$ является функцией, которая формирует отображение элементов GF(p) в <G>. Система проверки t произвольно ВЫБИРАЕТ случайный одноразовый идентификатор и посылает его на чип электронного МСПД. Описание псевдослучайной функции $\mathbf{R}_p()$ приводится ниже. Определение функции $f_G()$ изложено в [BCIMRT2010]. Информативное описание приводится в добавлении B.

DH

Функция **Map**:g → ĝ определяется как $\hat{g} = f_g(\mathbf{R}_p(s,t))$, где $\mathbf{R}_p()$ является псевдослучайной функцией, которая формирует отображение октетных строк в элементах GF(p), а $f_g()$ является функцией, которая формирует отображение элементов GF(p) в <g>. Система проверки t произвольно ВЫБИРАЕТ случайный одноразовый

идентификатор и посылает его на чип электронного МСПД. Описание псевдослучайной функции $\mathbf{R}_p()$ приводится ниже. Функция $f_g()$ определяется как $f_g(x)=x^a \bmod p$ и $a = (p-1)/q$ является сомножителем. При реализациях НЕОБХОДИМО убедиться в том, что $\hat{g} \neq 1$.

Отображение псевдослучайного номера

Функция $\mathbf{R}_p(s,t)$ представляет собой функцию, которая формирует отображение октетных строк s (длиной l бит) и t (длиной k бит) в элементах $int(x_1||x_2||...||x_n) \bmod p$ в GF(p). Функция $\mathbf{R}_p(s,t)$ приводится на рис. 2.

Построение основано на соответствующем блочном шифре **E**(), применяемом в режиме CBC согласно стандарту [ИСО/МЭК 10116] с IV=0, где k означает размер ключа (в битах) в **E**(). По мере необходимости выходной k_i ДОЛЖЕН быть сокращен до размера ключа *k*. В качестве значения *n* ВЫБИРАЕТСЯ наименьшее число, причем $n*l \geq \log_2 p + 64$.

Примечание. Сокращение необходимо только для AES-192. Используются октеты 1–24 ключа k_i; дополнительные октеты не используются. В случае DES размер k считается равным 128 бит, а результатом R(s,t) будет 128 бит.

Константы c_0 и c_1 определяются следующим образом:

- Для 3DES и AES-128 (l=128):
  - `c0=0xa668892a7c41e3ca739f40b057d85904`
  - `c1=0xa4e136ac725f738b01c1f60217c188ad`
- Для AES-192 и AES-256 (l=256):
  - `c0=`
    `0xd463d65234124ef7897054986dca0a174e28df758cbaa03f240616414d5a1676`
  - `c1=`
    `0x54bd7255f0aaf831bec3423fcf39d69b6cbf066677d0faae5aadd99df8e53517`

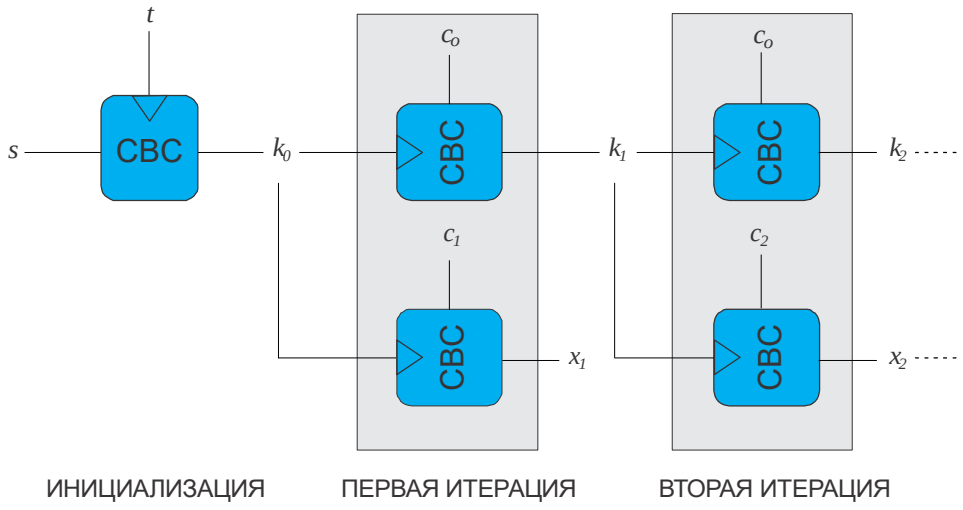

Рис. 2. Отображение псевдослучайного номера

4.4.3.3.3 Отображение для аутентификации чипа

Этап отображения у механизма PACE-CAM идентичен этапу отображения у механизма PACE-GM (см. раздел 4.4.3.3.1).

4.4.3.4 Аутентификационное маркерное изображение

Аутентификационное маркерное изображение ВЫЧИСЛЯЕТСЯ по объекту данных открытого ключа (см. раздел 9.4), содержащему идентификатор объекта, как указано в команде MSE:Set AT (см. раздел 4.4.4.1), и по полученному эфемерному открытому ключу (т. е. исключая параметры домена, см. раздел 9.4.5), используя код аутентификации и ключ KS_{MAC}, выработанный с помощью механизма согласования ключей.

Примечание. Заполнение осуществляется в рамках внутреннего механизма кодом аутентификации сообщения, т. е. никакого заполнения, специфического для того или иного приложения, не осуществляется.

3DES

3DES [FIPS 46-3] ИСПОЛЬЗУЕТСЯ в режиме Retail в соответствии с методом заполнения 2 алгоритма 3 MAC стандарта [ИСО/МЭК 9797-1] с блочным шифром DES и вектором IV=0.

AES

AES [FIPS 197] ИСПОЛЬЗУЕТСЯ в режиме CMAC [SP 800-38B] с длиной MAC, равной 8 бит.

4.4.3.5 Зашифрованные данные аутентификации чипа

Чип электронного МСПД ДОЛЖЕН предоставить пару(ы) статических ключей SK_{IC}, PK_{IC}, как описано в разделе 6.2. Зашифрованные данные аутентификации чипа являются ОБЯЗАТЕЛЬНЫМИ для PACE, если используется отображение для аутентификации чипа.

4.4.3.5.1 Генерирование чипом электронного МСПД

Данные аутентификации чипа ВЫЧИСЛЯЮТСЯ в виде $CA_{IC} = (SK_{IC})^{-1} * SK_{Map,IC} \bmod p$, где SK_{IC} – статически закрытый ключ чипа, $SK_{Map,\,IC}$ – эфемерный закрытый ключ, используемый чипом для вычисления H на этапе отображения PACE (см. раздел 4.4.3.3.1) и p – порядок используемой криптографической группы. Данные аутентификации чипа ШИФРУЮТСЯ с использованием ключа KS_{Enc}, выработанного с помощью механизма согласования ключей, в виде $A_{IC} = \mathbf{E}(KS_{Enc}, CA_{IC})$ для получения зашифрованных данных аутентификации чипа.

Примечание. $(SK_{IC})^{-1}$ *может быть заранее вычислен в ходе персонализации чипа* электронного *МСПД и храниться безопасным образом на чипе, избегая модульной инверсии в процессе работы чипа.*

4.4.3.5.2 Верификация терминалом

Терминал ДЕШИФРУЕТ A_{IC} для восстановления CA_{IC} и ВЕРИФИЦИРУЕТ $PK_{Map,IC} = \mathbf{KA}(CA_{IC}, PK_{IC}, D_{IC})$, где PK_{IC} представляет собой статический открытый ключ чипа электронного МСПД.

Примечание. Пассивная аутентификация ДОЛЖНА осуществляться в сочетании с отображением для аутентификации чипа. Только после успешной валидации соответствующего объекта защиты чип электронного МСПД может считаться подлинным.

4.4.3.5.3 Заполнение

Подлежащие шифрованию данные ЗАПОЛНЯЮТСЯ в соответствии с "методом заполнения 2" стандарта [ИСО/МЭК 9797-1].

4.4.3.5.4 AES

AES [19] ИСПОЛЬЗУЕТСЯ в режиме CBC в соответствии со стандартом [ИСО/МЭК 10116] с вектором $IV=E(KS_{Enc},-1)$, где *-1* представляет собой строку длиной 128 бит, в которой все биты установлены на 1.

4.4.4 Блоки протокольных данных приложения

Для реализации PACE ИСПОЛЬЗУЕТСЯ следующая последовательность команд:

1. MSE:Set AT;
2. GENERAL AUTHENTICATE.

4.4.4.1 Команда MSE:Set AT

Команда MSE:Set AT используется для выбора и инициализации протокола PACE. Использование команды MSE:Set AT для реализации PACE указывается идентификатором объекта PACE (см. разделы 4.4.3 и 9.2.3), приводимого в качестве исходных данных криптографического механизма с тегом 0x80, см. таблицу ниже.

Команда			
CLA		В зависимости от контекста	
INS	0x22	Управление средствами защиты	
P1/P2	0xC1A4	Установка шаблона аутентификации для взаимной аутентификации	
Данные	0x80	*Ссылочные данные криптографического механизма* Идентификатор объекта выбираемого протокола (только значение, тег 0x06 опускается)	ОБЯЗАТЕЛЬНЫЕ
	0x83	*Ссылочные данные открытого ключа/секретного ключа* В этом объекте данных подлежащий использованию пароль указывается следующими значениями: 0x01: МСЗ_информация 0x02: CAN	ОБЯЗАТЕЛЬНЫЕ
	0x84	*Ссылочные данные закрытого ключа/ссылочные данные для вычисления сеансового ключа* Этот объект данных является ОБЯЗАТЕЛЬНЫМ для указания идентификатора подлежащих использованию параметров домена,	УСЛОВНО ОБЯЗАТЕЛЬНЫЕ

		если параметры домена неоднозначны, т. е. имеется несколько наборов параметров домена для целей PACE	
	0x7F4C	*Шаблон определения полномочий обладателя сертификата* Этот объект данных (определенный в части 12 документа Doc 9303) ДОЛЖЕН присутствовать, если терминал требует представить ссылку(и) на сертифицирующий полномочный орган, которая(ые) используется(ются) при аутентификации терминала и подлежат возврату в рамках PACE (см. раздел 4.4.5). Идентификатор объекта, содержащийся в этом объекте данных, устанавливается на id-IS (см. часть 10 документа Doc 9303). Все биты доступа в дискреционном шаблоне данных устанавливаются терминалом на цифру 1 (единицу)	

Ответ		
Данные	–	Отсутствуют
Байты состояния	0x9000	*Нормальная обработка* Протокол выбран и инициализирован
	0x6A80	*Неправильные параметры в поле данных команды* Алгоритм не поддерживается или сбой инициализации
	0x6A88	*Ссылочные данные не найдены* Ссылочные данные (т. е. пароль или параметры домена) отсутствуют
	прочее	*Ошибка, зависящая от операционной системы* Сбой протокола инициализации

Примечание 1. Некоторые операционные системы принимают выбор отсутствующего ключа и сообщают об ошибке, только когда данный ключ используется для целей выбора.

Примечание 2. Для команды MSE:Set ИС СЛЕДУЕТ игнорировать объекты данных с тегами, которые конкретно для этой команды не указаны. Терминалу НЕ СЛЕДУЕТ включать объекты данных с тегами, не поддающимися распознаванию ИС.

4.4.4.2 Команда GENERAL AUTHENTICATE

Для выполнения протокола PACE используется определенная последовательность команд GENERAL AUTHENTACATE.

Команда			
CLA		В зависимости от контекста	
INS	0x86	GENERAL AUTHENTICATE	
P1/P2	0x0000	Ключи и протокол, известные в виде подразумеваемых	
Данные	0x7C	*Динамические аутентификационные данные* Объекты данных, присущие конкретному протоколу	ОБЯЗАТЕЛЬНЫЕ

Ответ			
Данные	0x7C	*Динамические аутентификационные данные* Объекты данных, присущие конкретному протоколу, как указано в разделе 4.4.5	ОБЯЗАТЕЛЬНЫЕ
Байты состояния	0x9000	*Нормальная обработка* Протокол (этап) был успешным	
	0x6300	*Сбой аутентификации* Сбой протокола (этапа)	
	0x6A80	*Неправильные данные в поле данных команды* Представленные данные недействительны	
	прочее	*Ошибка, зависящая от операционной системы* Сбой протокола (этапа)	

4.4.4.3 Составление последовательности команд

Для команды GENERAL AUTHENTICATE ДОЛЖНА использоваться определенная последовательность команд, чтобы увязать эту цепочку команд с выполнением протокола. Такая последовательность команд НЕ ДОЛЖНА использоваться для иных целей, если это четко не указано в чипе. Подробная информация по составлению последовательности команд содержится в стандарте [ИСО/МЭК 7816-4].

4.4.5 Обмениваемые данные

Объекты данных, присущие конкретному протоколу, ОБМЕНИВАЮТСЯ в рамках цепочки команд GENERAL AUTHENTICATE, при этом конкретные протокольные данные команд и ответов инкапсулируются в объекте данных динамической аутентификации (см. раздел 4.4.4.2) с тегами, зависящими от контекста, как указано в таблице 4.

Таблица 4. Обмениваемые данные для PACE

Этап	***Описание***	***Данные протокольных команд***		***Данные протокольных ответов***	
1.	Зашифрованный одноразовый идентификатор	-	Отсутствуют[1]	0x80	Зашифрованный одноразовый идентификатор
2.	Одноразовый идентификатор отображения	0x81	Данные отображения	0x82	Данные отображения
3.	Выполнение согласования ключей	0x83	Эфемерный открытый ключ	0x84	Эфемерный открытый ключ

1. Это означает незаполненный объект динамических аутентификационных данных.

4.	Взаимная аутентификация	0x85	Аутентификационное маркерное изображение	0x86	Аутентификационное маркерное изображение
				0x87	Ссылка на сертифицирующий полномочный орган (ФАКУЛЬТАТИВНАЯ)
				0x88	Ссылка на сертифицирующий полномочный орган (ФАКУЛЬТАТИВНАЯ)
				0x8A	Зашифрованные данные аутентификации чипа (УСЛОВНО ОБЯЗАТЕЛЬНЫЕ)

Ссылка(и) на сертифицирующий полномочный орган должна присутствовать, если в ходе выполнения PACE (см. раздел 4.4.4.1) объект данных 0x7F4C был передан ИС и ИС поддерживала аутентификацию терминала. В этом случае объект данных 0x87 СОДЕРЖИТ самую последнюю ссылку на сертифицирующий полномочный орган. Объект данных 0x88 МОЖЕТ содержать предыдущую ссылку на сертифицирующий полномочный орган.

Зашифрованные данные аутентификации чипа (см. раздел 4.4.3.5) ДОЛЖНЫ представляться, если используется отображение для аутентификации чипа, и НЕ ДОЛЖНЫ представляться в иных случаях.

4.4.5.1 Зашифрованный одноразовый идентификатор

Зашифрованный одноразовый идентификатор (см. раздел 4.4.3.3) ШИФРУЕТСЯ в виде 8-битной строки.

4.4.5.2 Данные отображения

Обмениваемые данные являются специфическими для используемого отображения:

4.4.5.2.1 Отображение общего типа

Эфемерные открытые ключи (см. раздел 4.4.3.3 и раздел 9.4.5) ШИФРУЮТСЯ в виде точки эллиптической кривой (ECDH) или неподписанного целого числа (DH).

4.4.5.2.2 Интегрированное отображение

Одноразовый идентификатор *t* ШИФРУЕТСЯ в виде восьмибитной строки.

Примечание. В случае интегрированного отображения зависящий от контекста объект данных 0x82 ЯВЛЯЕТСЯ пустым.

4.4.5.2.3 Отображение для аутентификации чипа

Шифрование данных отображения идентично отображению общего типа (см. раздел 4.4.5.2.1).

4.4.5.3 Открытые ключи

Открытые ключи ШИФРУЮТСЯ, как описано в разделе 9.4.5.

4.4.5.4 Аутентификационное маркерное изображение

Аутентификационное маркерное изображение (см. раздел 4.4.3.4) ШИФРУЕТСЯ в виде 8-битной строки.

4.4.5.5 Ссылка на сертифицирующий полномочный орган

Объекты данных ссылки на сертифицирующий полномочный орган (CAR) кодируются в соответствии с положениями части 12 документа Doc 9303.

4.4.5.6 Зашифрованные данные идентификации чипа

Данные аутентификации чипа КОДИРУЮТСЯ в виде 8-битной строки, используя для кодирования функцию FE2OS(), описанную в стандарте [TR-03111]. Следует учитывать, что FE2OS() требует кодирования с тем же количеством октетов, что и порядок группы, равный простому числу, т. е. возможно включение значений, начинающихся с 0x00. Зашифрованные данные аутентификации чипа КОДИРУЮТСЯ как 8-битные строки.

5. АУТЕНТИФИКАЦИЯ ДАННЫХ

Помимо групп данных LDS, бесконтактная ИС содержит также объект защиты документа (SO_D). Этот объект подписывается в цифровой форме государством выдачи и содержит хэшированное представление содержания LDS (см. часть 10 документа Doc 9303).

Система проверки, содержащая открытый ключ лица, подписывающего документы, каждого государства, или считавшая с электронного МСПД сертификат лица, подписывающего документы (C_{DS}), может верифицировать объект защиты документа (SO_D). Таким способом через содержание объекта защиты документа (SO_D) производится аутентификация содержания LDS.

Этот механизм верификации не требует использования процессорных возможностей бесконтактной ИС электронного МСПД. В этой связи он называется "пассивной аутентификацией" содержания бесконтактной ИС.

Пассивная аутентификация доказывает, что содержание объекта защиты документа (SO_D) и LDS является подлинным и не было изменено. Она не предотвращает точное копирование содержания бесконтактной ИС или ее подмену.

Следовательно, систему пассивной аутентификации СЛЕДУЕТ поддерживать дополнительной физической проверкой электронного МСПД.

5.1 Пассивная аутентификация

5.1.1 Процесс проверки

Система проверки выполняет следующие этапы:

1. Система проверки СЧИТЫВАЕТ с бесконтактной ИС объект защиты документа (SO_D) (который ДОЛЖЕН содержать сертификат лица, подписывающего документы (C_{DS}), см. также часть 10 документа Doc 9303).

2. Система проверки ФОРМИРУЕТ и ВАЛИДИРУЕТ путь сертификации от "якоря доверия" до сертификата лица, подписывающего документы, который используется для подписи объекта защиты документа (SO_D) в соответствии с частью 12 документа Doc 9303.

3. Система проверки ИСПОЛЬЗУЕТ верифицированный открытый ключ лица, подписывающего документы, для верификации подписи объекта защиты документа (SO_D).

4. Система проверки МОЖЕТ считать соответствующие группы данных с бесконтактной ИС.

5. Система проверки ОБЕСПЕЧИВАЕТ аутентичность и неизменность группы данных путем хэширования содержания и сравнения результата с соответствующим хэш-значением в объекте защиты документа (SO_D).

Нижеследующие дополнительные проверки считаются передовой практикой:

1. Система проверки или инспектор ДОЛЖНЫ проверить присутствие в сертификате лица, подписывающего документы, расширения для типа документа.

   - При наличии такового система проверки ДОЛЖНА убедиться в совпадении данных расширения для типа документа, данных о типе документа, взятых из группы данных 1, и данных о типе документа, взятых из визуальной МСЗ (см. соответственно части 12, 10 и 3 документа Doc 9303).

   - При отсутствии такового система проверки ДОЛЖНА удостовериться в том, что поле “применяемость ключа” в сертификате лица, подписывающего документы, установлено на цифровую подпись и что сертификат лица, подписывающего документы, не содержит расширения “расширенная применяемость ключей” (см. часть 12 документа Doc 9303).

2. Система проверки или инспектор ДОЛЖНЫ убедиться в совпадении кодов стран, содержащихся в следующих источниках:

   - поле субъекта и, если присутствует, альтернативное имя субъекта сертификата лица, подписывающего документы;

   - поле субъекта и, если присутствует, альтернативное имя субъекта "якоря доверия" (сертификат CSCA);

   - группа данных 1, считываемая с бесконтактной ИС;

   - визуальная МСЗ.

Кроме того, система проверки или инспектор МОГУТ сравнить содержание группы данных 1 с визуальной МСЗ (см. соответственно части 12, 10 и 3 документа Doc 9303).

3. Система проверки ДОЛЖНА убедиться в том, что дата выдачи электронного МСПД вписывается в период применяемости закрытого ключа, указанный в сертификате лица, подписывающего документы (см. часть 12 документа Doc 9303).

Теперь биометрическая информация может использоваться для осуществления биометрической верификации в отношении лица, которое представляет электронный МСПД.

5.1.2 Процедура дополнительной проверки для приложений LDS2

Данные, внесенные после выдачи электронного МСПД, объектом защиты данных, подписываемым лицом, выдающим документ, не защищаются. Для проверки аутентичности данных, внесенных после выдачи документа, в отношении каждого внесенного объекта данных система проверки ДОЛЖНА выполнить следующие этапы:

1. Система проверки ФОРМИРУЕТ и ВАЛИДИРУЕТ путь сертификации от "якоря доверия" до сертификата лица, подписывающего документы, который используется для подписи объекта данных в соответствии с частью 12 документа Doc 9303. Система проверки МОЖЕТ использовать как сертификаты, известные до этого, так и сертификаты, извлеченные из чипа для построения этого пути (см. часть 10 документа Doc 9303).

2. Для проверки подписи объекта данных система проверки ИСПОЛЬЗУЕТ верифицированный открытый ключ лица, подписавшего документ.

Примечание. В отношении объекта данных, подлинность которых рассматривается в качестве элемента, не имеющего значения для процесса проверки, используемого принимающим государством или организацией, выполнение этой процедуры можно пропустить.

6. АУТЕНТИФИКАЦИЯ БЕСКОНТАКТНОЙ ИС

Государство или организация выдачи МОГУТ принять решение по защите своих электронных МСПД от подмены чипа.

Имеются следующие механизмы верификации аутентичности чипа.

1. *Активная аутентификация,* как определено в разделе 6.1. Поддержка активной аутентификации указывается присутствием файла EF.DG15. При его наличии терминал МОЖЕТ считать и верифицировать файл EF.DG15 и выполнить активную аутентификацию.

2. *Аутентификация чипа,* как определено в разделе 6.2. Поддержка аутентификации чипа указывается присутствием соответствующего файла "Сведения о защите" (`SecurityInfos`) в файле EF.DG14/EF.CardSecurity. При его наличии терминал МОЖЕТ считать и верифицировать файлы EF.DG14/EF.CardSecurity и выполнить аутентификацию чипа.

3. *PACE с отображением для аутентификации чипа* (PACE-CAM), как определено в разделе 4.4. Поддержка указывается присутствием соответствующей структуры `PACEInfo` в файле EF.CardAccess. Если PACE-CAM был выполнен успешно в рамках процедуры доступа к чипу, то терминал МОЖЕТ выполнить следующие операции для аутентификации чипа:

   - считать и верифицировать файл EF.CardSecurity;

   - использовать открытый ключ из файла EF.CardSecurity вместе с данными отображений и данными для аутентификации чипа, полученными в рамках PACE-CAM для аутентификации чипа (раздел 4.4.3.5.2).

6.1 Активная аутентификация

Активная аутентификация аутентифицирует бесконтактную ИС путем подписания запроса, посылаемого устройством IFD (системой проверки) с использованием закрытого ключа, известного только ИС.

С этой целью бесконтактная ИС содержит собственную пару ключей активной аутентификации (KPr_{AA} и KPu_{AA}). Хэшированное представление группы данных 15 (информация об открытом ключе (KPu_{AA})) хранится в объекте защиты документа (SO_D) и, следовательно, аутентифицируется цифровой подписью выдающего лица. Соответствующий закрытый ключ (KPr_{AA}) хранится в защищенной памяти бесконтактной ИС.

Путем аутентификации визуальной МСЗ (через хэшированную МСЗ в объекте защиты документа (SO_D)) в сочетании с запросом-ответом система проверки, используя пару ключей активной аутентификации (KPr_{AA} и KPu_{AA}) электронного МСПД, подтверждает, что объект защиты документа считан с подлинной бесконтактной ИС, хранящейся в подлинном электронном МСПД.

Активная аутентификация требует использования процессорных возможностей бесконтактной ИС электронного МСПД.

6.1.1 Спецификация протокола

Активная аутентификация выполняется с использованием команды INTERNAL AUTHENTICATE стандарта [ИСО/МЭК 7816-4].

Если активная аутентификация выполняется после установления безопасного обмена сообщениями, все команды и ответы ДОЛЖНЫ передаваться в виде APDU безопасного обмена сообщениями в соответствии с разделом 9.8.

Конкретно, IFD (система проверки) и ИС (бесконтактная ИС электронного МСПД) выполняют следующие этапы:

1. IFD генерирует одноразовый идентификатор RND.IFD и посылает его на ИС, используя команду INTERNAL AUTHENTICATE.

2. ИС выполняет следующие операции:

   a) генерирует сообщение M;

   b) вычисляет h(M);

   c) вычисляет подпись σ и посылает ответ на IFD.

3. IFD верифицирует ответ по посланной команде INTERNAL AUTHENTICATE и проверяет, выдала ли ИС правильное значение.

6.1.2 Криптографические спецификации

6.1.2.1 Одноразовый идентификатор

Вводимый параметр представляет собой одноразовый идентификатор (RND.IFD), который ДОЛЖЕН составлять 8 бит.

Примечание. Одноразовые идентификаторы НЕ ДОЛЖНЫ повторно использоваться, например одноразовый идентификатор, использованный для BAC/PACE, НЕ ДОЛЖЕН повторно использоваться для активной аутентификации.

6.1.2.2 RSA

Когда используется механизм целочисленной факторизации, то ИС ВЫЧИСЛЯЕТ подпись в соответствии со схемой генерирования цифровой подписи 1 стандарта [ИСО/МЭК 9796-2].

В нижеследующем примере k означает длину ключа для генерирования подписи, а L_h – длину результата функции хэширования H, используемой во время генерирования подписи. Если при генерировании подписи используется SHA-1, то ДОЛЖЕН использоваться вариант 1 поля завершителя (и t устанавливается на 1), в противном случае ДОЛЖЕН использоваться вариант 2 поля концевика (и t устанавливается на 2).

Для варианта 2 ИСПОЛЬЗУЮТСЯ следующие значения поля завершителя:

Функция хэширования	SHA-224	SHA-256	SHA-384	SHA-512
Поле концевика	0x38CC	0x34CC	0x36CC	0x35CC

По соображения интероперабельности в качестве функции хэширования для активной аутентификации с использованием алгоритма Ривеста-Шамира-Адельмана (RSA) поддерживаются только SHA-1, SHA-224, SHA-256, SHA-384 и SHA-512.

Сообщение M, которое предстоит подписать, ЯВЛЯЕТСЯ конкатенацией M1 и M2, где M1 ДОЛЖЕН быть одноразовым идентификатором длиной c – 4 бит (RND.IC), генерированным электронным МСПД, где c (*емкость подписи*) соответствует $c = k – L_h – (8 \times t) – 4$, а M2 является RND.IFD, генерированным системой проверки.

Результатом вычисления подписи ДОЛЖНА быть подпись σ без невосстанавливаемой части сообщения M2.

В электронных МСПД СЛЕДУЕТ применять схему генерирования подписи, указанную в п. B.6 стандарта [ИСО/МЭК 9796-2], и НЕ СЛЕДУЕТ использовать схему генерирования подписи, изложенную в п. B.4 стандарта [ИСО/МЭК 9796-2]. В электронных МСПД НЕ ПРИМЕНЯЮТСЯ другие схемы генерирования подписи.

В системах проверки ПРИМЕНЯЮТ схему генерирования подписи, указанную в п. B.6 стандарта [ИСО/МЭК 9796-2], и в них СЛЕДУЕТ использовать систему генерирования подписи, указанную в п. B.4 стандарта [ИСО/МЭК 9796-2].

6.1.2.3 ECDSA

Для ECDSA ИСПОЛЬЗУЕТСЯ простой формат подписи согласно [TR-03111]. ИСПОЛЬЗУЮТСЯ только простые кривые с точками в несжатом формате. ИСПОЛЬЗУЕТСЯ алгоритм хэширования, выдающий результат, длина которого аналогична или короче длины используемого ключа ECDSA. В качестве функций хэширования поддерживаются только SHA-224, SHA-256, SHA-384 и SHA-512. RIPEMD-160 и SHA-1 НЕ ИСПОЛЬЗУЮТСЯ.

Сообщение M, которое предстоит подписать, представляет собой одноразовый идентификатор RND.IFD, предоставленный системой проверки.

6.1.3 Протокольные блоки данных приложения

Активная аутентификация выполняется путем единственного инициирования команды INTERNAL AUTHENTICATE, как указано в стандарте [ИСО/МЭК 7816-4].

Команда			
CLA		В зависимости от контекста	
INS	0x88	INTERNAL AUTHENTICATE	
P1/P2	0x0000	—	
Данные		*RND.IFD*	ОБЯЗАТЕЛЬНЫЕ
Ответ			
Данные		Подпись σ, генерированная ИС	ОБЯЗАТЕЛЬНЫЕ
Байты состояния	0x9000	*Нормальная обработка* Протокол успешно выполнен	
	Прочее	*Ошибка, зависящая от операционной системы* Сбой протокола	

6.1.4 Ключи активной аутентификации

Пары ключей активной аутентификации (KPr_{AA} и KPu_{AA}) ГЕНЕРИРУЮТСЯ безопасным образом.

Как открытый ключ активной аутентификации (KPu_{AA}), так и закрытый ключ активной аутентификации (KPr_{AA}) хранятся на бесконтактной ИС электронного МСПД. После этого никакой механизм управления ключами к этим ключам не применяется.

Примечание. Следует отметить, что при использовании длины ключа свыше 1848 бит (если применяется безопасный обмен сообщениями по стандарту 3DES) или 1792 бит (если применяется безопасный обмен сообщениями по стандарту AES) в процессе активной аутентификации с безопасным обменом сообщениями чип электронного МСПД и система проверки ДОЛЖНЫ поддерживать APDU увеличенной длины.

Государства или организации выдачи ВЫБИРАЮТ надлежащую длину ключей, обеспечивающую защиту от атак в течение всего срока службы электронного МСПД. СЛЕДУЕТ учитывать соответствующие криптографические каталоги.

6.1.5 Информация об открытых ключах активной аутентификации

Открытый ключ активной аутентификации хранится в группе данных 15 LDS. Формат этой структуры (`SubjectPublicKeyInfo`) указан в стандарте [RFC 5280] (см. раздел 9.1). Все объекты защиты ДОЛЖНЫ быть построены в формате, использующем особые правила кодирования (DER), чтобы сохранить целостность находящихся в них подписей.

```
ActiveAuthenticationPublicKeyInfo ::= SubjectPublicKeyInfo
```

6.1.6 Процесс проверки

Если системе проверки предоставляется электронный МСПД с группой данных 15, МОЖЕТ использоваться механизм активной аутентификации с целью гарантировать, что данные считываются с подлинной бесконтактной ИС и что бесконтактная ИС и физический документ принадлежат друг другу.

Проверочная система и бесконтактная ИС выполняют следующие этапы:

1. Вся МСЗ визуально считывается со страницы данных электронного МСПД (если она еще не считана в рамках процедуры базового контроля доступа) и сравнивается со значением МСЗ в группе данных 1. Поскольку аутентичность и целостность группы данных 1 проверены посредством пассивной аутентификации, сходство гарантирует, что визуальная МСЗ является аутентичной и не изменена.

2. Пассивная аутентификация также доказала аутентичность и целостность группы данных 15. Это гарантирует, что открытый ключ активной аутентификации (KPu_{AA}) является аутентичным и не изменен.

3. Чтобы гарантировать, что объект защиты документа (SO_D) не является копией, система проверки использует пару ключей активной аутентификации электронного МСПД (KPr_{AA} и KPu_{AA}) по запросноответному протоколу с бесконтактной ИС электронного МСПД, как описывается выше.

Успешное выполнение запросно-ответного протокола доказывает, что объект защиты документа (SO_D) принадлежит физическому документу, бесконтактная ИС является подлинной и бесконтактная ИС и физический документ принадлежат друг другу.

6.2 Аутентификация чипа

Протокол аутентификации чипа представляет собой протокол согласования эфемерно-статических ключей Диффи-Хеллмана, которые обеспечивают безопасный обмен информацией и одностороннюю аутентификацию чипа электронного МСПД.

Основные отличия от активной аутентификации состоят в следующем:

- семантика команды запроса недоступна, поскольку расшифровка, производимая этим протоколом, не подлежит передаче;

- помимо аутентификации чипа электронного МСПД данный протокол также предоставляет криптостойкие сеансовые ключи.

Подробная информация о семантике команды запроса содержится в добавлении C.

Пара(ы) статических ключей аутентификации чипа ДОЛЖНА (ДОЛЖНЫ) храниться на чипе электронного МСПД.

- Закрытый ключ ХРАНИТСЯ в защищенной памяти чипа электронного МСПД.

- Открытый ключ ПРЕДОСТАВЛЯЕТСЯ в поле `SubjectPublicKeyInfo` в структуре `ChipAuthenticationPublicKeyInfo` (см. раздел 9.2.6).

Указанный протокол обеспечивает неявную аутентификацию как самого чипа электронного МСПД, так и хранящихся данных путем обеспечения безопасного обмена сообщениями, используя новые сеансовые ключи.

Если ИС поддерживает аутентификацию чипа, то ИС МОЖЕТ также поддерживать аутентификацию чипа в мастер-файле и/или МОЖЕТ поддерживать аутентификацию чипа в приложении электронного МСПД. Если аутентификация чипа используется в сочетании с доступом к группам данных в приложениях LSD2, то ИС ДОЛЖНА поддерживать аутентификацию чипа в мастер-файле.

Примечание. Если требуется совместимость с европейским расширенным контролем доступа [TR-03110], то ИС ДОЛЖНА поддерживать аутентификацию чипа в приложении электронного МСПД.

6.2.1 Спецификации протоколов

Терминал и чип электронного МСПД выполняют следующие этапы:

1. Чип электронного МСПД посылает на терминал свой статический открытый ключ Диффи-Хеллмана PK_{IC} и параметры домена D_{IC}.

2. Терминал генерирует эфемерную пару ключей Диффи-Хеллмана $(SK_{DH,IFD}, PK_{DH,IFD}, D_{IC})$ и посылает эфемерный открытый ключ $PK_{DH,IFD}$ на чип электронного МСПД.

3. Чип электронного МСПД и терминал вычисляют следующее:

   a) совместный секретный ключ K = **KA**$(SK_{IC}, PK_{DH,IFD}, D_{IC})$ = **KA**$(SK_{DH,IFD}, PK_{IC}, D_{IC})$;

   b) сеансовые ключи KS_{MAC} = $\mathbf{KDF}_{MAC}(K)$ и KS_{Enc} = $\mathbf{KDF}_{Enc}(K)$, выработанные из ключа K для безопасного обмена сообщениями.

Упрощенная версия показана на рис. 3.

ИС (чип)		***IFD (система проверки)***
Пара статических ключей $(SK_{IC}, PK_{IC}, D_{IC})$		
	— PK_{IC}, D_{IC} →	
		Выбор произвольной пары эфемерных ключей $(SK_{DH,IFD}, PK_{DH,IFD}, D_{IC})$
	← $PK_{DH,IFD}$ —	
K = **KA**$(SK_{IC}, PK_{DH,IFD}, D_{IC})$		K = **KA**$(SK_{DH,IFD}, PK_{IC}, D_{IC})$

Рис. 3. Аутентификация чипа

Для верификации аутентичности PK_{IC} терминал ВЫПОЛНЯЕТ пассивную аутентификацию.

6.2.2 Статус защиты

Если аутентификация чипа была успешно выполнена, то безопасный обмен сообщениями возобновляется, используя выработанные сеансовые ключи KS_{MAC} и KS_{Enc}. В противном случае процесс безопасного обмена сообщениями продолжается, используя установленные ранее сеансовые ключи (PACE или базовый контроль доступа).

Примечание. Пассивная аутентификация ДОЛЖНА выполняться в сочетании с аутентификацией чипа. Только после успешной валидации соответствующего объекта защиты чип электронного *МСПД может считаться подлинным.*

6.2.3 Криптографические спецификации

Конкретные алгоритмы выбираются государством или организацией, выдающими МСПД. Система проверки ДОЛЖНА поддерживать все комбинации, описанные в следующих подразделах. Чип электронного МСПД Может поддерживать несколько комбинаций алгоритмов.

6.2.3.1 Аутентификация чипа с использованием DH

Для аутентификации чипа с использованием DH ДОЛЖНЫ применяться соответствующие алгоритмы и форматы, содержащиеся в разделе 9.6 и таблице 5. В случае открытых ключей вместо X9.42 [X9.42] ДОЛЖНЫ использоваться PKCS#3 [PKCS#3].

Таблица 5. Идентификаторы объекта для аутентификации чипа с использованием DH

OID	*Симметр. шифр*	*Длина ключа*	*Безопасный обмен сообщениями*
id-CA-DH-3DES-CBC-CBC	3DES	112	CBC/CBC
id-CA-DH-AES-CBC-CMAC-128	AES	128	CBC/CMAC
id-CA-DH-AES-CBC-CMAC-192	AES	192	CBC/CMAC
id-CA-DH-AES-CBC-CMAC-256	AES	256	CBC/CMAC

6.2.3.2 Аутентификация чипа с использованием ECDH

Для аутентификации чипа с помощью ECDH ДОЛЖНЫ использоваться соответствующие алгоритмы и форматы, содержащиеся в разделе 9.6 и таблице 6.

Таблица 6. Идентификаторы объекта для аутентификации чипа с помощью ECDH

OID	*Симметр. шифр*	*Длина ключа*	*Безопасный обмен сообщениями*
id-CA-ECDH-3DES-CBC-CBC	3DES	112	CBC / CBC
id-CA-ECDH-AES-CBC-CMAC-128	AES	128	CBC / CMAC
id-CA-ECDH-AES-CBC-CMAC-192	AES	192	CBC / CMAC
id-CA-ECDH-AES-CBC-CMAC-256	AES	256	CBC / CMAC

6.2.4 Протокольные блоки данных приложения

В зависимости от подлежащего использованию симметричного алгоритма имеются два варианта реализации аутентификации чипа.

- Для реализации аутентификации чипа с использованием безопасного обмена сообщениями на основе 3DES ПРИМЕНЯЕТСЯ следующая команда:

  1. MSE:Set KAT.

- Для аутентификации чипа с использованием безопасного обмена сообщениями на основе AES ПРИМЕНЯЕТСЯ и МОЖЕТ быть использована для осуществления аутентификации чипа с применением безопасного обмена сообщениями на основе 3DES следующая последовательность команд:

  1. MSE:Set AT;
  2. GENERAL AUTHENTICATE.

6.2.4.1 Варианты реализации с использованием команды MSE:Set KAT

Примечание. Команда MSE:Set KAT может использоваться только для `id-CA-DH-3DES-CBC-CBC` *и* `id-CA-ECDH-3DES-CBC-CBC`*, т. е. безопасный обмен сообщениями ограничен использованием 3DES.*

Команда			
CLA		В зависимости от контекста	
INS	0x22	Управление средствами защиты	
P1/P2	0x41A6	Установка шаблона согласования ключей для вычисления	
Данные	0x91	*Эфемерный открытый ключ* Эфемерный открытый ключ $PK_{DH,IFD}$ (см. раздел 9.4.5), закодированный как простое значение открытого ключа	ОБЯЗАТЕЛЬНЫЕ
	0x84	*Ссылочные данные закрытого ключа* Этот объект данных является ОБЯЗАТЕЛЬНЫМ, если закрытый ключ неоднозначен, т. е. для аутентификации чипа имеется несколько пар ключей (см. разделы 6.2 и 9.2.6)	УСЛОВНО ОБЯЗАТЕЛЬНЫЕ
Ответ			
Данные	–	Отсутствуют	
Байты состояния	0x9000	*Нормальная операция* Операция согласования ключей успешно выполнена. Новые сеансовые ключи выработаны	
	0x6A80	*Неправильные параметры в поле данных команды* Сбой валидации эфемерного открытого ключа	
	прочее	*Ошибка, зависящая от операционной системы* Установленные ранее сеансовые ключи остаются действительными	

6.2.4.2 Вариант реализации с использованием команд MSE:Set AT и GENERAL AUTHENTICATE

1. MSE:Set AT. Команда MSE:Set AT используется для выбора и инициализации протокола. Использование команды MSE:SetAT для аутентификации чипа обозначается идентификатором объекта аутентификации чипа (см. разделы 6.2.3 и 9.2.7), используемого в качестве ссылочных данных криптографического механизма с тегом 0x80, см. таблицу ниже.

Команда			
CLA		В зависимости от контекста	
INS	0x22	Управление средствами защиты	
P1/P2	0x41A4	*Аутентификация чипа* Установление шаблона аутентификации для внутренней аутентификации	
Данные	0x80	*Ссылочные данные криптографического механизма* Идентификатор объекта подлежащего выбору протокола (только значение, тег 0x06 опускается)	ОБЯЗАТЕЛЬНЫЕ
	0x84	*Ссылочные данные закрытого ключа* Этот объект данных является ОБЯЗАТЕЛЬНЫМ для указания подлежащего использованию идентификатора закрытого ключа, если закрытый ключ является неоднозначным, т. е. для аутентификации чипа имеется несколько закрытых ключей	УСЛОВНО ОБЯЗАТЕЛЬНЫЕ
Ответ			
Данные	–	Отсутствуют	
Байты состояния	0x9000	*Нормальная операция* Протокол выбран и инициализирован	
	0x6A80	*Неправильные параметры в поле данных команды* Алгоритм не поддерживается или сбой инициализации	
	0x6A88	*Ссылочные данные не найдены* Ссылочные данные (т. е. закрытый ключ) отсутствуют	
	прочее	*Ошибка, зависящая от операционной системы* Сбой инициализации протокола	

Примечание. Некоторые операционные системы принимают выбор отсутствующего ключа и сообщают об ошибке, только когда данный ключ используется для целей выбора.

2. Команда GENERAL AUTHENTICATE. Команда GENERAL AUTHENTICATE используется для аутентификации чипа.

Команда				
CLA		В зависимости от контекста		
INS	0x86	GENERAL AUTHENTICATE		
P1/P2	0x0000	Ключи и протокол, известные в виде подразумеваемых		
Данные	0x7C	*Динамические аутентификационные данные* Объекты данных, присущие конкретному протоколу		ОБЯЗАТЕЛЬНЫЕ
		0x80	Эфемерный открытый ключ	
Ответ				
Данные	0x7C	*Динамические аутентификационные данные* Объекты данных, присущие конкретному протоколу		ОБЯЗАТЕЛЬНЫЕ
Байты состоя-ния	0x9000	*Нормальная операция* Протокол (этап) был успешным		
	0x6300	*Сбой аутентификации* Сбой протокола (этапа)		
	0x6A80	*Неправильные параметры в поле данных* Представленные данные недействительны		
	0x6A88	*Ссылочные данные не найдены* Ссылочные данные (т. е. закрытый ключ) отсутствуют		
	прочее	*Ошибка, зависящая от операционной системы* Сбой протокола (этапа)		

Примечание. Открытые ключи для аутентификации чипа, поддерживаемые чипом, находятся в объекте защиты (см. раздел 9.2.11). Если поддерживаются несколько открытых ключей, терминал ДОЛЖЕН выбрать соответствующий закрытый ключ чипа для использования в рамках команды MSE:Set AT.

6.2.4.3 Эфемерный открытый ключ

Эфемерные открытые ключи (см. раздел 9.4.5) КОДИРУЮТСЯ в виде точки эллиптической кривой (ECDH) или неподписанного целого числа (DH).

7. ДОПОЛНИТЕЛЬНЫЕ МЕХАНИЗМЫ КОНТРОЛЯ ДОСТУПА

Личными данными, хранящимися на бесконтактной ИС, которые определяются как обязательный минимум для обеспечения глобальной интероперабельности, являются МСЗ и изображение лица владельца, хранящееся в цифровой форме. Оба элемента могут также просматриваться (считываться) визуально после того, как электронный МСПД открыт и предоставлен для проверки.

Помимо хранящегося цифрового изображения лица, как основного биометрического параметра для обеспечения глобальной интероперабельности, ИКАО одобряет использование хранящихся цифровых изображений пальцев и/или радужной оболочки глаза в дополнение к изображению лица. Для внутреннего или двустороннего использования государства МОГУТ предпочесть хранить шаблоны и/или МОГУТ ограничивать доступ или шифровать эти данные, и такое решение принимается самими государствами.

Доступ к этим более конфиденциальным личным данным СЛЕДУЕТ ограничивать в большей степени. В разделе 7.1 аутентификация терминала определяется в качестве интероперабельного механизма, предназначенного для обеспечения расширенного контроля доступа. Хотя эти варианты упоминаются в этом разделе, ИКАО в настоящее время не предлагает и не определяет каких-либо спецификаций или практических методов в этих сферах. Если обеспечивать интероперабельность не требуется, то могут использоваться другие механизмы.

7.1 Аутентификация терминала

Механизм аутентификации терминала является УСЛОВНО ОБЯЗАТЕЛЬНЫМ. Для приложений LDS2 реализация является ОБЯЗАТЕЛЬНОЙ. Аутентификация терминала МОЖЕТ использоваться для защиты дополнительных биометрических данных в приложении электронного МСПД.

Протокол аутентификации терминала представляет собой двухходовый протокол "запрос-ответ", обеспечивающий однозначную, одностороннюю аутентификацию терминала. Этот протокол основан на расширенном контроле доступа, как предусмотрено в документе [TR-03110]. Если этот протокол поддерживается ИС, она ДОЛЖНА поддерживать аутентификацию чипа или PACE с отображением для аутентификации чипа.

Данный протокол позволяет ИС удостовериться в том, что терминалу разрешен доступ к конфиденциальным данным. Поскольку в конечном итоге терминал может получить доступ к конфиденциальным данным, все дальнейшее коммуникационное взаимодействие ДОЛЖНО быть соответствующим образом защищено. В этой связи в рамках аутентификации терминала также проводится аутентификация выбранного терминалом эфемерного открытого ключа, который использовался для обеспечения безопасного обмена сообщениями с аутентификацией чипа или PACE с отображением для аутентификации чипа. ИС ДОЛЖНА увязывать права доступа терминала с безопасным обменом сообщениями, установленным аутентифицированным эфемерным открытым ключом терминала.

ИС МОЖЕТ поддерживать аутентификацию терминала в мастер-файле и/или приложении электронного МСПД. Если аутентификация терминала используется для защиты групп данных в других приложениях, не являющихся приложениями МСПД, ИС ДОЛЖНА поддерживать аутентификацию терминала в мастер-файле.

Примечание. Если необходимо обеспечить совместимость с расширенным контролем доступа Европейского союза [TR-03110], то ИС ДОЛЖНА поддерживать аутентификацию терминала в приложении электронного МСПД.

7.1.2 Спецификация протоколов

Терминал и ИС выполняют следующие этапы:

1. Терминал посылает на ИС цепочку сертификатов. Эта цепочка начинается переменной сертификата, проверяемого открытым ключом CVCA, хранимом на чипе, и заканчивается сертификатом терминала.

2. ИС проверяет сертификаты и извлекает открытый ключ PK_{IFD} терминала.

3. ИС произвольно выбирает запрос r_{IC} и посылает его в терминал.

4. В ответ терминал выдает подпись s_{IFD} = **Sign***(SK_{IFD}, $ID_{IC}||r_{IC}||$***Comp***($PK_{DH,IFD}$)).*

5. ИС проверяет, что верификация **Verify***(PK_{IFD} , s_{IFD}, ID_{IC} $||r_{IC}||$***Comp***(PKDH,IFD)) = true (является истиной)*.

Примечание. Ключ $PK_{DH,IFD}$ генерируется в ходе аутентификации чипа или выполнения PACE с отображением для аутентификации чипа. Если генерируются несколько ключей (например, аутентификация чипа выполняется после PACE с отображением для аутентификации чипа), то ДОЛЖЕН использоваться самый последний ключ.

В этом протоколе ID_{IC} является идентификатором ИС:

- Если используется BAC, то ID_{IC} является номером документа электронного МСПД, содержащимся в МСЗ, включая контрольную цифру.

- Если выполняется PACE, то ID_{IC} вычисляется с использованием эфемерного открытого ключа PACE интегральной схемы, т. е. ID_{IC} = **Comp***($PK_{DH,IC}$).*

Примечание. Для обеспечения возможности выполнения аутентификации терминала в MF прежде НЕОБХОДИМО успешно выполнить протокол PACE.

Упрощенная версия показана ниже:

ИС (чип)		**IFD (система проверки)**
Произвольный выбор r_{IC}		
	— r_{IC} →	
		s_{IFD} = **Sign**(SK_{IFD},$ID_{IC}||r_{IC}||$**Comp***($PK_{DH,IFD}$))
	← s_{IFD} —-	
Verify*(PK_{IFD} , s_{IFD}, $ID_{IC}||r_{IC}||$***Comp***($PK_{DH,IFD}$)) = true*		

Рис. 4. Аутентификация терминала

7.1.3 Статус защиты

Если аутентификация терминала выполнена успешно, то ИС ПРЕДОСТАВЛЯЕТ доступ к хранимым конфиденциальным данным в соответствии с эффективной авторизацией аутентифицированного терминала. Если эффективная авторизация не предоставляет прав доступа к каким-либо данным, то ИС ДОЛЖНА отказать в выборе этого приложения.

Однако ИС ОГРАНИЧИВАЕТ права доступа терминала к процессу защищенного обмена сообщениями, установленному аутентифицированным эфемерным открытым ключом, т. е. эфемерным открытым ключом, предоставленным терминалом в рамках аутентификации чипа или выполнения PACE с отображением для аутентификации чипа. В течение одного и того же сеанса ИС НЕ ДОЛЖНА допускать выполнения более одной аутентификации терминала (в отношении определения термина "сеанс" см. разделы 9.8.1 и 9.8.3).

Примечание 1. Права доступа действительны до тех пор, пока осуществляется процесс безопасного обмена сообщениями, установленный эфемерными открытыми ключами, поэтому выбор или отмена выбора приложений на статус защиты не влияет.

Примечание 2. Аутентификация терминала на безопасный обмен сообщениями не влияет. Чип электронного МСПД СОХРАНЯЕТ безопасный обмен сообщениями даже в случае неудачной аутентификации терминала (за исключением случаев возникновения ошибок при безопасном обмене сообщениями).

7.1.4 Криптографические спецификации

7.1.4.1 Аутентификация терминала с использованием RSA

Для аутентификации терминала с использованием RSA ДОЛЖНЫ применяться следующие алгоритмы и форматы.

7.1.4.1.1 Алгоритм подписи

ИСПОЛЬЗУЕТСЯ RSA [RFC-3447], [PKCS#1], как указано в таблице 7.

Таблица 7. Идентификаторы объектов для аутентификации терминалов с использованием RSA

OID	**Подпись**	**Хэш**	**Параметры**
id-TA-RSA-PSS-SHA-256	RSASSA-PSS	SHA-256	По умолчанию
id-TA-RSA-PSS-SHA-512	RSASSA-PSS	SHA-512	По умолчанию

Параметры по умолчанию, подлежащие использованию с RSA-PSS, определяются следующим образом:

- Алгоритм хэширования. Алгоритм хэширования выбирается в соответствии с таблицей 7.
- Алгоритм генерирования маски. MGF1 [RFC-3447], [PKCS#1], использующий выбранный алгоритм хэширования.

- Длина затравки. Длина октета выходных данных выбранного алгоритма хэширования.
- Поле концевика: 0xBC

7.1.4.1.2 Формат открытого ключа

ИСПОЛЬЗУЕТСЯ формат TLV-Format [ИСО/МЭК 7816-8], описание которого приводится в части 12 документа Doc 9303.

- Идентификатор объекта БЕРЕТСЯ из таблицы 7.
- Длина бита модуля СОСТАВЛЯЕТ 2048 или 3072.
- Длина биты экспоненты СОСТАВЛЯЕТ не более 32.

7.1.4.1.3 Сжатие открытого ключа

Сжатый эфемерный открытый ключ терминала **Comp**($PK_{DH,IFD}$) определяется как хэш SHA-1 открытого значения DH, т. е. как октетная строка фиксированной длины, составляющей 20.

7.1.4.2 Аутентификация терминала с использованием алгоритма цифровой подписи на основе эллиптических кривых (ECDSA)

Для аутентификации терминала с использованием ECDSA ДОЛЖНЫ применяться следующие алгоритмы.

7.1.4.2.1 Алгоритм подписи

ИСПОЛЬЗУЕТСЯ ECDSA с простым форматом подписи согласно документу [TR-03111], как указано в таблице 8.

Таблица 8. Идентификаторы объектов для аутентификации терминалов с использованием ECDSA

OID	**Подпись**	**Хэш**
id-TA-ECDSA-SHA-224	ECDSA	SHA-224
id-TA-ECDSA-SHA-256	ECDSA	SHA-256
id-TA-ECDSA-SHA-384	ECDSA	SHA-384
id-TA-ECDSA-SHA-512	ECDSA	SHA-512

7.1.4.2.2 Формат открытого ключа

ИСПОЛЬЗУЕТСЯ формат TLV-Format [ИСО/МЭК 7816-8], описание которого приводится в части 12 документа Doc 9303.

- Идентификатор объекта БЕРЕТСЯ из таблицы 8.
- Длина бита кривой СОСТАВЛЯЕТ 224, 256, 320, 384 или 512.
- Параметры домена СООТВЕТСТВУЮТ требованиям документа [TR-03111].

7.1.4.2.3 Сжатие открытого ключа

Сжатый эфемерный открытый ключ **Comp**($PK_{DH,IFD}$) определяется как x-координата открытой точки ECDH, т. е. как октетная строка фиксированной длины ⌈log256p⌉.

7.1.4.3 Валидация сертификата

Для валидации сертификата терминала ИС НЕОБХОДИМО предоставить цепочку сертификатов, начинающуюся в объекте доверия, информация о котором хранится на ИС. Эти объекты доверия в той или иной степени представляют собой актуальные открытые ключи CVCA ИС.

7.1.4.3.1 Первоначальное состояние информации об объекте(ах) доверия, хранимой на ИС

На этапе производства или (предварительной) персонализации первоначальная информация об объекте(ах) доверия надежно хранится в памяти ИС.

Агент по (предварительной) персонализации:

- устанавливает текущую дату ИС на дату (предварительной) персонализации;
- персонализирует ключ CVCA с использованием в качестве объекта доверия самой последней даты вступления в силу.

В качестве объекта доверия агент по предварительной персонализации МОЖЕТ дополнительно персонализировать предыдущий ключ CVCA.

7.1.4.3.2 Связующие сертификаты

Поскольку с течением времени пара ключей, используемых CVCA, изменяется, необходимо оформлять связующие сертификаты. Связующие сертификаты CVCA ДОЛЖНЫ подписываться с использованием предыдущего ключа CVCA, т. е. ключа CVCA с наиболее актуальной датой вступления в силу. Для внутреннего обновления информации об объекте(ах) доверия в соответствии с полученными действующими связующим сертификатами ТРЕБУЕТСЯ ИС.

ИС ДОЛЖНА располагать возможностью хранения информации вплоть до двух объектах доверия.

Примечание. Учитывая необходимость координации деятельности в области связующих сертификатов CVCA (см. часть 12 документа Doc 9303), на ИС должна храниться информация максимум о двух объектах доверия.

7.1.4.3.3 Текущая дата

ИС ДОЛЖНА принимать связующие сертификаты CVCA с истекшим сроком действия, но НЕ ДОЛЖНА принимать сертификаты DV и терминалов с истекшим сроком действия. Для определения факта истечения срока действия сертификата ИС ИСПОЛЬЗУЕТ текущую дату.

Текущая дата. Если у интегральной схемы нет внутренних часов, то текущая дата аппроксимируется, как указано ниже. Эта дата аппроксимируется ИС самостоятельно с использованием самой последней даты вступления сертификата в силу, содержащейся в действующем связующем сертификате CVCA, сертификате DV или точном сертификате терминала.

Точный сертификат терминала. Сертификат терминала является точным, если ИС санкционирует выдачу верификатором документа сертификатов терминала с правильной датой вступления сертификата в силу. Связующие сертификаты CVCA, сертификаты DV и сертификаты терминалов, выданные местным DV, РАССМАТРИВАЮТСЯ ИС в качестве точных. Другие сертификаты в качестве точных рассматриваться НЕ ДОЛЖНЫ.

Терминал МОЖЕТ посылать ИС связующие сертификаты CVCA, сертификаты DV и сертификаты терминалов для обновления текущей даты и информации об объекте доверия, хранимой на ИС, даже в том случае, когда терминал не планирует или не способен продолжать аутентификацию сертификата.

Примечание. ИС проверяет только то, что, судя по всему, сертификат является актуальным (т. е. в части касающейся аппроксимированной текущей даты), за исключением случаев, когда в ИС имеются внутренние часы.

7.1.4.3.4 Общая процедура валидации

Процедура валидации сертификатов состоит из трех этапов:

1. **Верификация сертификата.** Подпись ДОЛЖНА быть действительной и, если этот сертификат не является связующим сертификатом CVCA, то срок его действия НЕ ДОЛЖЕН быть истекшим. Если верификацию выполнить не удается, то выполнение процедуры ПРЕКРАЩАЕТСЯ.

   Примечание. Ситуация, в которой срок действия связующего сертификата CVCA истек, может возникнуть в том случае, когда ИС использует источник времени без учета аппроксимированной текущей даты, описание которой приведено выше.

2. **Обновление внутреннего статуса.** Для верификации сертификатов DV текущая дата ДОЛЖНА *обновляться*, открытый ключ и атрибуты (включая соответствующие расширения сертификатов) ДОЛЖНЫ импортироваться, новые объекты доверия ДОЛЖНЫ *активироваться*, а объекты доверия, срок действия которых истек, ДОЛЖНЫ *признаваться недействительными.*

3. **Очистка.** Чип ПРЕДОСТАВЛЯЕТ информацию максимум о двух объектах доверия на приложение. Если после обновления внутреннего статуса задействованными остаются более двух объектов доверия на приложение, то объект доверия с наименее актуальной датой вступления в силу ПРИЗНАЕТСЯ *НЕДЕЙСТВИТЕЛЬНЫМ*.

Операция *обновления* текущей даты и операции *активирования* и *признания недействительности* объектов доверия ДОЛЖНЫ выполняться в качестве атомарных операций.

Активация объекта доверия. Новый объект доверия ВНОСИТСЯ в список объектов доверия.

Признание объекта доверия недействительным. Для верификации сертификатов DV объекты доверия с истекшим сроком действия использоваться НЕ ДОЛЖНЫ. В случае ИС, когда текущая дата может опережать дату истечения срока действия объекта доверия, например, в случае ИС, использующих внутренние часы, объекты с истекшим сроком действия ДОЛЖНЫ оставаться пригодными для верификации связующих сертификатов CVCA. После успешного импорта последующего связующего сертификата объекты доверия, признанные недействительными, МОГУТ быть исключены.

7.1.4.3.5 Пример процедуры валидации

Представляемая ниже в качестве примера процедура валидации МОЖЕТ использоваться для валидации цепи сертификатов. В отношении каждого полученного сертификата ИС выполняет следующие этапы:

1. ИС проверяет подпись на сертификате. Если подпись неправильная, то верификация считается невыполненной.

2. Если сертификат не является связующим сертификатом CVCA, то дата истечения срока действия сертификата сравнивается с текущей датой ИС. Если дата истечения срока действия предшествует текущей дате, то верификация считается невыполненной.

3. Сертификат признается действительным, а содержащиеся в сертификате открытый ключ и атрибуты (включая соответствующие расширения сертификатов) импортируются.

   - Для сертификатов CVCA, DV и точных сертификатов терминалов: дата вступления сертификата в силу сравнивается с текущей датой ИС. Если текущая дата предшествует дате вступления в силу, то текущая дата обновляется и заменяется на дату вступления в силу.

   - Для связующих сертификатов CVCA: новый открытый ключ CVCA вносится в список объектов доверия, надежно хранимый в памяти ИС. Затем новый объект доверия активируется.

   - Для сертификатов DV и терминалов: новый открытый ключ DV или терминала временно импортируется для последующей верификации сертификата или аутентификации терминала соответственно.

4. Объекты доверия с истекшими сроками действия, надежно хранимые в памяти ИС, признаются недействительными для верификации сертификатов DV и из перечня объектов доверия могут быть исключены.

7.1.4.3.6 Эффективная авторизация

В каждом сертификате СОДЕРЖИТСЯ шаблон авторизации владельца сертификата и МОГУТ содержаться расширения авторизации (см. раздел 7.2.2.6 части 12 документа Doc 9303).

- Шаблон авторизации владельца сертификата идентифицирует тип терминала (в этой спецификации рассматриваются только системы проверки, однако в других спецификациях могут использоваться иные типы терминалов).

- Шаблон авторизации владельца сертификата и расширения авторизации определяют *относительную авторизацию* владельца сертификата, присвоенную полномочным органом, выдающим сертификаты.

Для определения *эффективной авторизации* владельца сертификата ИС ДОЛЖНА вычислить побитовое булевое "И" (and) относительной авторизации, содержащееся в сертификате терминала, зарегистрированном сертификате DV и зарегистрированном сертификате CVCA.

Эффективная авторизация ИНТЕРПРЕТИРУЕТСЯ ИС следующим образом:

- Эффективную роль играет CVCA:
  - Этот связующий сертификат выдан национальным CVCA.
  - ИС ДОЛЖНА обновлять информацию о своем внутреннем объекте доверия, то есть информацию об открытом ключе и эффективной авторизации.
  - Орган, выдающий сертификат, является достоверным источником времени и ИС ДОЛЖНА обновлять текущую дату, используя дату вступления сертификата в силу.
  - ИС НЕ ДОЛЖНА давать доступ CVCA к конфиденциальным данным (т. е. эффективную авторизацию СЛЕДУЕТ игнорировать).
- Эффективную роль играет DV:
  - Уполномоченному DV сертификат выдан национальным CVCA.
  - Орган, выдающий сертификат, является достоверным источником времени и ИС ДОЛЖНА обновлять текущую дату, используя дату вступления сертификата в силу.
  - ИС НЕ ДОЛЖНА давать доступ DV к конфиденциальным данным (т. е. эффективную авторизацию СЛЕДУЕТ игнорировать).
- Эффективную роль играет терминал:
  - Сертификат выдан национальным или зарубежным DV.
  - Если сертификат является точным сертификатом терминала (см. раздел 7.1.4.3.3), то орган, выдающий сертификаты, является достоверным источником времени и ИС ДОЛЖНА обновлять текущую дату, используя дату вступления сертификата в силу.
  - ИС ДОЛЖНА предоставлять аутентифицированному терминалу доступ к конфиденциальным данным в соответствии с эффективной авторизацией.

Примечание. Шаблон авторизации владельца сертификата и расширения авторизации могут содержать биты, не распределенные праву доступа (биты RFU). В ходе оценки прав доступа ИС ДОЛЖНА эти биты игнорировать.

7.1.4.3.7 Импорт открытых ключей

Открытые ключи, импортированные посредством реализации процедуры валидации сертификатов, хранятся в ИС *постоянно или временно*.

ИС СЛЕДУЕТ отказываться от импорта открытого ключа, если ИС уже известны ссылочные данные владельца сертификата.

Постоянный импорт. Открытые ключи, содержащиеся в связующих сертификатах CVCA ПОСТОЯННО ИМПОРТИРУЮТСЯ ИС и ДОЛЖНЫ надежно храниться в памяти ИС. Постоянно импортируемый открытый ключ и его метаданные ВЫПОЛНЯЮТ следующие условия:

- После истечения срока действия он МОЖЕТ быть заменен последующим постоянно импортируемым открытым ключом.

- Он ДОЛЖЕН быть заменен последующим постоянно импортируемым открытым ключом с теми же ссылочными данными владельца сертификата, или в импорте ДОЛЖНО быть отказано.

- Он НЕ ДОЛЖЕН заменяться временно импортируемым открытым ключом.

Активация или признание недействительным постоянно импортируемого открытого ключа ДОЛЖНЫ представлять собой атомарную операцию.

Временный импорт. ИС ВРЕМЕННО импортирует открытые ключи, содержащиеся в сертификатах DV и терминалов. Временно импортированный открытый ключ и его метаданные ВЫПОЛНЯЮТ следующие условия:

- Он НЕ ЯВЛЯЕТСЯ выбираемым или пригодным для использования после отключения питания ИС.

- Он ДОЛЖЕН оставаться пригодным для использования до тех пор, пока успешно не будет завершена последующая криптографическая операция (т. е. PSO: проверка сертификата или внешняя аутентификация).

- Он МОЖЕТ быть заменен последующим временно импортированным открытым ключом.

Терминал НЕ ДОЛЖЕН использовать какой-либо временно импортированный открытый ключ, кроме ключа, импортированного в последний раз.

Импортируемые метаданные. Для каждого постоянно или временно импортированного открытого ключа ДОЛЖНЫ храниться следующие дополнительные данные, содержащиеся в сертификате (см. часть 12 документа Doc 9303):

- Ссылочные данные владельца сертификата

- Авторизация владельца сертификата (эффективная роль и эффективная авторизация)

- Дата вступления сертификата в силу

- Дата истечения срока действия сертификата

- Расширения сертификата (при необходимости)

Описание порядка оценки эффективной роли (CVCA, DV или терминал) и эффективной авторизации владельца сертификата приводятся в разделе 7.1.4.3.6.

Примечание. Формат хранимых данных зависит от операционной системы и рамками настоящих спецификаций не охватывается.

7.1.5 Блоки данных прикладного протокола

Для аутентификации терминала в процессе безопасного обмена сообщениями ИСПОЛЬЗУЕТСЯ следующая последовательность команд:

- MSE:Set DST
- PSO:Verify Certificate (верификация сертификата)
- MSE:Set AT
- Get Challenge (получение запроса)
- External Authenticate (внешняя аутентификация)

Реализация этапов 1 и 2 повторяется для каждого сертификата CV, подлежащего верификации (связующие сертификаты CVCA, сертификат DV, сертификат терминала).

7.1.5.1 MSE:Set DST

Команда MSE:Set DST используется для запуска процессами верификации сертификата.

Команда			
CLA		В зависимости от контекста	
INS	0x22	Управление средствами защиты	
P1/P2	0x81B6	Установление шаблона цифровой записи для верификации	
Данные	0x83	*Ссылочные данные открытого ключа* Имя подлежащего установке открытого ключа, закодированное в соответствии с ИСО 8859-1	ОБЯЗАТЕЛЬНЫЕ

Ответ		
Данные	–	Отсутствуют
Байты состояния	0x9000 0x6A88 Прочее	*Нормальная операция* Выбран ключ для заданной цели. *Ссылочные данные не найдены* Выбор не состоялся ввиду отсутствия открытого ключа. *Ошибка, зависящая от операционной системы* Ключ не выбран

Примечание. Некоторые операционные системы принимают выбор открытого ключа, не отвечающего требованиям, и сообщают об ошибке только после использования ключа для реализации выбранной цели.

7.1.5.2 PSO:Verify Certificate

Команда PSO:Verify Certificate используется для верификации и импорта сертификатов.

Команда			
CLA		В зависимости от контекста	
INS	0x2A	Выполнение операций по обеспечению защиты	
P1/P2	0x00BE	Верификация сертификата, не требующего дополнительного описания	
Данные	0x7F4E 0x5F37	*Тело сертификата* *Тело сертификата*, *подлежащего верификации* *Подпись* *Подпись сертификата, подлежащего верификации*	ОБЯЗАТЕЛЬНЫЕ ОБЯЗАТЕЛЬНЫЕ

Ответ		
Данные	–	Отсутствуют
Байты состояния	0x9000 Прочее	*Обычная обработка* Сертификат успешно валидирован, а открытый ключ – импортирован. *Ошибка, зависящая от операционной системы* Импортировать открытый ключ не представилось возможным (например, сертификат не был принят).

7.1.5.3 MSE:Set AT

Использование команды MSE:Set AT для аутентификации терминала указывается установкой P1/P2 на 0x81A4, см. таблицу ниже.

Команда			
CLA		В зависимости от контекста	
INS	0x22	Управление средствами защиты	
P1/P2	0x81A4	Аутентификация терминала:	
Данные	0x83	*Ссылочные данные открытого ключа / секретный ключ* Этот объект данных используется для выбора открытого ключа терминала посредством его имени, закодированного согласно ИСО 8859-1	ОБЯЗАТЕЛЬНЫЕ

Ответ		
Данные	–	Отсутствуют
Байты состояния	0x9000 0x6A80 0x6A88 Прочее	*Нормальная обработка* *Протокол выбран и инициализирован.* *Неправильные параметры в поле данных команды* Алгоритм не поддерживается или инициализация не выполнена. *Ссылочные данные не найдены* Ссылочные данные не доступны. *Ошибка, зависящая от операционной системы* Инициализация протокола не выполнена.

Примечание. Некоторые операционные системы принимают выбор открытого ключа, не отвечающего требованиям, и сообщают об ошибке только после использования ключа для реализации выбранной цели.

7.1.5.4 Get Challenge

Команда			
CLA		В зависимости от контекста	
INS	0x84	Get Challenge	
P1/P2	0x0000		
Данные	–	Отсутствуют	
Le	0x08		ОБЯЗАТЕЛЬНЫЕ

Ответ		
Данные	r_{IC}	8 байтов случайности
Байты состояния	0x9000 Прочее	*Нормальная обработка* *Ошибка, зависящая от операционной системы*

7.1.5.5 Внешняя аутентификация

Команда			
CLA		В зависимости от контекста	
INS	0x82	Внешняя аутентификация	
P1/P2	0x0000	Ключи и алгоритмы потенциально известны.	
Данные		Подпись, сгенерированная терминалом.	ОБЯЗАТЕЛЬНЫЕ

Ответ		
Данные	–	Отсутствуют
Байты состояния	0x9000 0x6300 0x6982 Прочее	*Нормальная обработка* Аутентификация выполнена успешно. Доступ к группам данных будет предоставляться в соответствии с эффективной авторизацией соответствующего верифицированного сертификата. *Предупреждение* Верификация подписи не выполнена. *Неудовлетворительный статус защиты* Аутентификация не выполнена, поскольку текущий уровень аутентификации терминала не позволяет использовать аутентификацию терминала (например, аутентификация терминала уже выполнена, и т. д.). *Ошибка, зависящая от операционной системы* Аутентификация не выполнена.

7.2 Шифрование дополнительных биометрических параметров

Ограничение доступа к дополнительным биометрическим параметрам МОЖЕТ также производиться путем их шифрования. Чтобы иметь возможность расшифровать зашифрованные данные, система проверки ДОЛЖНА обеспечиваться ключом дешифрования. Определение алгоритма шифрования/расшифровки и ключей, подлежащих использованию, осуществляется по усмотрению внедряющего государства и выходит за рамки настоящего документа.

Осуществление защиты дополнительных биометрических параметров зависит от внутренних спецификаций государства или спецификаций, согласованных на двусторонней основе между государствами, обменивающимися такой информацией.

8. СИСТЕМА ПРОВЕРКИ

В целях обеспечения выполнения требуемых функций и определенных вариантов, внедряемых на предоставляемых электронных МСПД, система проверки должна удовлетворять некоторым предварительным условиям.

8.1 Базовый контроль доступа

Системы проверки, поддерживающие базовый контроль доступа, ДОЛЖНЫ удовлетворять следующим предварительным условиям:

1. Система проверки оснащена средствами считывания МСЗ с физического документа для выведения ключей доступа к документу (K_{Enc} и K_{MAC}) с электронного МСПД.

2. Программное обеспечение системы проверки поддерживает протокол, описываемый в разделе 4.3, в случае, когда системе предоставляется электронный МСПД с базовым контролем доступа, включая шифрование передачи данных с безопасным обменом сообщениями.

8.2 Установление соединения с аутентификацией паролем

Системы проверки, поддерживающие PACE, ДОЛЖНЫ удовлетворять следующим предварительным условиям:

1. Система проверки оснащена средствами считывания МСЗ и/или CAN с физического документа.

2. Программное обеспечение системы проверки поддерживает протокол, описываемый в разделе 4.4, в случае, когда системе предоставляется электронный МСПД с PACE, включая шифрование и передачу данных с безопасным обменом сообщениями.

8.3 Пассивная аутентификация

Для осуществления пассивной аутентификации данных, хранящихся на бесконтактной ИС электронных МСПД, система проверки должна знать ключевую информацию государств или организаций выдачи.

1. Сертификат подписывающегося CA страны каждого государства или организации выдачи или соответствующая информация, извлекаемая из сертификата, ХРАНЯТСЯ безопасным образом в системе проверки.

2. Как альтернатива, сертификаты лиц, подписывающих документы (C_{DS}), каждого государства или организации выдачи или соответствующая информация, извлекаемая из сертификатов, ХРАНЯТСЯ безопасным образом в системе проверки.

Прежде чем использовать открытый ключ подписывающегося CA государства или организации выдачи, принимающее государство или организация ДОЛЖНЫ быть уверенными в таком ключе.

Прежде чем использовать сертификат лица, подписывающего документы (C_{DS}) для верификации SO_D, система проверки ВЕРИФИЦИРУЕТ его цифровую подпись, используя открытый ключ подписывающегося CA страны.

Кроме того, системы проверки ИМЕЮТ доступ к верифицированной информации об отзывах.

8.4 Активная аутентификация

Поддержка активной аутентификации системами проверки является ФАКУЛЬТАТИВНОЙ.

Если система проверки поддерживает активную аутентификацию, ТРЕБУЕТСЯ, чтобы система проверки была способна считывать визуальную МСЗ.

Если система проверки поддерживает активную аутентификацию, то программное обеспечение систем проверки ПОДДЕРЖИВАЕТ протокол активной аутентификации, описание которого приведено в разделе 6.1.

8.5 Аутентификация чипа

Поддержка аутентификации чипа системами проверки является ФАКУЛЬТАТИВНОЙ.

Если система проверки поддерживает аутентификацию чипа, ТРЕБУЕТСЯ, чтобы система проверки была способна считывать визуальную МСЗ.

Если система проверки поддерживает аутентификацию чипа, то программное обеспечение систем проверки ПОДДЕРЖИВАЕТ протокол аутентификации чипа, описание которого приведено в разделе 6.2.

8.6 Аутентификация терминала

Поддержка аутентификации терминала системами проверки является ФАКУЛЬТАТИВНОЙ.

Если система проверки поддерживает аутентификацию терминала, ТРЕБУЕТСЯ, чтобы система проверки была способна надежно хранить открытый ключ системы проверки. Для возобновления сертификата терминала система проверки ДОЛЖНА иметь доступ к своему DV через регулярные интервалы.

Если система проверки поддерживают аутентификацию терминала, программные средства системы проверки ПОДДЕРЖИВАЮТ протокол аутентификации терминала, как указано в разделе 7.1

8.7 Расшифровка дополнительных биометрических параметров

Осуществление защиты факультативных дополнительных биометрических параметров зависит от внутренних спецификаций государства или спецификаций, согласованных на двусторонней основе между государствами, обменивающимися такой информацией.

9. ОБЩИЕ СПЕЦИФИКАЦИИ

9.1 Структуры ASN.1

Структуры данных "Информация об открытом ключе субъекта" и "Идентификатор алгоритма" определяются следующим образом:

```
SubjectPublicKeyInfo ::= SEQUENCE {
  algorithm  AlgorithmIdentifier,
  subjectPublicKey    BIT STRING
}

AlgorithmIdentifier ::= SEQUENCE {
  algorithm OBJECT IDENTIFIER,
  parameters ANY DEFINED BY algorithm OPTIONAL
}
```

Подробная информация о параметрах приводится в [X9.42] и [TR-03111].

9.2 Информация о поддерживаемых протоколах и поддерживаемых приложениях

Структура данных ASN.1 "Сведения о защите" (SecurityInfos) ПРЕДОСТАВЛЯЕТСЯ чипом электронного МСПД для указания поддерживаемых протоколов защиты. Эта структура данных приводится ниже:

```
SecurityInfos  ::= SET OF SecurityInfo

  SecurityInfo ::= SEQUENCE {
  protocol     OBJECT IDENTIFIER,
  requiredData ANY DEFINED BY protocol,
  optionalData ANY DEFINED BY protocol OPTIONAL
}
```

Элементы, содержащиеся в структуре данных "Сведения о защите" (SecurityInfo), имеют следующие значения:

- протокол идентификатора объекта идентифицирует поддерживаемый протокол;
- требуемые данные открытого типа содержат обязательные данные, относящиеся конкретно к этому протоколу;
- факультативные данные открытого типа содержат факультативные данные, относящиеся конкретно к этому протоколу.

Сведения о защите для PACE

Для указания поддержки PACE поле "Сведения о защите" может содержать следующие записи:

- ДОЛЖЕН присутствовать по крайней мере один элемент "Информация PACE", использующий стандартизированный параметр домена;
- для каждого поддерживаемого набора явных параметров домена ДОЛЖЕН присутствовать элемент "Информация о параметре домена для PACE".

Сведения о защите для активной аутентификации

Если для активной аутентификации чип электронного МСПД использует алгоритм подписи на основе ECDSA, информация о защите ДОЛЖНА содержать следующую запись в поле "Сведения о защите":

- ActiveAuthenticationInfo

Сведения о защите для аутентификации чипа

Для указания поддержки аутентификации чипа поле "Сведения о защите" может содержать следующие записи:

- ДОЛЖНЫ присутствовать по крайней мере информационная запись по аутентификации чипа (ChipAuthenticationInfo) и соответствующая информация об открытом ключе аутентификации чипа (ChipAuthenticationPublicKeyInfo) с использованием явных параметров домена.

Сведения о защите для аутентификации терминалов

Для указания поддержки аутентификации терминалов поле "сведения о защите" может содержать следующие записи:

- По крайней мере присутствует один элемент.

Сведения о защите для действующих приложений

В разделе 3.11.2 части 10 документа Doc 9303 содержится рекомендация о присутствии транспарентного элементарного файла EF.DIR для указания поддерживаемых приложений. Этот файл является обязательным, если присутствует какое-либо приложение LDS2. Поскольку файл EF.DIR не подписывается и, в этой связи, им можно манипулировать, например для сокрытия существующих приложений от IFD, защищенная копия файла EF.DIR предоставляется в качестве `SecurityInfo`, если присутствует какое-либо приложение LDS2.

Сведения о защите для других протоколов

Сведения о защите (`SecurityInfos`) МОГУТ содержать дополнительные записи, указывающие на поддержку других протоколов, или предоставляющие другую информацию. Система проверки МОЖЕТ не учитывать любую неизвестную запись.

9.2.1 Информация о PACE (PACEInfo)

Эта структура данных предоставляет подробную информацию о применении PACE.

- Идентификатор объекта `protocol` (протокол) ИДЕНТИФИЦИРУЕТ подлежащие использованию алгоритмы (т. е. согласование ключей, симметричный шифр и MAC).

- Целочисленный элемент `version` (версия) ИДЕНТИФИЦИРУЕТ версию протокола. Эта спецификация поддерживает только версию 2.

- Целочисленный элемент `parameterId` (ID параметра) используется для указания идентификатора параметра домена. Он ДОЛЖЕН использоваться, если чип электронного МСПД использует стандартизированные параметры домена (см. раздел 9.5.1), предоставляет множественные явные параметры домена для PACE, или если `protocol` является одним из идентификаторов объекта (OID) `*-CAM-*`. В случае PACE с отображением для аутентификации чипа `parameterID` также обозначает идентификатор используемого ключа аутентификации чипа, т. е. чип ДОЛЖЕН предоставить информацию об открытом ключе аутентификации чипа (`ChipAuthenticationPublicKeyInfo`) с идентификатором ключа (`keyID`), который равен ID ключа (`parameterID`) из этой структуры данных.

```
PACEInfo ::= SEQUENCE {
    protocol    OBJECT IDENTIFIER(
                id-PACE-DH-GM-3DES-CBC-CBC |
                id-PACE-DH-GM-AES-CBC-CMAC-128 |
                id-PACE-DH-GM-AES-CBC-CMAC-192 |
                id-PACE-DH-GM-AES-CBC-CMAC-256 |
                id-PACE-ECDH-GM-3DES-CBC-CBC |
                id-PACE-ECDH-GM-AES-CBC-CMAC-128 |
```

```
                id-PACE-ECDH-GM-AES-CBC-CMAC-192 |
                id-PACE-ECDH-GM-AES-CBC-CMAC-256 |
                id-PACE-DH-IM-3DES-CBC-CBC |
                id-PACE-DH-IM-AES-CBC-CMAC-128 |
                id-PACE-DH-IM-AES-CBC-CMAC-192 |
                id-PACE-DH-IM-AES-CBC-CMAC-256 |
                id-PACE-ECDH-IM-3DES-CBC-CBC |
                id-PACE-ECDH-IM-AES-CBC-CMAC-128 |
                id-PACE-ECDH-IM-AES-CBC-CMAC-192 |
                id-PACE-ECDH-IM-AES-CBC-CMAC-256
                id-PACE-ECDH-CAM-AES-CBC-CMAC-128 |
                id-PACE-ECDH-CAM-AES-CBC-CMAC-192 |
                id-PACE-ECDH-CAM-AES-CBC-CMAC-256),
    version     INTEGER, -- MUST be 2
    parameterId INTEGER OPTIONAL
}
```

9.2.2 Информация о параметрах домена для PACE (PACEDomainParameterInfo)

Эта структура данных является ОБЯЗАТЕЛЬНОЙ, если чип электронного МСПД предоставляет явные параметры домена для PACE, в противном случае она ДОЛЖНА пропускаться.

- Идентификатор объекта `protocol` (протокол) ИДЕНТИФИЦИРУЕТ тип параметров домена (т. е. DH или ECDH).
- Последовательность в поле `domainParameter` (параметр домена) СОДЕРЖИТ параметры домена.
- Целочисленный элемент `parameterId` (ID параметра) МОЖЕТ использоваться для указания местного идентификатора параметра домена. Он ДОЛЖЕН использоваться, если чип электронного МСПД предоставляет множественные явные параметры домена для PACE.

```
PACEDomainParameterInfo ::= SEQUENCE {
  protocol          OBJECT IDENTIFIER(
                    id-PACE-DH-GM |
                    id-PACE-ECDH-GM |
                    id-PACE-DH-IM |
                    id-PACE-ECDH-IM |
                    id-PACE-ECDH-CAM),
  domainParameter   AlgorithmIdentifier,
  parameterId       INTEGER OPTIONAL
}
```

Примечание. Чип электронного МСПД МОЖЕТ поддерживать более одного набора явных параметров домена (т. е. чип может поддерживать различные алгоритмы и/или ключи различной длины). В этом случае идентификатор ДОЛЖЕН быть получен из соответствующего поля `PACEDomainParameterInfo`. (*Информация о параметрах домена для PACE).*

Параметры домена, содержащиеся в поле `PACEDomainParameterInfo`, не защищены и могут быть небезопасными. Использование незащищенных параметров домена для PACE даст утечку информации об

используемом пароле. Чипы электронных МСПД ДОЛЖНЫ поддерживать по крайней мере один набор стандартизированных параметров домена, как указано в разделе 9.5.1. Системы проверки ДОЛЖНЫ использовать явные параметры домена, предоставляемые электронным МСПД, только в том случае, если системам проверки явно известно, что они защищены.

Обмен эфемерными открытыми ключами ДОЛЖЕН осуществляться как простыми значениями открытых ключей. Дополнительная информация о шифровании приводится в разделе 9.4.5.

9.2.3 Идентификатор объекта PACE

Идентификатор объекта, используемый в PACE, содержится в поддереве `bsi-de`:

```
bsi-de OBJECT IDENTIFIER ::= {
    itu-t(0) identified-organization(4) etsi(0)
    reserved(127) etsi-identified-organization(0) 7
}
```

ИСПОЛЬЗУЮТСЯ следующие идентификаторы объекта:

```
id-PACE OBJECT IDENTIFIER ::= {
    bsi-de protocols(2) smartcard(2) 4
}

id-PACE-DH-GM                    OBJECT IDENTIFIER ::= {id-PACE 1}
id-PACE-DH-GM-3DES-CBC-CBC       OBJECT IDENTIFIER ::= {id-PACE-DH-GM 1}
id-PACE-DH-GM-AES-CBC-CMAC-128   OBJECT IDENTIFIER ::= {id-PACE-DH-GM 2}
id-PACE-DH-GM-AES-CBC-CMAC-192   OBJECT IDENTIFIER ::= {id-PACE-DH-GM 3}
id-PACE-DH-GM-AES-CBC-CMAC-256   OBJECT IDENTIFIER ::= {id-PACE-DH-GM 4}

id-PACE-ECDH-GM                  OBJECT IDENTIFIER ::= {id-PACE 2}
id-PACE-ECDH-GM-3DES-CBC-CBC     OBJECT IDENTIFIER ::= {id-PACE-ECDH-GM 1}
id-PACE-ECDH-GM-AES-CBC-CMAC-128 OBJECT IDENTIFIER ::= {id-PACE-ECDH-GM 2}
id-PACE-ECDH-GM-AES-CBC-CMAC-192 OBJECT IDENTIFIER ::= {id-PACE-ECDH-GM 3}
id-PACE-ECDH-GM-AES-CBC-CMAC-256 OBJECT IDENTIFIER ::= {id-PACE-ECDH-GM 4}

id-PACE-DH-IM                    OBJECT IDENTIFIER ::= {id-PACE 3}
id-PACE-DH-IM-3DES-CBC-CBC       OBJECT IDENTIFIER ::= {id-PACE-DH-IM 1}
id-PACE-DH-IM-AES-CBC-CMAC-128   OBJECT IDENTIFIER ::= {id-PACE-DH-IM 2}
id-PACE-DH-IM-AES-CBC-CMAC-192   OBJECT IDENTIFIER ::= {id-PACE-DH-IM 3}
id-PACE-DH-IM-AES-CBC-CMAC-256   OBJECT IDENTIFIER ::= {id-PACE-DH-IM 4}

id-PACE-ECDH-IM                  OBJECT IDENTIFIER ::= {id-PACE 4}
id-PACE-ECDH-IM-3DES-CBC-CBC     OBJECT IDENTIFIER ::= {id-PACE-ECDH-IM 1}
id-PACE-ECDH-IM-AES-CBC-CMAC-128 OBJECT IDENTIFIER ::= {id-PACE-ECDH-IM 2}
id-PACE-ECDH-IM-AES-CBC-CMAC-192 OBJECT IDENTIFIER ::= {id-PACE-ECDH-IM 3}
id-PACE-ECDH-IM-AES-CBC-CMAC-256 OBJECT IDENTIFIER ::= {id-PACE-ECDH-IM 4}

id-PACE-ECDH-CAM                 OBJECT IDENTIFIER ::= {id-PACE 6}
id-PACE-ECDH-CAM-AES-CBC-CMAC-128 OBJECT IDENTIFIER ::= {id-PACE-ECDH-CAM 2}
id-PACE-ECDH-CAM-AES-CBC-CMAC-192 OBJECT IDENTIFIER ::= {id-PACE-ECDH-CAM 3}
id-PACE-ECDH-CAM-AES-CBC-CMAC-256 OBJECT IDENTIFIER ::= {id-PACE-ECDH-CAM 4}
```

9.2.4 Сведения об активной аутентификации (ActiveAuthenticationInfo)

Если для активной аутентификации чип электронного МСПД использует алгоритм подписи на основе ECDSA, SecurityInfos (информация о защите) в группе данных 14 LDS чипа электронного МСПД ДОЛЖНА содержать следующую запись в поле SecurityInfo:

```
ActiveAuthenticationInfo ::= SEQUENCE {
  protocol           OBJECT IDENTIFIER(id-icao-mrtd-security-aaProtocolObject),
  version            INTEGER, -- MUST be 1
  signatureAlgorithm OBJECT IDENTIFIER
}
id-icao-mrtd-security-aaProtocolObject OBJECT IDENTIFIER ::=
                                  { id-icao-mrtd-security 5 }
```

Для алгоритма подписи (signatureAlgorithm) ИСПОЛЬЗУЮТСЯ идентификаторы объекта, которые определены в документе [TR-03111].

Примечание. Идентификатор объекта `id-icao-mrtd-security` определен в части 10 документа Doc 9303.

9.2.5 Сведения об аутентификации чипа (ChipAuthenticationInfo)

Эта структура данных предоставляет подробную информацию о реализации механизма аутентификации чипа.

- Идентификатор объекта protocol (протокол) ИДЕНТИФИЦИРУЕТ подлежащие использованию алгоритмы (т. е. согласование ключей, симметричный шифр и MAC).
- Целочисленный элемент version (версия) ИДЕНТИФИЦИРУЕТ версию протокола. В настоящее время эта спецификация поддерживает только версию 1.
- Целочисленный элемент keyId (ID ключа) МОЖЕТ использоваться для указания местного идентификатора ключа. Он ДОЛЖЕН использоваться, если чип электронного МСПД предоставляет множественные открытые ключи для аутентификации чипа.

```
ChipAuthenticationInfo ::= SEQUENCE {
  protocol   OBJECT IDENTIFIER(
             id-CA-DH-3DES-CBC-CBC |
             id-CA-DH-AES-CBC-CMAC-128 |
             id-CA-DH-AES-CBC-CMAC-192 |
             id-CA-DH-AES-CBC-CMAC-256 |
             id-CA-ECDH-3DES-CBC-CBC |
             id-CA-ECDH-AES-CBC-CMAC-128 |
             id-CA-ECDH-AES-CBC-CMAC-192 |
             id-CA-ECDH-AES-CBC-CMAC-256),
  version    INTEGER, -- MUST be 1
  keyId      INTEGER OPTIONAL
}
```

9.2.6 Сведения об открытом ключе аутентификации чипа (ChipAuthenticationPublicKeyInfo)

Эта структура данных предоставляет открытый ключ для аутентификации чипа или для PACE с обозначением, используемым для аутентификации чипа электронного МСПД.

- Идентификатор объекта `protocol` (протокол) ИДЕНТИФИЦИРУЕТ чип открытого ключа (т. е. DH или ECDH).
- Последовательность в поле `chipAuthenticationPublicKey` СОДЕРЖИТ открытый ключ в закодированной форме.
- Целочисленный элемент `keyId` (ID ключа) МОЖЕТ использоваться для указания местного идентификатора ключа. Он ДОЛЖЕН использоваться, если чип электронного МСПД предоставляет множественные открытые ключи для аутентификации чипа или если этот ключ используется для PACE с использованием отображения для аутентификации чипа.

```
ChipAuthenticationPublicKeyInfo ::= SEQUENCE {
    protocol                    OBJECT IDENTIFIER(id-PK-DH | id-PK-ECDH),
    chipAuthenticationPublicKey SubjectPublicKeyInfo,
    keyId                       INTEGER OPTIONAL
}
```

Примечание. Чип электронного МСПД МОЖЕТ поддерживать более одной пары ключей аутентификации чипа (т. е. чип может поддерживать различные алгоритмы и/или ключи различной длины). В этом случае местный идентификатор ключа ДОЛЖЕН быть указан в соответствующем поле "Информация об аутентификации чипа" (`ChipAuthenticationInfo`) и поле "Информация об открытом ключе аутентификации чипа" (`ChipAuthenticationPublicKeyInfo`).

9.2.7 Идентификатор объекта аутентификации чипа

ИСПОЛЬЗУЕТСЯ следующий идентификатор объекта:

```
id-PK OBJECT IDENTIFIER ::= {
  bsi-de protocols(2) smartcard(2) 1
}
id-PK-DH                    OBJECT IDENTIFIER ::= {id-PK 1}
id-PK-ECDH                  OBJECT IDENTIFIER ::= {id-PK 2}

id-CA OBJECT IDENTIFIER ::= {
  bsi-de protocols(2) smartcard(2) 3
}

id-CA-DH                    OBJECT IDENTIFIER ::= {id-CA 1}
id-CA-DH-3DES-CBC-CBC       OBJECT IDENTIFIER ::= {id-CA-DH 1}
id-CA-DH-AES-CBC-CMAC-128   OBJECT IDENTIFIER ::= {id-CA-DH 2}
id-CA-DH-AES-CBC-CMAC-192   OBJECT IDENTIFIER ::= {id-CA-DH 3}
id-CA-DH-AES-CBC-CMAC-256   OBJECT IDENTIFIER ::= {id-CA-DH 4}

id-CA-ECDH                  OBJECT IDENTIFIER ::= {id-CA 2}
id-CA-ECDH-3DES-CBC-CBC     OBJECT IDENTIFIER ::= {id-CA-ECDH 1}
id-CA-ECDH-AES-CBC-CMAC-128 OBJECT IDENTIFIER ::= {id-CA-ECDH 2}
id-CA-ECDH-AES-CBC-CMAC-192 OBJECT IDENTIFIER ::= {id-CA-ECDH 3}
id-CA-ECDH-AES-CBC-CMAC-256 OBJECT IDENTIFIER ::= {id-CA-ECDH 4}
```

9.2.8 TerminalAuthenticationInfo

Эта структура данных предоставляет подробную информацию о реализации механизма аутентификации терминала.

- Идентификатор объекта `protocol` (протокол) ИДЕНТИФИЦИРУЕТ *общий* протокол аутентификации терминала, поскольку с течением времени конкретный протокол может измениться.
- Целочисленный элемент `version` (версия) ИДЕНТИФИЦИРУЕТ версию протокола. В настоящее время эта спецификация поддерживает только версию 1. Следует отметить, что более поздние версии документа [TR-03110] определяют версию 2 этого протокола, которая в рамках данной спецификации не рассматривается.

```
TerminalAuthenticationInfo ::= SEQUENCE {
   protocol OBJECT IDENTIFIER(id-TA),
   version  INTEGER        -- MUST be 1
 }
```

9.2.9 Идентификаторы объекта аутентификации терминала

ИСПОЛЬЗУЕТСЯ следующий идентификатор объекта:

```
id-TA OBJECT IDENTIFIER ::= {
  bsi-de protocols(2) smartcard(2) 2
}

id-TA-RSA               OBJECT IDENTIFIER ::= {id-TA 1}
id-TA-RSA-PSS-SHA-256   OBJECT IDENTIFIER ::= {id-TA-RSA 4}
id-TA-RSA-PSS-SHA-512   OBJECT IDENTIFIER ::= {id-TA-RSA 6}

id-TA-ECDSA             OBJECT IDENTIFIER ::= {id-TA 2}
id-TA-ECDSA-SHA-224     OBJECT IDENTIFIER ::= {id-TA-ECDSA 2}
id-TA-ECDSA-SHA-256     OBJECT IDENTIFIER ::= {id-TA-ECDSA 3}
id-TA-ECDSA-SHA-384     OBJECT IDENTIFIER ::= {id-TA-ECDSA 4}
id-TA-ECDSA-SHA-512     OBJECT IDENTIFIER ::= {id-TA-ECDSA 5}
```

9.2.10 EFDIRInfo

Эта структура данных формирует полную копию контента траспарентного элементарного файла EF.DIR, соддержащегося в мастер-файле.

```
  EFDIRInfo ::= SEQUENCE {
       protocol                        OBJECT IDENTIFIER(id-EFDIR),
       eFDIR                           OCTET STRING
  }

id-EFDIR OBJECT IDENTIFIER ::={
  id-icao-mrtd-security 13
}
```

9.2.11 Хранение на чипе

Для указания поддержки протокола и поддерживаемых параметров чип электронного МСПД ПРЕДОСТАВЛЯЕТ сведения о защите в транспарентных элементарных файлах (общая структура этих файлов приводится в части 10 документа Doc 9303):

- Файл EF.CardAccess, содержащийся в мастер-файле, является ОБЯЗАТЕЛЬНЫМ, если чип электронного МСПД поддерживает PACE, и СОДЕРЖИТ соответствующие сведения о защите, необходимые для PACE:
  - `PACEInfo;`
  - `PACEDomainParameterInfo.`
- Файл EF.CardSecurity, содержащийся в мастер-файле, является ОБЯЗАТЕЛЬНЫМ, если:
  - PACE с отображением для аутентификации чипа поддерживается чипом электронного МСПД , или
  - файл аутентификации терминала, содержащийся в мастер-файле, поддерживается чипом электронного МСПД, или
  - файл аутентификации чипа, содержащийся в мастер-файле, поддерживается электронным МСПД

  и СОДЕРЖИТ следующее поле `SecurityInfos`:
  - `ChipAuthenticationInfo`, как это требует аутентификация чипа
  - `ChipAuthenticationPublicKeyInfo`, как требуется для PACE-CAM/аутентификации чипа
  - `TermninalAuthenticationInfo`, как этого требует аутентификация терминала
  - `EFDIRInfo`, если на чипе помимо приложения электронного МСПД присутствуют другие приложения
  - поле `SecurityInfos`, содержащееся в файле EF.CardAccess.
- Файл EF.DG14, содержащийся в приложении электронного МСПД, является ОБЯЗАТЕЛЬНЫМ, если:
- PACE-GM/-IM поддерживается чипом электронного МСПД
- Аутентификация терминала в приложении электронного МСПД поддерживается чипом электронного МСПД, или
- Аутентификация чипа в приложении электронного МСПД поддерживается чипом электронного МСПД

и СОДЕРЖИТ следующее поле `SecurityInfos`:

- `ChipAuthenticationInfo`, как это требуется для аутентификации чипа
- `ChipAuthenticationPublicKeyInfo`, как это требуется для аутентификации чипа
- `TermninalAuthenticationInfo`, как того требует аутентификация терминала
- сведения о защите (`SecurityInfos`), содержащиеся в файле EF.CardAccess.

• Полный набор сведений о защите (включая сведения о защите, содержащиеся в файле EF.CardAccess, не указанные в документе Doc 9303) дополнительно ХРАНИТСЯ в файле EF.DG14 приложения электронного МСПД (см. часть 10 документа Doc 9303).

Эти файлы МОГУТ содержать дополнительные сведения о защите, не охватываемые данной спецификацией.

Примечание. Хотя аутентичность сведений о защите, хранящихся в EF.DG14 и файле EF.CardSecurity, защищается с помощью пассивной аутентификации, файл EF.CardAccess является незащищенным.

9.3 Блоки APDU

9.3.1 Увеличенная длина

В зависимости от размера криптографических объектов (например, открытые ключи, подписи) для посылки этих данных на чип электронного МСПД ДОЛЖНЫ использоваться блоки APDU с полями увеличенной длины. Подробная информация по увеличенной длине приводится в стандарте [ИСО/МЭК 7816-4].

9.3.1.1 Чипы электронных МСПД

Для чипов электронных МСПД поддержка увеличенной длины является УСЛОВНО ОБЯЗАТЕЛЬНОЙ. Если выбранные государством выдачи криптографические алгоритмы и размеры ключей требуют использования увеличенной длины, чипы электронных МСПД ПОДДЕРЖИВАЮТ увеличенную длину. Если чип электронного МСПД поддерживает увеличенную длину, это ДОЛЖНО быть указано в ATR/ATS или в файле EF.ATR/INFO, как оговорено в стандарте [ИСО/МЭК 7816-4].

9.3.1.2 Терминалы

Для терминалов поддержка увеличенной длины является ОБЯЗАТЕЛЬНОЙ. Прежде чем использовать этот вариант, терминалу СЛЕДУЕТ проверить, указана ли поддержка увеличенной длины в ATR/ATS или в файле EF.ATR/INFO чипа электронного МСПД. Терминал НЕ ДОЛЖЕН использовать увеличенную длину для блоков APDU, кроме нижеследующих команд, за исключением случаев, когда в ATR/ATS или в файле EF.ATR/INFO четко указаны размеры входного и выходного буфера чипа электронного МСПД.

- MSE:Set KAT;
- GENERAL AUTHENTICATE.

9.3.2 Составление последовательности команд

Для команды GENERAL AUTHENTICATE ДОЛЖНА использоваться определенная последовательность команд, чтобы увязать эту цепочку команд с выполнением протокола PACE. Такая последовательность команд НЕ ДОЛЖНА использоваться для иных целей, если только это четко не указано в чипе. Подробная информация по составлению последовательности команд содержится в стандарте [ИСО/МЭК 7816-4].

9.3.3 Объекты данных

Отправитель команды или ответа APDU ДОЛЖЕН передавать объекты данных в поле данных в порядке, определенном в описаниях APDU.

Примечание. Принятие объектов данных в любом порядке не требуется, однако для некоторых команд это повышает степень интероперабельности, например MSE:Set AT/GENERAL AUTHENTICATE. Вместе с тем, при этом следует проявлять осторожность в случае использования таких команд, как PSO:Verify Certifcate, когда по криптографическим соображениям порядок является фиксированным.

9.4 Объекты данных открытых ключей

Объект данных открытого ключа представляет собой построенную с помощью BER TLV структуру, которая содержит идентификатор объекта и несколько зависящих от контекста объектов данных, помещенных в шаблон открытого ключа 0x7F49 владельца карточки.

- Идентификатор объекта является специфическим для конкретного приложения и относится не только к формату открытого ключа (т. е. специфические для контекста объекта данных), но также к его применению.
- Специфические для контекста объекты данных определяются идентификатором объекта и содержат значение открытого ключа и параметры домена.

Описание формата объектов данных открытых ключей, используемых в этой спецификации, приводится ниже.

9.4.1 Кодирование объекта данных

Неподписанное целое число ПРЕОБРАЗОВЫВАЕТСЯ в восьмибитную строку с использованием двоичного представления целого числа в формате с обратным порядком байтов. ИСПОЛЬЗУЕТСЯ минимальное количество октетов, т. е. начальные октеты со значением 0x00 НЕ ДОЛЖНЫ использоваться.

Для кодирования точек эллиптических кривых ИСПОЛЬЗУЕТСЯ несжатое кодирование в соответствии с [TR-03111].

9.4.2 Открытые ключи RSA

Объекты данных, содержащиеся в открытом ключе RSA, показаны в таблице 9. Порядок объектов данных является фиксированным.

Таблица 9. Открытый ключ RSA

Объект данных	Обозначение	Тег	Тип	Сертификат CV
Идентификатор объекта		0x06	Идентификатор объекта	m
Сложный модуль	*n*	0x81	Неподписанное целое число	m
Открытая экспонента	*e*	0xl82	Неподписанное целое число	m

9.4.3 Открытые ключи Диффи-Хеллмана

Объекты данных, содержащиеся в открытом ключе, показаны в таблице 10. Порядок объектов данных является фиксированным.

Таблица 10. Объекты данных для открытых ключей DH

Объект данных	Обозначение	Тег	Тип
Идентификатор объекта		0x06	Идентификатор объекта
Простой модуль	p	0x81	Неподписанное целое число
Порядок подгруппы	q	0x82	Неподписанное целое число
Генератор	g	0x83	Неподписанное целое число
Открытое значение	y	0x84	Неподписанное целое число

Примечание. Кодирование компонентов ключа в виде неподписанного целого числа подразумевает, что каждый из них кодируется по наименьшему возможному количеству байтов, т. е. без предшествующих байтов, установленных на 0x00. В частности, открытый ключ DH может кодироваться по количеству байтов меньшему, чем количество байтов простого модуля.

9.4.4 Открытые ключи на основе эллиптической кривой

Объекты данных в открытом ключе на основе EC приводятся в таблице 11. Порядок объектов данных является фиксированным, УСЛОВНО ОБЯЗАТЕЛЬНЫЕ параметры домена ДОЛЖНЫ либо все присутствовать, за исключением сомножителя, либо все отсутствовать, как указано ниже.

Таблица 11. Объекты данных для открытых ключей ECDH

Объект данных	Обозначение	Тег	Тип
Идентификатор объекта		0x06	Идентификатор объекта
Простой модуль	p	0x81	Неподписанное целое число
Первый коэффициент	a	0x82	Неподписанное целое число
Второй коэффициент	b	0x83	Неподписанное целое число
Базовая точка	G	0x84	Точка эллиптической кривой
Порядок базовой точки	r	0x85	Неподписанное целое число
Открытая точка	Y	0x86	Точка эллиптической кривой
Сомножитель	f	0x87	Неподписанное целое число

9.4.5 Эфемерные открытые ключи

Для эфемерных открытых ключей формат и параметры домена уже известны. Поэтому для передачи эфемерного открытого ключа в специфическом для контекста объекте данных используется только простое значение открытого ключа, т. е. открытое значение y для открытых ключей Диффи-Хеллмана и открытая точка Y для открытых ключей эллиптической кривой.

Примечание. Валидация эфемерных открытых ключей РЕКОМЕНДУЕТСЯ. Для DH алгоритм валидации требует, чтобы у чипа электронного МСПД была более детальная информация о параметрах домена (т. е. порядок используемой подгруппы), чем это обычно предоставляется PKCS#3.

9.5 Параметры домена

За исключением параметров домена, содержащихся в поле "Информация PACE", все параметры домена предоставляются как идентификаторы алгоритма (см. раздел 9.1).

В рамках информации PACE обращение к идентификаторам стандартизированных параметров домена, приведенных в таблице 12, ОСУЩЕСТВЛЯЕТСЯ напрямую. В явных параметрах домена, содержащихся в поле "Информация параметров домена PACE", НЕ ДОЛЖНЫ использоваться ID, зарезервированные для стандартизированных параметров домена.

9.5.1 Стандартизированные параметры домена

СЛЕДУЕТ использовать идентификаторы (ID) стандартизированных параметров домена. В явных параметрах домена НЕ ДОЛЖНЫ использоваться ID, зарезервированные для стандартизированных параметров домена.

Для ссылки на стандартизированные параметры домена в поле "Идентификатор алгоритма" СЛЕДУЕТ использовать указанный ниже идентификатор объекта (см. раздел 9.1):

```
standardizedDomainParameters OBJECT IDENTIFIER ::= {
  bsi-de algorithms(1) 2
}
```

В рамках идентификатора алгоритма этот идентификатор объекта ОБЕСПЕЧИВАЕТ ссылку на ID стандартизированного параметра домена, указанного в таблице, в виде целого числа, содержащегося в качестве параметра в поле "Идентификатор алгоритма".

Таблица 12. Стандартизированные параметры домена

ID	*Название*	*Размер (биты)*	*Тип*	*Ссылка*
0	1024-битная группа MODP с 160-битовой подгруппой с порядком, равным простому числу	1024/160	GFP	[RFC 5114]
1	2048-битная группа MODP с 224-битовой подгруппой с порядком, равным простому числу	2048/224	GFP	[RFC 5114]
2	2048-битная группа MODP с 256-битовой подгруппой с порядком, равным простому числу	2048/256	GFP	[RFC 5114]
3–7	RFU			
8	NIST P-192 (secp192r1)	192	ECP	[RFC 5114], [FIPS 186-4]
9	BrainpoolP192r1	192	ECP	[RFC 5639]
10	NIST P-224 (secp224r1) *	224	ECP	[RFC 5114], [FIPS 186-4]
11	BrainpoolP224r1	224	ECP	[RFC 5639]
12	NIST P-256 (secp256r1)	256	ECP	[RFC 5114], [FIPS 186-4]
13	BrainpoolP256r1	256	ECP	[RFC 5639]
14	BrainpoolP320r1	320	ECP	[RFC 5639]
15	NIST P-384 (secp384r1)	384	ECP	[RFC 5114], [FIPS 186-4]
16	BrainpoolP384r1	384	ECP	[RFC 5639]
17	BrainpoolP512r1	512	ECP	[RFC 5639]
18	NIST P-521 (secp521r1)	521	ECP	[RFC 5114], [FIPS 186-4]
19–31	RFU			

* Данная кривая не может использоваться с интегрированным отображением.

9.5.2 Явные параметры домена

Для ссылки на явные параметры домена в поле "Идентификатор алгоритма" ИСПОЛЬЗУЕТСЯ идентификатор объекта "Открытый номер DH" или "Открытый ключ EC" соответственно для DH или ECDH (см. раздел 9.1):

```
dhpublicnumber OBJECT IDENTIFIER ::= {
  iso(1) member-body(2) us(840) ansi-x942(10046) number-type(2) 1
}
```

```
ecPublicKey OBJECT IDENTIFIER ::= {
  iso(1) member-body(2) us(840) ansi-x962(10045) keyType(2) 1
}
```

В случае эллиптических кривых параметры домена ДОЛЖНЫ быть четко описаны в структуре поля "Параметры EC", содержащегося в качестве "Параметров" в "Идентификаторе алгоритма", т. е. именованные кривые и неявные параметры домена НЕ ДОЛЖНЫ использоваться.

9.6 Алгоритмы согласования ключей

Данная спецификация поддерживает алгоритм согласования ключей Диффи-Хеллмана и алгоритм согласования ключей Диффи-Хеллмана с использованием эллиптической кривой, как это указано в обобщенном виде в следующей таблице.

Таблица 7. Алгоритмы согласования ключей

Алгоритм/формат	***DH***	***ECDH***
Алгоритм согласования ключей	[PKCS#3]	ECKA [TR-03111]
Формат открытого ключа X.509	[X9.42]	[TR-03111]
Формат открытого ключа TLV	TLV, см. раздел 9.4.3	TLV, см. раздел 9.4.4
Валидация эфемерного открытого ключа	[RFC 2631]	[TR-03111]

9.7 Механизм выработки ключа

9.7.1 Функция выработки ключа

Функция выработки ключа **KDF**(K,c) определяется следующим образом:

Вводные данные. Требуются следующие вводные данные:

- совместно используемый секретный ключ K (ОБЯЗАТЕЛЬНЫЙ);
- 32-битный счетчик целых чисел с обратным порядком байтов *c* (ОБЯЗАТЕЛЬНЫЙ).

Выходные данные. Данные ключа в виде 8-битной строки.

Действия. Выполняются следующие действия:

- данные ключа = **H**(K || c);
- выходные данные ключа в виде 8-битной строки.

Функция выработки ключей **KDF**(K,c) требует применения соответствующей хэш-функции, обозначаемой **H**(), т. е. длина в битах хэш-функции ПРЕВЫШАЕТ или РАВНА длине выработанного ключа в битах. Хэш-значение ИНТЕРПРЕТИРУЕТСЯ как результат с обратным порядком байтов.

Примечание. Совместно используемый секретный ключ K определяется в виде 8-битной строки. Если совместно применяемый секретный параметр генерируется с помощью ECKA [TR-03111], ИСПОЛЬЗУЕТСЯ координата x генерируемой точки.

9.7.1.1 3DES

Для выработки 128-битных (112-битных, исключая биты четности) ключей 3DES [FIPS 46-3] ИСПОЛЬЗУЕТСЯ хэш-функция SHA-1 [FIPS 180-4] и ДОЛЖНЫ быть выполнены следующие дополнительные этапы:

- Использовать октеты 1–8 данных ключа для формирования данных ключа A и октеты 9–16 данных ключа для формирования данных ключа B; при этом дополнительные октеты не используются.
- Скорректировать биты четности данных ключа A и данных ключа B для формирования правильных DES-ключей (ФАКУЛЬТАТИВНО).

9.7.1.2 AES

Для выработки 128-битных AES-ключей [FIPS 197] ИСПОЛЬЗУЕТСЯ хэш-функция SHA-1 [FIPS 180-4], при этом ДОЛЖЕН быть выполнен следующий дополнительный этап:

- использовать октеты 1–16 данных ключа; дополнительные октеты не используются.

Для выработки 192- и 256-битных AES-ключей [FIPS 197] ИСПОЛЬЗУЕТСЯ SHA-256 [FIPS 180-4]. Для 192-битных AES-ключей ДОЛЖЕН быть выполнен следующий дополнительный этап:

- использовать октеты 1–24 данных ключа; дополнительные октеты не используются.

9.7.2 Базовые ключи доступа к документу

Для установления базовых ключей доступа к документу K_{Enc} = **KDF**(K,1) и K_{MAC} = **KDF**(K,2) вычисляются два 3DES-ключа из начального числа ключа (K).

9.7.3 PACE

Пусть $\mathbf{KDF}_{\pi}(\pi)$ = **KDF**(f(π),3) является функцией выработки ключей для получения ключей кодирования из пароля π. Кодировка паролей, т. е. K= f(π) изложена в таблице 14.

Таблица 8. Кодировка паролей

Пароль	**Кодировка**
MCЗ	SHA-1 (номер документа || дата рождения || дата истечения срока действия)
CAN	Закодированная по стандарту [ИСО/МЭК 8859-1] строка знаков

Примечание. Номер документа, подлежащий использованию в качестве входных данных, всегда представляет собой полный номер документа. В случае документов TD-1 с номерами документов, превышающими 9 знаков, номер документа необходимо конкатенировать из поля номера документа и поля факультативных данных МСЗ, за исключением знака заполнителя. Также см. примечание j) в разделе 4.2.2 части 5 документа Doc 9303.

9.7.4 Ключи безопасного обмена сообщениями

Ключи для кодировки и аутентификации вырабатываются с помощью соответственно $\mathbf{KDF}_{Enc}(K) = \mathbf{KDF}(K,1)$ и $\mathbf{KDF}_{MAC}(K) = \mathbf{KDF}(K,2)$ из совместно используемого секретного ключа K.

9.8 Безопасный обмен сообщениями

9.8.1 Инициирование сеанса

Сеанс начинается с момента установления безопасного обмена сообщениями. В ходе сеанса ключи безопасного обмена сообщениями (т. е. установленные посредством базового контроля доступа, PACE или аутентификации чипа) могут быть изменены.

Безопасный обмен сообщениями основан либо на 3DES [FIPS 46-3], либо на AES [FIPS 197] в режиме "зашифровать–затем–аутентифицировать", т. е. данные заполняются, зашифровываются и затем сформатированные зашифрованные данные используются в качестве вводных данных для аутентификационных вычислений. Сеансовые ключи ВЫЧИСЛЯЮТСЯ с помощью функции выработки ключей, описанной в разделе 9.7.1.

Примечание. Заполнение всегда осуществляется с использованием безопасного уровня передачи сообщений, поэтому лежащему в основе коду аутентификации сообщения нет необходимости выполнять какое-либо внутреннее заполнение.

9.8.2 Счетчик посылаемых блоков

В качестве счетчика посылаемых блоков (SSC) ИСПОЛЬЗУЕТСЯ неподписанное целое число. Размер бита SSC РАВНЯЕТСЯ размеру блока блочного шифра, используемого для безопасного обмена сообщениями, т. е. 64 бит для 3DES и 128 бит для AES.

Значение SSC УВЕЛИЧИВАЕТСЯ каждый раз перед генерированием APDU команды или ответа, т. е. если начальное значение составляет x, то в первой команде значение SSC составляет x+1. Значение SSC в первом ответе составляет x+2.

Если безопасный обмен сообщениями возобновляется, SSC используется следующим образом:

- Команды, используемые для согласования ключей, защищены старыми сеансовыми ключами и старым SSC. Это применяется, в частности, к ответу на последнюю команду, использованную для согласования сеансовых ключей.
- Счетчик посылаемых блоков устанавливается на свое новое начальное значение (см. раздел 9.8.6.3 для 3DES и раздел 9.8.7.3 для AES).
- Для защиты последующих команд/ответов используются новые сеансовые ключи и новый SSC.

9.8.3 Завершение сеанса

Чип электронного МСПД ДОЛЖЕН прервать безопасный обмен сообщениями тогда и только тогда, когда возникает ошибка в безопасном обмене сообщениями или получен открытый APDU.

Если безопасный обмен сообщениями прерывается, чип электронного МСПД УДАЛЯЕТ хранящиеся сеансовые ключи и сбрасывает назначенные права доступа для этого терминала.

Примечание. Когда сеанс завершается, чип электронного МСПД МОЖЕТ выбрать мастер-файл как подразумеваемый.

9.8.4 Структура сообщений SM APDU

Объекты данных SM (см. [ИСО/МЭК 7816-4]) ДОЛЖНЫ использоваться в следующем порядке:

- APDU команды: [DO‘85’ or DO‘87’] [DO‘97’] DO‘8E’.
- APDU ответа: [DO‘85’ or DO‘87’] [DO‘99’] DO‘8E’.

В случае четного INS ИСПОЛЬЗУЕТСЯ DO‘87’, а в случае нечетного INS ИСПОЛЬЗУЕТСЯ DO‘85’.

Все объекты данных SM ДОЛЖНЫ быть закодированы в BER TLV, как указано в [ИСО/МЭК 7816-4]. Заголовок команды ДОЛЖЕН быть включен в процесс вычисления MAC, поэтому ДОЛЖЕН использоваться байт класса CLA = 0X0C.

Фактическое значение Lc будет изменено на Lc’ после применения безопасного обмена сообщениями. При необходимости соответствующий объект данных факультативно можно включать в данные APDU для передачи исходного значения Lc.

На рис. 5 показана схема преобразования незащищенного APDU команды в защищенный APDU команды в случае наличия *данных* и *Le*. Если *данные* отсутствуют, построение DO ‘87’ не осуществляется. Если *Le* отсутствует, построение DO ‘97’ не осуществляется. Чтобы не возникало неоднозначности, РЕКОМЕНДУЕТСЯ не использовать пустое поле значения объекта данных Le (см. также раздел 10.4 стандарта [ИСО/МЭК 7816-4]).

На рис. 6 показано преобразование незащищенного APDU ответа в защищенный APDU ответа в случае наличия *данных.* Если *данные* отсутствуют, построение DO ‘87’ не осуществляется.

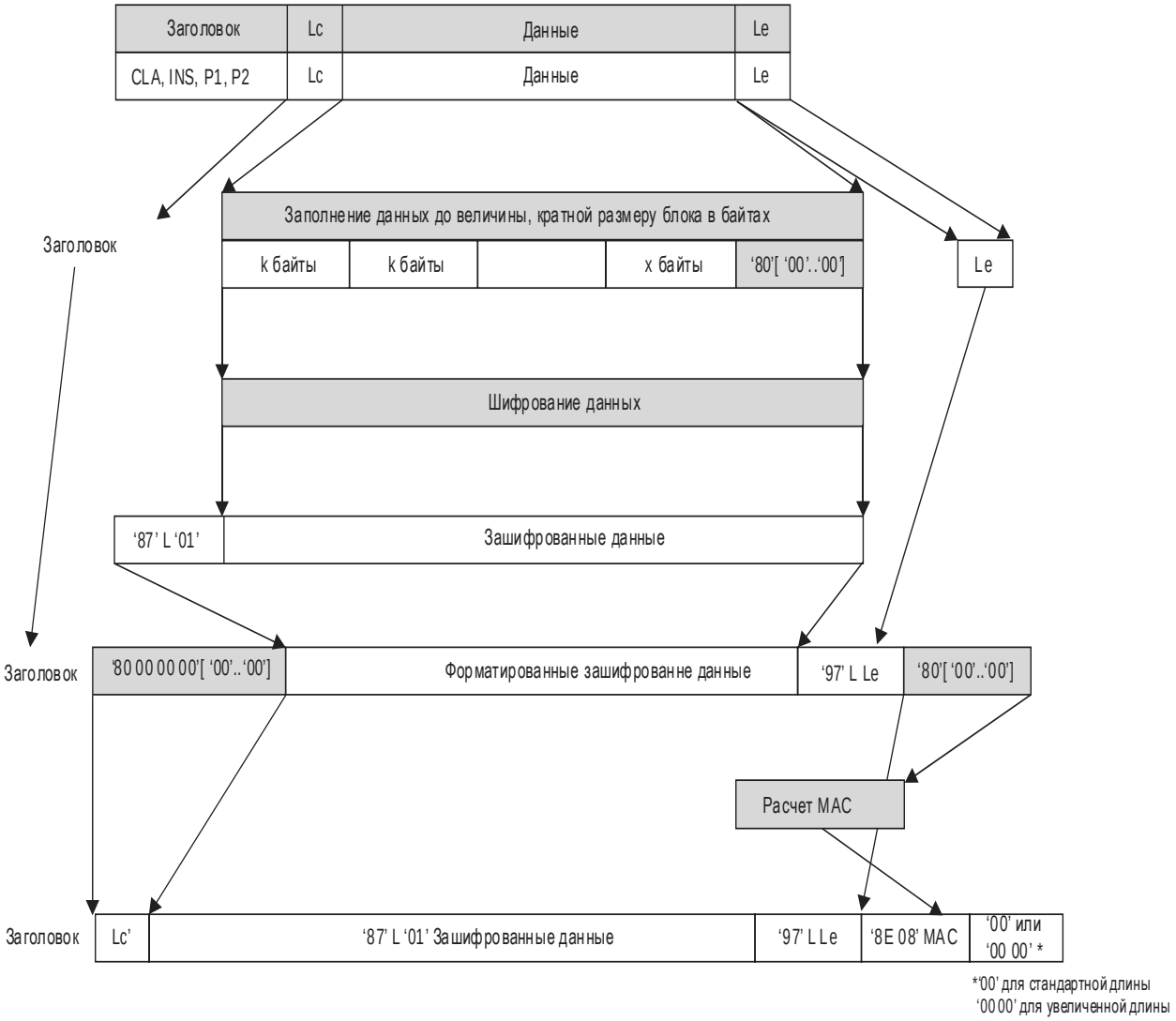

Рис. 5. Вычисление APDU SM команды для четного байта INS

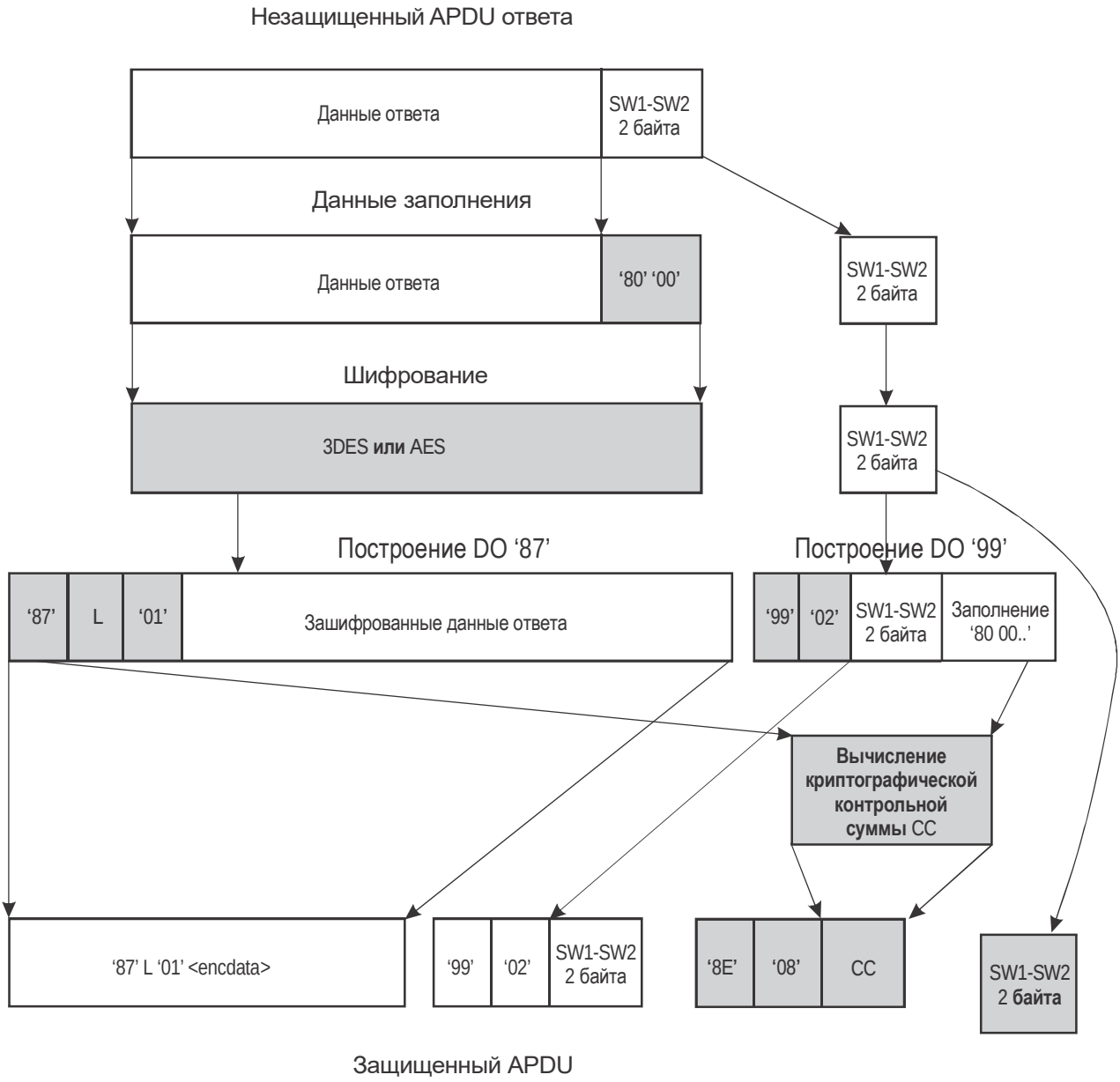

Рис. 6. Вычисление APDU SM ответа для четного байта INS

9.8.5 Ошибки SM

Прерывание защищенного канала приложения электронного МСПД имеет место в следующих случаях:

- бесконтактная ИС обесточена или
- бесконтактная ИС опознает ошибку SM в процессе интерпретации команды. В этом случае байты состояния должны быть возвращены без SM.

Если безопасный обмен сообщениями прерывается, чип электронного МСПД УДАЛЯЕТ хранящиеся сеансовые ключи и СБРАСЫВАЕТ назначенные права доступа для этого терминала.

Примечание. МОГУТ быть другие обстоятельства, в которых ИС прерывает сеанс. Составление всеобъемлющего перечня таких обстоятельств не представляется практически осуществимым.

9.8.6 Режимы работы 3DES

9.8.6.1 Шифрование

Используется двухключевой 3DES в режиме CBC с нулевым вектором IV (т. е. 0x00 00 00 00 00 00 00 00) в соответствии со стандартом [ИСО/МЭК 11568-2]. Используется заполнение в соответствии с методом заполнения 2 стандарта [ИСО/МЭК 9797-1].

9.8.6.2 Аутентификация сообщений

Криптографические контрольные суммы вычисляются с использованием MAC алгоритма 3 стандарта [ИСО/МЭК 9797-1] с блочным шифром DES (нулевой вектор IV (8 байтов)) и метода заполнения 2 стандарта [ИСО/МЭК 9797-1]. Длина MAC ДОЛЖНА быть 8 байтов.

После успешной аутентификации датаграмма, подлежащая кодированию с помощью MAC, ДОЛЖНА быть добавлена к началу счетчиком посылаемых блоков.

9.8.6.3 Счетчик посылаемых блоков

Для осуществления безопасного обмена сообщениями в соответствии с BAC счетчик посылаемых блоков ИНИЦИАЛИЗИРУЕТСЯ путем конкатенации соответственно четырех наименее значимых байтов RND.IC и RND.IFD:

SSC = RND.IC (4 наименее значимых байта) || RND.IFD (4 наименее значимых байта).

Во всех других случаях исходное состояние SSC УСТАНАВЛИВАЕТСЯ на нули (т. е. 0x00 00 00 00 00 00 00 00).

9.8.7 Режимы работы AES

9.8.7.1 Шифрование

Для шифрования сообщений ИСПОЛЬЗУЕТСЯ AES [FIPS 197] в режиме CBC в соответствии со стандартом [ИСО/МЭК 10116] с применением ключа KS_{Enc} и вектора инициализации IV = **E**(KS_{Enc}, SSC).

9.8.7.2 Аутентификация сообщений

Для аутентификации сообщений ИСПОЛЬЗУЕТСЯ AES в режиме CMAC [SP 800-38B] с применением ключа KS_{MAC} с длиной MAC 8 байтов. Подлежащая аутентификации датаграмма ДОБАВЛЯЕТСЯ счетчиком посылаемых блоков.

9.8.7.3 Счетчик посылаемых блоков

Счетчик посылаемых блоков ИНИЦИАЛИЗИРУЕТСЯ с установкой на ноль (т. е. 0x00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00).

10. СПРАВОЧНЫЕ МАТЕРИАЛЫ (НОРМАТИВНЫЕ)

[X9.42] ANSI: X9.42, Public Key Cryptography for the Financial Services Industry: Agreement of Symmetric Keys Using Discrete Logarithm Cryptography, 1999

[ИСО/МЭК 7816-4] ИСО/МЭК 7816-4:2013. Карты идентификационные. Карты на интегральных схемах. Часть 4. Организация, защита и команды для обмена

[ИСО/МЭК 7816-8] ИСО/МЭК 7816-8:2019. Карты идентификационные. Карты на интегральных схемах. Часть 8. Команды и механизмы для операций по защите информации

[ИСО/МЭК 8859-1] ИСО/МЭК 8859-1:1998. Информационные технологии. 8-битовые однобайтовые наборы кодированных графических знаков. Часть 1. Латинский алфавит № 1

[ИСО/МЭК 9796-2] ИСО/МЭК 9796-2:2010. Информационные технологии. Методы обеспечения безопасности. Схемы цифровой подписи, обеспечивающие восстановление сообщений. Часть 2. Механизмы на основе целочисленной факторизации

[ИСО/МЭК 9797-1] ИСО/МЭК 9797-1:2011. Информационные технологии. Методы защиты. Коды аутентификации сообщений (MAC). Часть 1. Механизмы, использующие блочный шифр

[ИСО/МЭК 10116] ИСО/МЭК 10116:2017. Информационные технологии. Методы обеспечения безопасности. Режимы работы для n-битовых блочных шифров

[ИСО/МЭК 11568-2] ИСО/МЭК 11568-2:2012. Банковское дело. Менеджмент ключей (розничная торговля). Часть 2. Симметричные алгоритмы шифрования, управление их ключами и жизненный цикл

[ИСО/МЭК 11770-2] ИСО/МЭК 11770-2:2018. Информационные технологии. Методы защиты. Управление ключами защиты. Часть 2. Механизмы, использующие симметричные методы

[FIPS 46-3] NIST FIPS PUB 46-3. Стандарт кодирования данных (DES), 1999.

[FIPS 180-4] NIST FIPS PUB 180-4. Стандарт хэш-функций защиты , 2015.

[FIPS 186-4] NIST FIPS PUB 186-4. Стандарт на цифровую подпись (DSS), 2013.

[FIPS 197] NIST FIPS PUB 197. Спецификации усовершенствованного стандарта кодирования (AES), 2001.

[SP 800-38B] NIST Special Publication 800-38B, Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, 2005.

[RFC 2631] Rescorla, Eric: RFC 2631 Diffie-Hellman key agreement method, 1999.

[RFC 3447] Jonsson, Jakob and Kaliski, Burt: RFC 3447, Public-key cryptography standards (PKCS) #1: RSA cryptography specifications version 2.1, 2003

[RFC 5114] Lepinski, Matt; Kent, Stephen: RFC 5114 Additional Diffie-Hellman Groups for Use with IETF Standards, 2008.

[RFC 5280] D. Cooper, S. Santesson, S. Farrell, S. Boyen, R. Housley, W. Polk, RFC 5280 Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, 2008.

[RFC 5639] Lochter, Manfred; Merkle, Johannes: RFC 5639 Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation, 2010.

[TR-03110] BSI: Technical Guideline TR-03110: Advanced Security Mechanisms for Machine Readable Travel Documents

[TR-03111] BSI: Техническое руководство TR-03111: Криптография эллиптических кривых, версия 2.0, 2012.

[PKCS#1] RSA Laboratories, PKCS#1 v2.2: RSA Cryptography Standard, 2012

[PKCS#3] RSA Laboratories, PKCS#3: Diffie-Hellman key-agreement standard, 1993.

[Keesing2009] J. Bender, D. Kügler: Introducing the PACE solution, in: Keesing Journal of Documents & Identity, Issue 30, Keesing, 2009.

[BFK2009] J. Bender, M. Fischlin, D. Kügler: Security Analysis of the PACE Key-Agreement Protocol, in: Proceedings ISC 2009, LNCS volume 5735, Springer, 2009.

[BCIMRT2010] Brier, Eric; Coron, Jean-Sébastien; Icart, Thomas; Madore, David; Randriam, Hugues; and Tibouch, Mehdi, Efficient Indifferentiable Hashing into Ordinary Elliptic Curves, Advances in Cryptology – CRYPTO 2010, Springer-Verlag, 2010.

— — — — — — — —

Добавление A к части 11

ЭНТРОПИЯ КЛЮЧЕЙ ДОСТУПА, ПОЛУЧЕННЫХ ИЗ МСЗ (ИНФОРМАЦИОННОЕ)

Благодаря своей простоте базовый контроль доступа оказался очень успешным протоколом и применяется почти в каждом электронном МСПД.

Защита, обеспечиваемая базовым контролем доступа, ограничена структурой протокола. Базовые ключи доступа к документу (K_{Enc} и K_{MAC}) генерируются из напечатанных данных с очень ограниченной степенью случайности. Данными, используемыми для генерирования ключей, являются номер документа, дата рождения и дата истечения срока действия. Как следствие, полученные в результате ключи обладают относительно низкой энтропией и являются криптографически слабыми. Фактическая энтропия в основном зависит от типа номера документа. Для проездного документа со сроком действия 10 лет максимальная криптостойкость таких ключей приблизительно составляет:

- 56 бит для числового номера документа ($365^2 * 10^{12}$ возможностей);
- 73 бит для буквенно-числового номера документа ($365^2*36^9*10^3$ возможностей).

Особенно во втором случае такая оценка предусматривает, чтобы номер документа выбирался случайным и единообразным способом, что обычно не соответствует действительности. В зависимости от осведомленности злоумышленника фактическая энтропия базового ключа доступа к документу может быть ниже, например, если злоумышленник знает все используемые номера документов или способен соотнести номера документов с датами истечения срока их действия.

Прямого способа усиления базового контроля доступа не существует, поскольку его ограничения заложены в структуре протокола, основанного на симметричной ("секретный ключ") криптографии. Криптостойкий механизм контроля доступа должен (дополнительно) использовать асимметричную ("открытый ключ") криптографию.

Для преодоления этой проблемы был разработан механизм установления соединения аутентификацией паролем (PACE). В нем применяется асимметричная криптография для установления сеансовых ключей, чья стойкость не зависит от энтропии используемого пароля. Если PACE осуществляется с использованием криптографических методов на основе эллиптических кривых с 256-битными кривыми и AES-128 (обычный выбор), энтропия сеансовых ключей составляет 128 бит.

Необходимо различать два типа атаки:

- Скимминг. Это атака в режиме онлайн, т. е. злоумышленник пытается получить доступ к бесконтактной ИС в реальном режиме времени, например путем угадывания пароля. Если протокол, используемый для защиты бесконтактной ИС, не является криптографически слабым, вероятность успеха злоумышленника определяется временем, в течение которого он имеет доступ к ИС, продолжительностью единственной попытки угадать пароль и энтропией пароля.

- Перехват. Это атака в офлайновом режиме, т. е. злоумышленник пытается расшифровать перехваченный обмен сообщениями без доступа к бесконтактной ИС. Если протокол, используемый для установления сеансовых ключей, не является криптографически слабым, вероятность успеха злоумышленника определяется надежностью сеансовых ключей и мощностью вычислительных средств, которыми располагает злоумышленник.

Дополнительная информация в отношении общих аспектов энтропии сеансовых ключей и сравнения механизмов BAC и PACE приводится в документе [Keesing2009], а в отношении криптографического анализа PACE – в документе [BFK2009].

— — — — — — — —

Добавление B к части 11

КОДИРОВАНИЕ ТОЧЕК ДЛЯ ИНТЕГРИРОВАННЫХ НА ОСНОВЕ ECDH ОТОБРАЖЕНИЙ (ИНФОРМАЦИОННОЕ)

B.1 ОПИСАНИЕ ВЫСОКОГО УРОВНЯ МЕТОДА КОДИРОВАНИЯ ТОЧЕК

В алгоритме используются в качестве вводных данных параметры кривой (a, b, p, f), где (a, b) представляют собой коэффициенты кривой, p является характеристикой простого поля, по которому определяется кривая:

$$E : y^2 \equiv x^3 + ax + b \ (mod\ p).$$

Порядок E всегда имеет форму fq для некоторых простых q, а f называется сомножителем. Для PACE v2 требуется генерирование точки, принадлежащей подгруппе q кривой E, которую мы обозначаем $E[q]$. Механизм кодирования точки использует также в качестве вводных данных число t, причем:

$$0 < t < p,$$

и возвращает через фиксированный промежуток времени точку, принадлежащую $E[q]$. Как описано в [BCIMRT2010], кодирование точек осуществляется двумя способами, в зависимости от системы координат, предпочитаемой в конкретном варианте реализации:

- в первом варианте применения, описанном в разделе B.2, результатом является точка эллиптической кривой в аффинных координатах (x, y);
- в альтернативном варианте применения, представленном в разделе B.3, результатом является та же точка в координатах Якоби (X, Y, Z).

Независимо от принятого варианта генерируемая точка является идентичной в том смысле, что:

$$x = XZ^{-2} \ mod\ p \text{ и } y = YZ^{-3} \ mod\ p,$$

и поэтому при выполнении последующего этапа PACE v2 (этап обмена ключами Диффи-Хеллмана, выработанными на основе эллиптической кривой) можно воспользоваться возможностью применить вариант, который наиболее подходит для интерфейса криптографической системы API, выполняющего операции с использованием эллиптической кривой.

Как указывается далее, для шифрования точек в аффинных координатах требуется примерно два модулярных возведения в степень по модулю p, в то время как при шифровании точек в координатах Якоби требуется лишь одно.

Следует учесть, что для двух имеющихся вариантов расчета кодирование точек однозначно требует, чтобы $p \equiv 3 \ mod\ 4$.

В.2 ВАРИАНТ РАСЧЕТА В АФФИННЫХ КООРДИНАТАХ

Алгоритм осуществляется следующим образом:

Вводные данные. Параметры кривой (a, b, p, f) и t, причем $0 < t < p$.

Выходные данные. Точка (x, y) в подгруппе с простым порядком $E[q]$ из E.

1. Вычисление $\alpha = -t^2 \bmod p$.
2. Вычисление $X_2 = -ba^{-1}(1+(\alpha+\alpha^2)^{-1}) \bmod p$.
3. Вычисление $X_3 = \alpha X_2 \bmod p$.
4. Вычисление $h_2 = (X_2)^3 + a X_2 + b \bmod p$.
5. Вычисление $h_3 = (X_3)^3 + a X_3 + b \bmod p$.
6. Вычисление $U = t^3 h_2 \bmod p$.
7. Вычисление $A = (h_2)^{p-1-(p+1)/4} \bmod p$.
8. Если $A^2 h_2 = 1 \bmod p$, определяется $(x, y) = (X_2, A h_2 \bmod p)$.
9. В иных случаях определяется $(x, y) = (X_3, A U \bmod p)$.
10. Выходные данные $(x, y) = [f] (x, y)$.

Замечания по выполнению действий

Если не считать модулярные операции умножения и сложения, время выполнения вышеупомянутых действий в основном зависит от двух модулярных возведений в степень:

- этап 2 можно переписать в виде

$$X_2 = -ba^{-1}(1+(\alpha+\alpha^2)^{-1}) = -b(1+\alpha+\alpha^2)\,(a(\alpha+\alpha^2))^{p-2} \bmod p,$$

  что по существу равняется модулярному возведению в степень $p-2$;

- этап 7 представляет собой модулярное возведение в степень $p-1-(p+1)/4$.

Примечание. Этап 10 предусматривает скалярное умножение на сомножитель f. *Для многих кривых сомножитель равняется 1, поэтому данное скалярное умножение можно избежать.*

В.3 ВАРИАНТ РАСЧЕТА В КООРДИНАТАХ ЯКОБИ

Алгоритм осуществляется следующим образом:

Вводные данные. Параметры кривой (a, b, p, f) и t, причем $0 < t < p$.

Выходные данные. Точка (X, Y, Z) в подгруппе с простым порядком $E[q]$ из E.

1. Вычисление $\alpha = -t^2 \bmod p$.
2. Вычисление $Z = a(\alpha+\alpha^2) \bmod p$.
3. Вычисление $X_2 = -bZ\,(1+\alpha+\alpha^2) \bmod p$.
4. Вычисление $X_3 = \alpha X_2 \bmod p$.
5. Вычисление $h_2 = (X_2)^3 + a X_2 Z^4 + b Z^6 \bmod p$.
6. Вычисление $h_3 = (X_3)^3 + a X_3 Z^4 + b Z^6 \bmod p$.
7. Вычисление $U = -\alpha\, t\, h_2 \bmod p$.

8. Вычисление $A = (h_2)^{p-1-(p+1)/4} \bmod p$.
9. Если $A^2 h_2 = 1 \bmod p$, определяется $(X, Y, Z) = (X_2, A\, h_2 \bmod p, Z)$.
10. В иных случаях определяется $(X, Y, Z) = (X_3, A\, U \bmod p, Z)$.
11. Выходные данные $(X, Y, Z) = [f](X, Y, Z)$.

Замечания по выполнению действий

Если не считать модулярные операции умножения и сложения, время выполнения вышеупомянутых действий в основном зависит от единственного модулярного возведения в степень на этапе 7. Поэтому ожидается, что осуществление данного алгоритма займет примерно вдвое меньше времени, чем расчеты в аффинных координатах.

Примечание. Скалярное умножение на этапе 10 можно полностью избежать, когда сомножитель f *равен 1.*

— — — — — — — —

Добавление C к части 11

СЕМАНТИКА КОМАНДЫ CHALLENGE (ЗАПРОС) (ИНФОРМАЦИОННОЕ)

Рассмотрим основанный на подписи запросно-ответный протокол между чипом электронного МСПД (ИС) и терминалом (IFD), когда чип электронного МСПД хочет доказать знание своего закрытого ключа SK_{IC}:

- терминал посылает выбранный случайным образом запрос *c* на чип электронного МСПД;
- чип электронного МСПД отвечает подписью *s*=**Sign**(SK_{IC},c).

Хотя это очень простой и эффективный протокол, чип электронного МСПД фактически подписывает сообщение *c,* не зная семантики этого сообщения. Поскольку подписи обеспечивают доказательство аутентичности, которое можно передавать, любая третья сторона может (в принципе) быть убеждена в том, что чип электронного МСПД действительно подписал это сообщение.

Хотя сообщение *c* должно представлять собой случайную строку битов, терминал может также сгенерировать эту строку битов непредсказуемым, но (открыто) верифицируемым способом, например, пусть SK_{IFD} является закрытым ключом терминала, а

c = **Sign**$(SK_{IFD},ID_{IC}||Date||Time||Location)$

является командой запроса, генерируемой с использованием схемы подписи с восстановлением сообщения. Подпись гарантирует, что терминал действительно сгенерировал этот запрос. Благодаря возможности передачи подписи терминала, любая третья сторона, доверяющая терминалу и знающая соответствующий открытый ключ PK_{IFD}, может проверить, правильно ли был создан запрос, путем верификации этой подписи. Более того, благодаря возможности передачи подписи чипа электронного МСПД по запросу третья сторона может прийти к заключению, что это утверждение отвечает истине: чип электронного МСПД действительно оказался на определенную дату и в определенное время в определенном месте.

Позитивной стороной является то, что государства могут использовать семантику запросов для своих внутренних целей, например для доказательства того, что определенное лицо действительно эмигрировало. Негативной стороной является то, что такими доказательствами можно злоупотреблять для отслеживания лиц. В частности, поскольку активная аутентификация не запрещена для санкционированных терминалов, злоупотребление возможно. Наихудшим сценарием был бы вариант, когда чипы электронного МСПД обеспечивали бы активную аутентификацию без базового контроля доступа. В этом случае может быть установлена очень мощная система слежения путем размещения безопасных модулей аппаратного оборудования в находящихся на виду местах. Полученные в результате файлы регистрации не могут быть подделаны благодаря подписям. В определенной системе базовый контроль доступа уменьшает эту проблему, так как требуется взаимодействие с владельцем документа. Тем не менее проблема остается, но ограничена местами, где проездные документы владельца в любом случае считываются, например, авиакомпаниями или гостиницами.

Могут возникнуть возражения относительно того, что, особенно в рамках бесконтактного сценария, команды запроса могут быть перехвачены и повторно использованы в другой день, в другое время или в другом месте, и таким образом доказательство станет по крайней мере ненадежным. Хотя перехват команд запроса технически возможен, этот аргумент, тем не менее, не является убедительным. По предположению, терминалу доверяют в том, что он правильно составляет запросы, и можно предположить, что он проверил идентификационные данные чипа электронного МСПД, прежде чем начать процесс активной аутентификации. Таким образом, перехваченный запрос будет содержать идентификационные данные, отличные от идентификационных данных подтверждающего субъекта, который подписывает запрос.

— — — — — — — —

Добавление D к части 11

ПРИМЕР С РЕШЕНИЯМИ: БАЗОВЫЙ КОНТРОЛЬ ДОСТУПА (ИНФОРМАЦИОННОЕ)

D.1 ВЫЧИСЛЕНИЕ КЛЮЧЕЙ ИЗ НАЧАЛЬНОГО ЧИСЛА КЛЮЧА (K_{SEED})

В настоящем разделе приводится пример выработки 3DES-ключей из значения начального числа ключа K_{seed}. Данная процедура будет использоваться в примерах базового контроля доступа в качестве "стандартной подпрограммы".

Ввод:
K_{seed} = ‘239AB9CB282DAF66231DC5A4DF6BFBAE’

Вычисление ключа шифрования (c = ‘00000001’):

1. Конкатенация K_{seed} и c:
   D = ‘239AB9CB282DAF66231DC5A4DF6BFBAE00000001’

2. Вычисление SHA-1 хэш D:
   H_{SHA-1}(D) = ‘AB94FCEDF2664EDFB9B291F85D7F77F27F2F4A9D’

3. Формирование DES-ключей K_a и K_b, предназначенных для использования в качестве первого и второго ключа для 3DES (т. е 3DES-ключ является конкатенацией K_a и K_b):
   $K_{a'}$ = ‘AB94FCEDF2664EDF’
   $K_{b'}$ = ‘B9B291F85D7F77F2’

4. Корректировка битов четности:
   K_a = ‘AB94FDECF2674FDF’
   K_b = ‘B9B391F85D7F76F2’

Вычисление ключа расчета MAC (c = ‘00000002’):

1. Конкатенация K_{seed} и c:
   D = ‘239AB9CB282DAF66231DC5A4DF6BFBAE00000002’

2. Вычисление SHA-1 хэш для D:
   H_{SHA-1}(D) = ‘7862D9ECE03C1BCD4D77089DCF131442814EA70A’

3. Формирование ключей $K_{a'}$ и $K_{b'}$:
   $K_{a'}$ = ‘7862D9ECE03C1BCD’
   $K_{b'}$ = ‘4D77089DCF131442’

4. Корректировка битов четности:
   K_a = ‘7962D9ECE03D1ACD’
   K_b = ‘4C76089DCE131543’

D.2 ПОЛУЧЕНИЕ БАЗОВЫХ КЛЮЧЕЙ ДОСТУПА К ДОКУМЕНТУ (K_{ENC} И K_{MAC})

В настоящем разделе приводятся примеры базовых ключей доступа, выработанных из МСЗ.

МСЗ ПД2, номер документа состоит более чем из 9 знаков

1. Считывание МСЗ

   МСЗ = `I<UTOSTEVENSON<<PETER<JOHN<<<<<<<<<<`
   `D23145890<UTO3407127M95071227349<<<8`

2. Построение ‘зоны МСЗ_информации’ из МСЗ

Номер документа	= D23145890734	контрольная цифра = 9
Дата рождения	= 340712	контрольная цифра = 7
Дата истечения срока действия	= 950712	контрольная цифра = 2
МСЗ_информация	= D23145890734934071279507122	

Продолжить этап 3.

МСЗ ПД2, номер документа состоит из 9 знаков

1. Считывание МСЗ:

   МСЗ = `I<UTOERIKSSON<<ANNA<MARIA<<<<<<<<<<`
   `L898902C<3UTO6908061F9406236<<<<<<<8`

2. Построение ‘МСЗ_информации’ из МСЗ:

Номер документа	= L898902C<	контрольная цифра = 3
Дата рождения	= 690806	контрольная цифра = 1
Дата истечения срока действия	= 940623	контрольная цифра = 6
МСЗ_информация	= L898902C<369080619406236	

Продолжить этап 3.

МСЗ ПД1, номер документа состоит более чем из 9 знаков

1. Считывание МСЗ

   МСЗ = `I<UTOD23145890<7349<<<<<<<<<<`
   `3407127M9507122UTO<<<<<<<<<<2`
   `STEVENSON<<PETER<JOHN<<<<<<<<<`

2. Построение ‘зоны МСЗ_информации’ из МСЗ

Номер документа	= D23145890734	контрольная цифра = 9
Дата рождения	= 340712	контрольная цифра = 7
Дата истечения срока действия	= 950712	контрольная цифра = 2
МСЗ информация	= D23145890734934071279507122	

Продолжить этап 3.

МСЗ ПД1, номер документа состоит из 9 знаков

1. Считывание МСЗ

   МСЗ = `I<UTOL898902C<3<<<<<<<<<<<<<<`
   `6908061F9406236UTO<<<<<<<<<<1`
   `ERIKSSON<<ANNA<MARIA<<<<<<<<<<`

2. Построение ‘MC3_информации’ из MC3

Номер документа	= L898902C<	контрольная цифра = 3
Дата рождения	= 690806	контрольная цифра = 1
Дата истечения срока действия	= 940623	контрольная цифра = 6
MC3_информация	= L898902C<369080619406236	

3. Вычисление SHA-1 хэш ‘MC3_информации’:
H_{SHA-1}(MC3_информация) = ‘239AB9CB282DAF66231DC5A4DF6BFBAEDF477565’

4. Использование наиболее значимых 16 байтов для формирования K_{seed}:
K_{seed} = ‘239AB9CB282DAF66231DC5A4DF6BFBAE’

5. Вычисление базовых ключей доступа (K_{Enc} и K_{MAC}) согласно разделу 9.7.1/добавлению D.1:
K_{Enc} = ‘AB94FDECF2674FDFB9B391F85D7F76F2’
K_{MAC} = ‘7962D9ECE03D1ACD4C76089DCE131543’

D.3 АУТЕНТИФИКАЦИЯ И УСТАНОВЛЕНИЕ СЕАНСОВЫХ КЛЮЧЕЙ

В данном разделе приводится пример осуществления базового контроля доступа.

Система проверки:

1. Запрос 8-байтового произвольного числа с бесконтактной ИС электронного МСПД:

APDU команды:				
CLA	**INS**	**P1**	**P2**	**Le**
00	84	00	00	08

APDU ответа:	
Поле данных ответа	**SW1-SW2**
RND.IC	9000

RND.IC = ‘4608F91988702212’

2. Генерирование 8-байтового и 16-байтового произвольного числа:
RND.IFD = ‘781723860C06C226’
K_{IFD} = ‘0B795240CB7049B01C19B33E32804F0B’

3. Конкатенация RND.IFD, RND.IC и K_{IFD}:
S = ‘781723860C06C2264608F91988702212
0B795240CB7049B01C19B33E32804F0B’

4. Шифрование S ключом 3DES K_{Enc}:
E_{IFD} = ‘72C29C2371CC9BDB65B779B8E8D37B29
ECC154AA56A8799FAE2F498F76ED92F2’

5. Вычисление MAC по E_{IFD} ключом 3DES K_{MAC}:
   M_{IFD} = ‘5F1448EEA8AD90A7’

6. Построение данных команды EXTERNAL AUTHENTICATE и посылка APDU команды на бесконтактную ИС электронного МСПД:
   cmd_data = ‘72C29C2371CC9BDB65B779B8E8D37B29ECC154AA
   56A8799FAE2F498F76ED92F25F1448EEA8AD90A7’

APDU команды:						
CLA	**INS**	**P1**	**P2**	**Lc**	**Поле данных команды**	**Le**
00	82	00	00	28	cmd_data	28

Бесконтактная ИС электронного МСПД:

1. Дешифрование и верификация полученных данных и сравнение RND.IC с ответом на команду GET CHALLENGE.

2. Генерирование 16-байтового произвольного числа:
   K_{IC} = ‘0B4F80323EB3191CB04970CB4052790B’

3. Вычисление XOR K_{IFD} и K_{IC}:
   K_{seed} = ‘0036D272F5C350ACAC50C3F572D23600’

4. Вычисление сеансовых ключей (KS_{Enc} и KS_{MAC}) согласно разделу 9.7.1/добавлению D.1:
   KS_{Enc} = ‘979EC13B1CBFE9DCD01AB0FED307EAE5’
   KS_{MAC} = ‘F1CB1F1FB5ADF208806B89DC579DC1F8’

5. Вычисление счетчика посылаемых блоков:
   SSC = ‘887022120C06C226’

6. Конкатенация RND.IC, RND.IFD и K_{IC}:
   R = ‘4608F91988702212781723860C06C226
   0B4F80323EB3191CB04970CB4052790B’

7. Шифрование R с ключом 3DES K_{Enc}:
   E_{IC} = ‘46B9342A41396CD7386BF5803104D7CE
   DC122B9132139BAF2EEDC94EE178534F’

8. Вычисление MAC по E_{IC} с ключом 3DES K_{MAC}:
   M_{IC} = ‘2F2D235D074D7449’

9. Построение данных ответа на команду EXTERNAL AUTHENTICATE и посылка APDU ответа в систему проверки:
   resp_data = ‘46B9342A41396CD7386BF5803104D7CEDC122B91
   32139BAF2EEDC94EE178534F2F2D235D074D7449’

APDU ответа:	
Поле данных ответа	**SW1-SW2**
resp_data	9000

Система проверки:

1. Дешифрование и верификация полученных данных и сравнение полученного RND.IFD с генерированным RND.IFD.

2. Вычисление XOR of K_{IFD} и K_{IC}:
   K_{seed} = ‘0036D272F5C350ACAC50C3F572D23600’

3. Вычисление сеансовых ключей (KS_{Enc} и KS_{MAC}) согласно разделу 9.7.1/добавлению D.1:
   KS_{Enc} = ‘979EC13B1CBFE9DCD01AB0FED307EAE5’
   KS_{MAC} = ‘F1CB1F1FB5ADF208806B89DC579DC1F8’

4. Вычисление счетчика посылаемых блоков:
   SSC = ‘887022120C06C226’

D.4 БЕЗОПАСНЫЙ ОБМЕН СООБЩЕНИЯМИ

После аутентификации и установления сеансовых ключей система проверки выбирает EF.COM (файл ID = ‘011E’) и считывает данные, используя метод безопасного обмена сообщениями. Будут использоваться вычисленные KS_{Enc}, KS_{MAC} и SSC (предыдущие этапы 3 и 4 системы проверки).

Сначала выбирается EF.COM, затем первые четыре байта этого файла считываются для определения длины структуры файла, после чего считываются остальные байты.

1. Выбор EF.COM

   Незащищенный APDU команды:

CLA	INS	P1	P2	Lc	Поле данных команды
00	A4	02	0C	02	01 1E

   a) Маскирование байта класса и заполнение заголовка команды:
      Заголовок команды = ‘0CA4020C80000000’

   b) Данные заполнения:
      Данные = ‘011E800000000000’

   c) Шифрование данных с KS_{Enc}:
      Зашифрованные данные = ‘6375432908C044F6’

   d) Построение DO‘87’:
      DO87 = ‘8709016375432908C044F6’

   e) Конкатенация заголовка команды и DO‘87’:
      M = ‘0CA4020C8000000087090163754329 08C044F6’

f) Вычисление MAC от M:

i) приращение SSC на 1:

SSC = ‘887022120C06C227’

ii) конкатенация SSC и M и добавление заполнения:

N = ‘887022120C06C2270CA4020C80000000
8709016375432908C044F68000000000’

iii) вычисление MAC по N с KS_{MAC}:

CC = ‘BF8B92D635FF24F8’

g) Построение DO‘8E’:

DO8E = ‘8E08BF8B92D635FF24F8’

h) Построение и посылка защищенного APDU:

Защищенный APDU =‘0CA4020C158709016375432908C0
44F68E08BF8B92D635FF24F800’

i) Получение APDU ответа бесконтактной ИС электронного МСПД:

RAPDU = ‘990290008E08FA855A5D4C50A8ED9000’

j) Верификация RAPDU CC путем вычисления MAC DO‘99’:

i) приращение SSC на 1:

SSC = ‘887022120C06C228’

ii) конкатенация SSC и DO‘99’ и добавление заполнения:

K = ‘887022120C06C2289902900080000000’

iii) вычисление MAC с KS_{MAC}:

CC’ = ‘FA855A5D4C50A8ED’

iv) сравнение CC’ с данными DO‘8E’ RAPDU.

‘FA855A5D4C50A8ED’ == ‘FA855A5D4C50A8ED’ ? YES.

2. Считывание бинарных данных первых четырех байтов:

Незащищенный APDU команды:

CLA	INS	P1	P2	Le
00	B0	00	00	04

a) Маскирование байта класса и заполнение заголовка команды:

Заголовок команды = ‘0CB0000080000000’

b) Построение DO‘97’:

DO97 = ‘970104’

c) Конкатенация заголовка команды и DO‘97’:

M = ‘0CB00000800000000970104’

d) Вычисление MAC от M:

i) приращение SSC на 1:
SSC = ‘887022120C06C229’

ii) конкатенация SSC и M и добавление заполнения:
N = ‘887022120C06C2290CB00000
800000009701048000000000’

iii) вычисление MAC по N с KSMAC:
CC = ‘ED6705417E96BA55’

e) Построение DO‘8E’:
DO8E = ‘8E08ED6705417E96BA55’

f) Построение и посылка защищенного APDU:
Защищенный APDU = ‘0CB000000D9701048E08ED6705417E96BA5500’

g) Получение APDU ответа бесконтактной ИС электронного МСПД:
RAPDU = ‘8709019FF0EC34F992265199029000
8E08AD55CC17140B2DED9000’

h) Верификация RAPDU CC путем вычисления MAC конкатенации DO‘87’ и DO‘99’:

i) приращение SSC на 1:
SSC = ‘887022120C06C22A’

ii) конкатенация SSC, DO‘87’ и DO‘99’ и добавление заполнения:
K = ‘887022120C06C22A8709019F
F0EC34F99226519902900080’

iii) вычисление MAC с KS_{MAC}:
CC’ = ‘AD55CC17140B2DED’

iv) сравнение CC’ с данными DO‘8E’ RAPDU:
‘AD55CC17140B2DED’ == ‘AD55CC17140B2DED’ ? YES.

i) Дешифрование данных DO‘87’ с KS_{Enc}:
Дешифрованные данные = ‘60145F01’

j) Определение длины структуры:
L = ‘14’ + 2 = 22 байта

3. Считывание бинарных данных остальных 18 байтов от смещения 4:

Незащищенный APDU команды:

CLA	INS	P1	P2	Le
00	B0	00	04	12

a) Маскирование байта класса и заполнение заголовка команды:
Заголовок команды = ‘0CB0000480000000’

b) Построение DO‘97’:
DO97 = ‘970112’

c) Конкатенация заголовка команды и DO‘97’:
M = ‘0CB000048000000970112’

d) Вычисление MAC от M:

i) приращение SSC на 1:
SSC = ‘887022120C06C22B’

ii) конкатенация SSC и M и добавление заполнения:
N = ‘887022120C06C22B0CB00004
80000000970112800000000’

iii) вычисление MAC по N с KS_{MAC}:
CC = ‘2EA28A70F3C7B535’

e) Построение DO‘8E’:
DO8E = ‘8E082EA28A70F3C7B535’

f) Построение и посылка защищенного APDU:
Защищенный APDU = ‘0CB000040D9701128E082EA28A70F3C7B53500’

g) Получение APDU ответа бесконтактной ИС электронного МСПД:
RAPDU = ‘871901FB9235F4E4037F2327DCC8964F1F9B8C30F42
C8E2FFF224A990290008E08C8B2787EAEA07D749000’

h) Верификация RAPDU CC путем вычисления MAC конкатенации DO‘87’ и DO‘99’:

i) приращение SSC на 1:
SSC = ‘887022120C06C22C’

ii) конкатенация SSC, DO‘87’ и DO‘99’ и добавление заполнения:
K = ‘887022120C06C22C871901FB9235F4E4037F232
7DCC8964F1F9B8C30F42C8E2FFF224A99029000’

iii) вычисление MAC по KS_{MAC}:
CC’ = ‘C8B2787EAEA07D74’

iv) сравнение CC’ с данными DO‘8E’ RAPDU:
‘C8B2787EAEA07D74’ == ‘C8B2787EAEA07D74’ ? YES.

i) Расшифровка данных DO‘87’ с KS_{Enc}:
Расшифрованные данные = ‘04303130365F36063034303030305C026175’

РЕЗУЛЬТАТ:

данные EF.COM = ‘60145F0104303130365F36063034303030305C026175’

— — — — — — — —

Добавление E к части 11

ПРИМЕР С РЕШЕНИЯМИ: ПАССИВНАЯ АУТЕНТИФИКАЦИЯ (ИНФОРМАЦИОННОЕ)

Этап 1. Считывание объекта защиты документа (SO_D) (факультативно содержит сертификат лица, подписывающего документы (C_{DS})) с бесконтактной ИС.

Этап 2. Считывание данных лица, подписывающего документы (DS), с объекта защиты документа (SO_D).

Этап 3. Верификация SO_D системой проверки путем использования открытого ключа лица, подписывающего документы.

Этап 4. Верификация C_{DS} системой проверки путем использования открытого ключа подписывающегося CA страны.

Если обе верификации на этапе 3 и 4 правильные, то это означает, что содержанию SO_D можно доверять и его можно использовать в процессе проверки.

Этап 5. Считывание соответствующих групп данных с LDS.

Этап 6. Вычисление хэш-значений соответствующих групп данных.

Этап 7. Сравнение вычисленных хэш-значений с соответствующими хэш-значениями в SO_D.

Если хэш-значения на этапе 7 идентичны, это означает, что содержание группы данных является аутентичным и не изменено.

— — — — — — — —

Добавление F к части 11

ПРИМЕР С РЕШЕНИЯМИ: АКТИВНАЯ АУТЕНТИФИКАЦИЯ (ИНФОРМАЦИОННОЕ)

В этом примере используются следующие установочные параметры:

1. Механизм, основанный на целостной факторизации: RSA

2. Длина модуля (k): 1024 бит (128 байтов)

3. Алгоритм хэширования: SHA-1

Система проверки:

Этап 1. Генерирование 8-байтового произвольного числа:
RND.IFD = ‘F173589974BF40C6’

Этап 2. Построение команды внутренней аутентификации и посылка APDU команды на бесконтактную ИС электронного МСПД:

APDU команды

CLA	INS	P1	P2	Lc	Поле данных команды	Le
00	88	00	00	08	RND.IFD	00

Бесконтактная ИС электронного МСПД:

Этап 3. Определение M_2 из входящего APDU:
M_2 = ‘F173589974BF40C6’

Этап 4. Создание завершителя:
T = ‘BC’ (т. е. SHA-1)
t (длина T в октетах) = 1

Этап 5. Определение длины:
a. $c = k – L_h – 8t – 4 = 1024 – 160 – 8 – 4 = 852$ bits
b. $L_{M1} = c – 4 = 848$ bits

Этап 6. Генерирование одноразового идентификатора M_1 длиной L_{M1}:
M_1 = ‘9D2784A67F8E7C659973EA1AEA25D95B
6C8F91E5002F369F0FBDCE8A3CEC1991
B543F1696546C5524CF23A5303CD6C98
599F40B79F377B5F3A1406B3B4D8F967
84D23AA88DB7E1032A405E69325FA91A
6E86F5C71AEA978264C4A207446DAD4E
7292E2DCDA3024B47DA8’

Этап 7. Создание M:

M = M_1 | M_2 =‘9D2784A67F8E7C659973EA1AEA25D95B
6C8F91E5002F369F0FBDCE8A3CEC1991
B543F1696546C5524CF23A5303CD6C98
599F40B79F377B5F3A1406B3B4D8F967
84D23AA88DB7E1032A405E69325FA91A
6E86F5C71AEA978264C4A207446DAD4E
7292E2DCDA3024B47DA8F173589974BF
40C6’

Этап 8. Вычисление SHA-1 краткой формы M:

H = SHA-1(M) = ‘C063AA1E6D22FBD976AB0FE73D94D2D9
C6D88127’

Этап 9.[2] Построение репрезентатива сообщения:

F = ‘6A’ | M_1 | H | T =
‘6A9D2784A67F8E7C659973EA1AEA25D9
5B6C8F91E5002F369F0FBDCE8A3CEC19
91B543F1696546C5524CF23A5303CD6C
98599F40B79F377B5F3A1406B3B4D8F9
6784D23AA88DB7E1032A405E69325FA9
1A6E86F5C71AEA978264C4A207446DAD
4E7292E2DCDA3024B47DA8C063AA1E6D
22FBD976AB0FE73D94D2D9C6D88127BC’

Этап 10. Шифрование F с помощью открытого ключа активной аутентификации для формирования подписи:

S = ‘756B683B036A6368F4A2EB29EA700F96
E26100AFC0809F60A91733BA29CAB362
8CB1A017190A85DADE83F0B977BB513F
C9C672E5C93EFEBBE250FE1B722C7CEE
F35D26FC8F19219C92D362758FA8CB0F
F68CEF320A8753913ED25F69F7CEE772
6923B2C43437800BBC9BC028C49806CF
2E47D16AE2B2CC1678F2A4456EF98FC9’

Этап 11. Построение данных ответа на команду INTERNAL AUTHENTICATE и посылка APDU ответа в систему проверки:

APDU ответа:

Поле данных ответа	**SW1-SW2**
S	9000

2 Поскольку известная часть (RND.IFD) не возвращена, но должна быть добавлена самим IFD, то применяется частичное восстановление (‘6A’).

Система проверки:

Этап 12. Дешифровка подписи с помощью открытого ключа:

F = ‘6A9D2784A67F8E7C659973EA1AEA25D9
5B6C8F91E5002F369F0FBDCE8A3CEC19
91B543F1696546C5524CF23A5303CD6C
98599F40B79F377B5F3A1406B3B4D8F9
6784D23AA88DB7E1032A405E69325FA9
1A6E86F5C71AEA978264C4A207446DAD
4E7292E2DCDA3024B47DA8C063AA1E6D
22FBD976AB0FE73D94D2D9C6D88127BC’

Этап 13. Определение хэш-алгоритма по концевику T*:

T = ‘BC’ (т. е. SHA-1)

Этап 14. Выделение краткой формы:

D = ‘C063AA1E6D22FBD976AB0FE73D94D2D9
C6D88127’

Этап 15. Выделение M_1:

M_1 =‘9D2784A67F8E7C659973EA1AEA25D95B
6C8F91E5002F369F0FBDCE8A3CEC1991
B543F1696546C5524CF23A5303CD6C98
599F40B79F377B5F3A1406B3B4D8F967
84D23AA88DB7E1032A405E69325FA91A
6E86F5C71AEA978264C4A207446DAD4E
7292E2DCDA3024B47DA8’

Этап 16. Заголовок указывает частичное восстановление, но подпись имеет длину модуля для конкатенации M_1 с известным M_2 (т. е. RND.IFD):

M* = ‘9D2784A67F8E7C659973EA1AEA25D95B
6C8F91E5002F369F0FBDCE8A3CEC1991
B543F1696546C5524CF23A5303CD6C98
599F40B79F377B5F3A1406B3B4D8F967
84D23AA88DB7E1032A405E69325FA91A
6E86F5C71AEA978264C4A207446DAD4E
7292E2DCDA3024B47DA8F173589974BF
40C6’

Этап 17. Вычисление SHA-1 краткой формы M*:

D* =‘C063AA1E6D22FBD976AB0FE73D94D2D9
C6D88127’

Этап 18. Сравнение D и D*:

D равняется D*, т. е. верификация прошла успешно.

— — — — — — — —

Добавление G к части 11

ПРИМЕР С РЕШЕНИЯМИ: PACE – ОТОБРАЖЕНИЕ ОБЩЕГО ТИПА (ИНФОРМАЦИОННОЕ)

В настоящем добавлении приводятся два примера с решениями для протокола PACE, изложенного в разделе 4.4, с использованием отображения общего типа. Первый пример основан на ECDH, а второй – на использовании DH. Все номера, содержащиеся в таблицах, представляют собой шестнадцатеричные значения.

В обоих примерах в качестве пароля используется MRZ. Это также означает использование одного и того же симметричного ключа K_{π}. Соответствующие поля данных MRZ, в том числе контрольные цифры, включают следующее:

- номер документа: T220001293;
- дату рождения: 6408125;
- дату истечения срока действия: 1010318.

Таким образом, кодировкой K зоны MRZ и выработанным ключом кодировки K_{π} являются:

K	7E2D2A41 C74EA0B3 8CD36F86 3939BFA8 E9032AAD
K_{π}	89DED1B2 6624EC1E 634C1989 302849DD

G.1 ПРИМЕР, ОСНОВАННЫЙ НА ECDH

Этот пример основан на ECDH с применением стандартизированных по BrainpoolP256r1 параметров домена (см. [RFC 5639]).

В первом разделе приводится соответствующая информация PACE. Впоследствии перечисляются и рассматриваются обмениваемые блоки APDU, включая все генерированные одноразовые идентификаторы и эфемерные ключи.

Параметры эллиптической кривой

При использовании стандартизированных параметров домена вся информация, необходимая для осуществления PACE, содержится в структуре данных "Информация PACE". В частности, никакой информации о параметрах домена PACE не требуется.

Информация PACE	3012060A 04007F00 07020204 02020201 0202010D

Подробная структура информации PACE разбита по пунктам в следующей таблице.

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***
30	12		ПОСЛЕДОВА-ТЕЛЬНОСТЬ	Информация PACE
06	0A	04 00 7F 00 07 02 02 04 02 02	ИДЕНТИФИКА-ТОР ОБЪЕКТА	PACE с ECDH, отображение общего типа и сеансовые ключи AES 128
02	01	02	ЦЕЛОЕ ЧИСЛО	Версия 2
02	01	0D	ЦЕЛОЕ ЧИСЛО	Стандартизированные параметры домена по Brainpool P256r1

Для удобства ниже приводится основанное на ASN.1 кодирование параметров домена по BrainpoolP256r1.

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***
30	81 EC		ПОСЛЕДОВА-ТЕЛЬНОСТЬ	Параметры домена
06	07	2A 86 48 CE 3D 02 01	ИДЕНТИФИКА-ТОР ОБЪЕКТА	Алгоритм id-ecPublicKey
30	81 E0		ПОСЛЕДОВА-ТЕЛЬНОСТЬ	Параметры домена
02	01	01	ЦЕЛОЕ ЧИСЛО	Версия
30	2C		ПОСЛЕДОВА-ТЕЛЬНОСТЬ	Исходное поле
06	07	2A 86 48 CE 3D 01 01	ИДЕНТИФИКА-ТОР ОБЪЕКТА	Простое поле
02	21	00 A9 FB 57 DB A1 EE A9 BC 3E 66 0A 90 9D 83 8D 72 6E 3B F6 23 D5 26 20 28 20 13 48 1D 1F 6E 53 77	ЦЕЛОЕ ЧИСЛО	Простое p
30	44		ПОСЛЕДОВА-ТЕЛЬНОСТЬ	Уравнение кривой
04	20	7D 5A 09 75 FC 2C 30 57 EE F6 75 30 41 7A FF E7 FB 80 55 C1 26 DC 5C 6C E9 4A 4B 44 F3 30 B5 D9	ОКТЕТНАЯ СТРОКА	Параметр a
04	20	26 DC 5C 6C E9 4A 4B 44 F3 30 B5 D9 BB D7 7C BF 95 84 16 29 5C F7 E1 CE 6B CC DC 18 FF 8C 07 B6	ОКТЕТНАЯ СТРОКА	Параметр b
04	41		ОКТЕТНАЯ СТРОКА	Генератор группы G

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***		
		04	-			Несжатая точка
		8B D2 AE B9 CB 7E 57 CB 2C 4B 48 2F FC 81 B7 AF B9 DE 27 E1 E3 BD 23 C2 3A 44 53 BD 9A CE 32 62	-			Координата x
		54 7E F8 35 C3 DA C4 FD 97 F8 46 1A 14 61 1D C9 C2 77 45 13 2D ED 8E 54 5C 1D 54 C7 2F 04 69 97	-			Координата y
02	21	00 A9 FB 57 DB A1 EE A9 BC 3E 66 0A 90 9D 83 8D 71 8C 39 7A A3 B5 61 A6 F7 90 1E 0E 82 97 48 56 A7	ЦЕЛОЕ ЧИСЛО		Порядок группы n	
02	01	01	ЦЕЛОЕ ЧИСЛО		Сомножитель f	

Блок-схема приложения применительно к примеру, основанному на ECDH

Для инициализации PACE терминал посылает на чип команду MSE:Set AT.

T>C :	`00 22 C1 A4 0F 80 0A 04 00 7F 00 07 02 02 04 02 02 83 01 01`
C>T :	`90 00`

Здесь T>C является сокращением для APDU, посланного с терминала на чип, в то время как C>T обозначает соответствующий ответ, посланный чипом на терминал. Шифрование команды поясняется в ниже-следующей таблице.

Команда				
CLA	00		Открытое	
INS	22		Управление средствами защиты	
P1/P2	C1 A4		Установка шаблона аутентификации для взаимной аутентификации	
Lc	0F		Длина поля данных	
Данные	Тег	Длина	Значение	Замечания
	80	0A	04 00 7F 00 07 02 02 04 02 02	Криптографический механизм: PACE с ECDH, отображение общего типа и сеансовые ключи AES128
	83	01	01	Пароль: MC3
Ответ				
Байты состояния	90 00		Нормальная операция	

Зашифрованный одноразовый идентификатор

На следующем этапе чип случайным образом генерирует одноразовый идентификатор s и кодирует его с помощью K_π.

Дешифрованный одноразовый идентификатор s	3F00C4D3 9D153F2B 2A214A07 8D899B22
Зашифрованный одноразовый идентификатор z	95A3A016 522EE98D 01E76CB6 B98B42C3

Зашифрованный одноразовый идентификатор запрашивается терминалом.

T>C :	10 86 00 00 02 7C 00 00
C>T :	7C 12 80 10 95 A3 A0 16 52 2E E9 8D 01 E7 6C B6 B9 8B 42 C3 90 00

Кодировка APDU команды и соответствующий ответ указываются в следующей таблице.

Команда				
CLA	10		Составление последовательности команд	
INS	86		GENERAL AUTHENTICATE	
P1/P2	00 00		Ключи и протоколы, известные в виде подразумеваемых	
Lc	02		Длина данных	
Данные	Тег	Длина	Значение	Замечания
	7C	00	-	Отсутствуют
Le	00		Ожидаемая максимальная длина поля данных ответа составляет 256 байтов	
Ответ				
Данные	Тег	Длина	Значение	Замечания
	7C	12		Динамические данные аутентификации
	80	10	95 A3 A0 16 52 2E E9 8D 01 E7 6C B6 B9 8B 42 C3	Зашифрованный одноразовый идентификатор
Байты состояния	90 00		Нормальная операция	

Одноразовый идентификатор отображения

Этот одноразовый идентификатор отображается в генераторе эфемерной группы с помощью отображения общего типа. В нижеследующей таблице также собраны требуемые эфемерные ключи, выбираемые случайным образом.

Закрытый ключ терминала	7F4EF07B 9EA82FD7 8AD689B3 8D0BC78C F21F249D 953BC46F 4C6E1925 9C010F99
Открытый ключ терминала	7ACF3EFC 982EC455 65A4B155 129EFBC7 4650DCBF A6362D89 6FC70262 E0C2CC5E, 544552DC B6725218 799115B5 5C9BAA6D 9F6BC3A9 618E70C2 5AF71777 A9C4922D
Закрытый ключ чипа	498FF497 56F2DC15 87840041 839A8598 2BE7761D 14715FB0 91EFA7BC E9058560
Открытый ключ чипа	824FBA91 C9CBE26B EF53A0EB E7342A3B F178CEA9 F45DE0B7 0AA60165 1FBA3F57, 30D8C879 AAA9C9F7 3991E61B 58F4D52E B87A0A0C 709A49DC 63719363 CCD13C54
Совместно используемый секретный H	60332EF2 450B5D24 7EF6D386 8397D398 852ED6E8 CAF6FFEE F6BF85CA 57057FD5, 0840CA74 15BAF3E4 3BD414D3 5AA4608B 93A2CAF3 A4E3EA4E 82C9C13D 03EB7181
Отображаемый генератор $\hat{G}$	8CED63C9 1426D4F0 EB1435E7 CB1D74A4 6723A0AF 21C89634 F65A9AE8 7A9265E2, 8C879506 743F8611 AC33645C 5B985C80 B5F09A0B 83407C1B 6A4D857A E76FE522

Для отображения одноразового идентификатора терминал и чип обмениваются следующими блоками APDU.

T>C :	10 86 00 00 45 7C 43 81 41 04 7A CF 3E FC 98 2E C4 55 65 A4 B1 55 12 9E FB C7 46 50 DC BF A6 36 2D 89 6F C7 02 62 E0 C2 CC 5E 54 45 52 DC B6 72 52 18 79 91 15 B5 5C 9B AA 6D 9F 6B C3 A9 61 8E 70 C2 5A F7 17 77 A9 C4 92 2D 00
C>T :	7C 43 82 41 04 82 4F BA 91 C9 CB E2 6B EF 53 A0 EB E7 34 2A 3B F1 78 CE A9 F4 5D E0 B7 0A A6 01 65 1F BA 3F 57 30 D8 C8 79 AA A9 C9 F7 39 91 E6 1B 58 F4 D5 2E B8 7A 0A 0C 70 9A 49 DC 63 71 93 63 CC D1 3C 54 90 00

Структуру блоков APDU можно описать следующим образом:

Команда					
CLA	10		Составление последовательности команд		
INS	86		GENERAL AUTHENTICATE		
P1/P2	00 00		Ключи и протоколы, известные в виде подразумеваемых		
Lc	45		Длина данных		
Данные	Тег	Длина	Значение	Замечания	
	7C	43	-	Динамические аутентификационные данные	
	81	41		Данные отображения	
			04		Несжатая точка
			7A CF 3E FC 98 2E ... C2 CC 5E		Координата x
			54 45 52 DC B6 72 ... C4 92 2D		Координата y
Le	00		Ожидаемая максимальная длина поля данных ответа составляет 256 байтов		
Ответ					
Данные	Тег	Длина	Значение	Замечания	
	7C	43		Динамические аутентификационные данные	
	82	41		Данные отображения	
			04		Несжатая точка
			82 4F BA 91 C9 CB ... BA 3F 57		Координата x
			30 D8 C8 79 AA A9 ... D1 3C 54		Координата y
Байты состояния	90 00		Нормальная операция		

Выполнение согласования ключей

На третьем этапе чип и терминал выполняют анонимное согласование ключей на основе ECDH, используя параметры домена, установленные генератором эфемерной группы на предыдущем этапе. В качестве совместно используемого секретного параметра требуется только координата x, поскольку для выработки сеансовых ключей KDF использует только первую координату.

Закрытый ключ терминала	A73FB703 AC1436A1 8E0CFA5A BB3F7BEC 7A070E7A 6788486B EE230C4A 22762595
Открытый ключ терминала	2DB7A64C 0355044E C9DF1905 14C625CB A2CEA487 54887122 F3A5EF0D 5EDD301C, 3556F3B3 B186DF10 B857B58F 6A7EB80F 20BA5DC7 BE1D43D9 BF850149 FBB36462
Закрытый ключ чипа	107CF586 96EF6155 053340FD 633392BA 81909DF7 B9706F22 6F32086C 7AFF974A
Открытый ключ чипа	9E880F84 2905B8B3 181F7AF7 CAA9F0EF B743847F 44A306D2 D28C1D9E C65DF6DB, 7764B222 77A2EDDC 3C265A9F 018F9CB8 52E111B7 68B32690 4B59A019 3776F094
Совместно используемый секретный параметр	28768D20 701247DA E81804C9 E780EDE5 82A9996D B4A31502 0B273319 7DB84925

Согласование ключей выполняется следующим образом:

T>C :	10 86 00 00 45 7C 43 83 41 04 2D B7 A6 4C 03 55 04 4E C9 DF 19 05 14 C6 25 CB A2 CE A4 87 54 88 71 22 F3 A5 EF 0D 5E DD 30 1C 35 56 F3 B3 B1 86 DF 10 B8 57 B5 8F 6A 7E B8 0F 20 BA 5D C7 BE 1D 43 D9 BF 85 01 49 FB B3 64 62 00
C>T :	7C 43 84 41 04 9E 88 0F 84 29 05 B8 B3 18 1F 7A F7 CA A9 F0 EF B7 43 84 7F 44 A3 06 D2 D2 8C 1D 9E C6 5D F6 DB 77 64 B2 22 77 A2 ED DC 3C 26 5A 9F 01 8F 9C B8 52 E1 11 B7 68 B3 26 90 4B 59 A0 19 37 76 F0 94 90 00

Шифрование согласования ключей рассматривается в следующей таблице:

Команда					
CLA	10		Составление последовательности команд		
INS	86		GENERAL AUTHENTICATE		
P1/P2	00 00		Ключи и протокол, известные в виде подразумеваемых		
Lc	45		Длина данных		
Данные	Тег	Длина	Значение	Замечания	
	7C	43	-	Динамические аутентификационные данные	
	83	41		Эфемерный открытый ключ терминала	
			04		Несжатая точка
			2D B7 A6 4C 03 55 ... DD 30 1C		Координата x

			35 56 F3 B3 B1 86 ... B3 64 62		Координата y
Le	00		Ожидаемая максимальная длина поля данных ответа составляет 256 байтов		
Ответ					
Данные	Тег	Длина	Значение	Замечания	
	7C	43		Динамические аутентификационные данные	
	84	41		Эфемерный открытый ключ чипа	
			04		Несжатая точка
			9E 88 0F 84 29 05 ... 5D F6 DB		Координата x
			77 64 B2 22 77 A2 ... 76 F0 94		Координата y
Байты состояния	90 00		Нормальная операция		

С помощью KDF и совместно используемого секретного ключа вырабатываются сеансовые ключи KS_{Enc} и KS_{MAC} на основе AES 128. Таковыми являются:

KS_{Enc}	F5F0E35C 0D7161EE 6724EE51 3A0D9A7F
KS_{MAC}	FE251C78 58B356B2 4514B3BD 5F4297D1

Взаимная аутентификация

Аутентификационные маркерные изображения вырабатываются с помощью ключа KS_{MAC}, используя

Вводные данные для T_{IFD}	7F494F06 0A04007F 00070202 04020286 41049E88 0F842905 B8B3181F 7AF7CAA9 F0EFB743 847F44A3 06D2D28C 1D9EC65D F6DB7764 B22277A2 EDDC3C26 5A9F018F 9CB852E1 11B768B3 26904B59 A0193776 F094
Вводные данные для T_{IC}	7F494F06 0A04007F 00070202 04020286 41042DB7 A64C0355 044EC9DF 190514C6 25CBA2CE A4875488 7122F3A5 EF0D5EDD 301C3556 F3B3B186 DF10B857 B58F6A7E B80F20BA 5DC7BE1D 43D9BF85 0149FBB3 6462

в качестве вводных параметров. Шифрование вводных данных показано ниже.

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***
7F49	4F		ОТКРЫТЫЙ КЛЮЧ	Вводные данные для T_{IFD}
06	0A	04 00 7F 00 07 02 02 04 02 02	ИДЕНТИФИКАТОР ОБЪЕКТА	PACE с ECDH, отображение общего типа и сеансовые ключи AES 128
86	41		ТОЧКА ЭЛЛИПТИЧЕСКОЙ КРИВОЙ	Эфемерная открытая точка чипа
		04		Несжатая точка
		9E 88 0F 84 29 ... 5D F6 DB		Координата x
		77 64 B2 22 77 ... 76 F0 94		Координата y

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***
7F49	4F		ОТКРЫТЫЙ КЛЮЧ	Вводные данные для T_{IC}
06	0A	04 00 7F 00 07 02 02 04 02 02	ИДЕНТИФИКАТОР ОБЪЕКТА	PACE с ECDH, отображение общего типа и сеансовые ключи AES 128
86	41		ТОЧКА ЭЛЛИПТИЧЕСКОЙ КРИВОЙ	Эфемерная открытая точка терминала
		04		Несжатая точка
		2D B7 A6 4C 03 ... DD 30 1C		Координата x
		35 56 F3 B3 B1 ... B3 64 62		Координата y

Вычисленными аутентификационными маркерными изображениями являются:

T_{IFD}	C2B0BD78 D94BA866
T_{IC}	3ABB9674 BCE93C08

Наконец, производятся обмен этими маркерными изображениями и их верификация.

T>C :	00 86 00 00 0C 7C 0A 85 08 C2 B0 BD 78 D9 4B A8 66 00
C>T :	7C 0A 86 08 3A BB 96 74 BC E9 3C 08 90 00

G.2 ПРИМЕР, ОСНОВАННЫЙ НА DH

Второй пример основан на DH с использованием 1024-битной группы MODP с 160-битной подгруппой с простым порядком, как указано в [RFC 5114]. Параметрами группы являются:

Простое *p*	B10B8F96 A080E01D DE92DE5E AE5D54EC 52C99FBC FB06A3C6 9A6A9DCA 52D23B61 6073E286 75A23D18 9838EF1E 2EE652C0 13ECB4AE A9061123 24975C3C D49B83BF ACCBDD7D 90C4BD70 98488E9C 219A7372 4EFFD6FA E5644738 FAA31A4F F55BCCC0 A151AF5F 0DC8B4BD 45BF37DF 365C1A65 E68CFDA7 6D4DA708 DF1FB2BC 2E4A4371
Генератор подгруппы *g*	A4D1CBD5 C3FD3412 6765A442 EFB99905 F8104DD2 58AC507F D6406CFF 14266D31 266FEA1E 5C41564B 777E690F 5504F213 160217B4 B01B886A 5E91547F 9E2749F4 D7FBD7D3 B9A92EE1 909D0D22 63F80A76 A6A24C08 7A091F53 1DBF0A01 69B6A28A D662A4D1 8E73AFA3 2D779D59 18D08BC8 858F4DCE F97C2A24 855E6EEB 22B3B2E5
Простой порядок *q* в рамках *g*	F518AA87 81A8DF27 8ABA4E7D 64B7CB9D 49462353

В первом разделе представляется структура "Информация PACE". Впоследствии перечисляются и рассматриваются блоки APDU, которыми был произведен обмен, в том числе все сгенерированные одноразовые идентификаторы и эфемерные ключи.

Параметры Диффи-Хеллмана

Соответствующая связанная с PACE информация предоставляется структурой данных `PACEInfo`.

PACEInfo	3012060A 04007F00 07020204 01020201 02020100

Детальная структура `PACEInfo` представляет собой следующее:

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***	
30	12		ПОСЛЕДОВА-ТЕЛЬНОСТЬ	PACEInfo	
06	0A	04 00 7F 00 07 02 02 04 01 02	ИДЕНТИФИКА-ТОР ОБЪЕКТА		OID: PACE с DH, отображение общего типа и сеансовые ключи AES 128
02	01	02	ЦЕЛОЕ ЧИСЛО		Версия 2
02	01	00	ЦЕЛОЕ ЧИСЛО		Стандартизированная 1024-битная группа, указанная в RFC 5114

Блок-схема приложения применительно к примеру, основанному на DH

Для инициализации PACE терминал посылает на чип команду MSE:AT.

T>C :	00 22 C1 A4 0F 80 0A 04 00 7F 00 07 02 02 04 01 02 83 01 01
C>T :	90 00

Описание шифрования команды приводится в нижеследующей таблице.

Команда				
CLA	00		Открытая	
INS	22		Управление средствами защиты	
P1/P2	C1 A4		Установка шаблона аутентификации для взаимной аутентификации	
Lc	0F		Длина поля данных	
Данные	Тег	Длина	Значение	Замечания
	80	0A	04 00 7F 00 07 02 02 04 01 02	OID: криптографический механизм: PACE с DH, отображение общего типа и AES128
	83	01	01	Пароль: MC3
Ответ				
Байты состояния	90 00		Нормальная операция	

Зашифрованный одноразовый идентификатор

Затем терминал запрашивает у чипа одноразовый идентификатор.

Дешифрованный одноразовый идентификатор s	FA5B7E3E 49753A0D B9178B7B 9BD898C8
Зашифрованный одноразовый идентификатор z	854D8DF5 827FA685 2D1A4FA7 01CDDDCA

Обмен сообщениями выглядит следующим образом:

T>C :	10 86 00 00 02 7C 00 00
C>T :	7C 12 80 10 85 4D 8D F5 82 7F A6 85 2D 1A 4F A7 01 CD DD CA 90 00

Описание кодировки APDU команды и соответствующего ответа приводится в нижеследующей таблице.

Команда				
CLA	10		Составление последовательности команд	
INS	86		GENERAL AUTHENTICATE	
P1/P2	00 00		Ключи и протокол, известные в виде подразумеваемых	
Lc	02		Длина данных	
Данные	Тег	Длина	Значение	Замечания
	7C	00	-	Отсутствуют
Le	00		Ожидаемая максимальная длина поля данных ответа составляет 256 байтов	
Ответ				
Данные	Тег	Длина	Значение	Замечания
	7C	12		Динамические аутентификационные данные
	80	10	85 4D 8D F5 82 7F A6 85 2D 1A 4F A7 01 CD DD CA	Зашифрованный одноразовый идентификатор
Байты состояния	90 00		Нормальная операция	

Одноразовый идентификатор отображения

Посредством отображения общего типа одноразовый идентификатор отображается в генераторе эфемерной группы. Для этой цели терминал и чип случайным образом генерируют следующие эфемерные ключи.

Закрытый ключ терминала	5265030F 751F4AD1 8B08AC56 5FC7AC95 2E41618D
Открытый ключ терминала	23FB3749 EA030D2A 25B278D2 A562047A DE3F01B7 4F17A154 02CB7352 CA7D2B3E B71C343D B13D1DEB CE9A3666 DBCFC920 B49174A6 02CB4796 5CAA73DC 702489A4 4D41DB91 4DE9613D C5E98C94 160551C0 DF86274B 9359BC04 90D01B03 AD54022D CB4F57FA D6322497 D7A1E28D 46710F46 1AFE710F BBBC5F8B A166F431 1975EC6C

Закрытый ключ чипа	66DDAFEA C1609CB5 B963BB0C B3FF8B3E 047F336C
Открытый ключ чипа	78879F57 225AA808 0D52ED0F C890A4B2 5336F699 AA89A2D3 A189654A F70729E6 23EA5738 B26381E4 DA19E004 706FACE7 B235C2DB F2F38748 312F3C98 C2DD4882 A41947B3 24AA1259 AC22579D B93F7085 655AF308 89DBB845 D9E6783F E42C9F24 49400306 254C8AE8 EE9DD812 A804C0B6 6E8CAFC1 4F84D825 8950A91B 44126EE6
Совместно используемый секретный H	5BABEBEF 5B74E5BA 94B5C063 FDA15F1F 1CDE9487 3EE0A5D3 A2FCAB49 F258D07F 544F13CB 66658C3A FEE9E727 389BE3F6 CBBBD321 28A8C21D D6EEA3CF 7091CDDF B08B8D00 7D40318D CCA4FFBF 51208790 FB4BD111 E5A968ED 6B6F08B2 6CA87C41 0B3CE0C3 10CE104E ABD16629 AA48620C 1279270C B0750C0D 37C57FFF E302AE7F
Отображаемый генератор Ĝ	7C9CBFE9 8F9FBDDA 8D143506 FA7D9306 F4CB17E3 C71707AF F5E1C1A1 23702496 84D64EE3 7AF44B8D BD9D45BF 6023919C BAA027AB 97ACC771 666C8E98 FF483301 BFA4872D EDE9034E DFACB708 14166B7F 36067682 9B826BEA 57291B5A D69FBC84 EF1E7790 32A30580 3F743417 93E86974 2D401325 B37EE856 5FFCDEE6 18342DC5

Для отображения одноразового идентификатора терминал и чип обмениваются следующими блоками APDU.

T>C :	10 86 00 00 86 7C 81 83 81 81 80 23 FB 37 49 EA 03 0D 2A 25 B2 78 D2 A5 62 04 7A DE 3F 01 B7 4F 17 A1 54 02 CB 73 52 CA 7D 2B 3E B7 1C 34 3D B1 3D 1D EB CE 9A 36 66 DB CF C9 20 B4 91 74 A6 02 CB 47 96 5C AA 73 DC 70 24 89 A4 4D 41 DB 91 4D E9 61 3D C5 E9 8C 94 16 05 51 C0 DF 86 27 4B 93 59 BC 04 90 D0 1B 03 AD 54 02 2D CB 4F 57 FA D6 32 24 97 D7 A1 E2 8D 46 71 0F 46 1A FE 71 0F BB BC 5F 8B A1 66 F4 31 19 75 EC 6C 00
C>T :	7C 81 83 82 81 80 78 87 9F 57 22 5A A8 08 0D 52 ED 0F C8 90 A4 B2 53 36 F6 99 AA 89 A2 D3 A1 89 65 4A F7 07 29 E6 23 EA 57 38 B2 63 81 E4 DA 1 9E0 04 70 6F AC E7 B2 35 C2 DB F2 F3 87 48 31 2F 3C 98 C2 DD 48 82 A4 19 47 B3 24 AA 12 59 AC 22 57 9D B9 3F 70 85 65 5A F3 08 89 DB B8 45 D9 E6 78 3F E4 2C 9F 24 49 40 03 06 25 4C 8A E8 EE 9D D8 12 A8 04 C0 B6 6E 8C AF C1 4F 84 D8 25 89 50 A9 1B 44 12 6E E6 90 00

Структуру блоков APDU можно описать следующим образом:

Команда				
CLA	10		Составление последовательности команд	
INS	86		GENERAL AUTHENTICATE	
P1/P2	00 00		Ключи и протокол, известные в виде подразумеваемых	
Lc	86		Длина данных	
Данные	Тег	Длина	Значение	Замечания
	7C	81 83	-	Динамические аутентификационные данные
	81	81 80	23 FB 37 49 EA 03 ... 75 EC 6C	Данные отображения
Le	00		Ожидаемая максимальная длина поля данных ответа составляет 256 байтов	
Ответ				
Данные	Тег	Длина	Значение	Замечания
	7C	81 83		Динамические аутентификационные данные
	82	81 80	ED 0F C8 90 A4 B2 ... 12 6E E6	Данные отображения
Байты состояния	90 00		Нормальная операция	

Выполнение согласования ключей

После этого чип и терминал выполняют анонимное согласование ключей на основе DH, используя новые параметры домена, установленные генератором эфемерной группы на предыдущем этапе.

Закрытый ключ терминала	89CCD99B 0E8D3B1F 11E1296D CA68EC53 411CF2CA
Открытый ключ терминала	00907D89 E2D425A1 78AA81AF 4A7774EC 8E388C11 5CAE6703 1E85EECE 520BD911 551B9AE4 D04369F2 9A02626C 86FBC674 7CC7BC35 2645B616 1A2A42D4 4EDA80A0 8FA8D61B 76D3A154 AD8A5A51 786B0BC0 71470578 71A92221 2C5F67F4 31731722 36B7747D 1671E6D6 92A3C7D4 0A0C3C5C E397545D 015C175E B5130551 EDBC2EE5 D4
Закрытый ключ чипа	A5B78012 6B7C980E 9FCEA1D4 539DA1D2 7C342DFA

Открытый ключ чипа	075693D9 AE941877 573E634B 6E644F8E 60AF17A0 076B8B12 3D920107 4D36152B D8B3A213 F53820C4 2ADC79AB 5D0AEEC3 AEFB9139 4DA476BD 97B9B14D 0A65C1FC 71A0E019 CB08AF55 E1F72900 5FBA7E3F A5DC4189 9238A250 767A6D46 DB974064 386CD456 743585F8 E5D90CC8 B4004B1F 6D866C79 CE0584E4 9687FF61 BC29AEA1
Совместно используемый секретный ключ	6BABC7B3 A72BCD7E A385E4C6 2DB2625B D8613B24 149E146A 629311C4 CA6698E3 8B834B6A 9E9CD718 4BA8834A FF5043D4 36950C4C 1E783236 7C10CB8C 314D40E5 990B0DF7 013E64B4 549E2270 923D06F0 8CFF6BD3 E977DDE6 ABE4C31D 55C0FA2E 465E553E 77BDF75E 3193D383 4FC26E8E B1EE2FA1 E4FC97C1 8C3F6CFF FE2607FD

Согласование ключей выполняется следующим образом:

T>C :	10 86 00 00 86 7C 81 83 83 81 80 90 7D 89 E2 D4 25 A1 78 AA 81 AF 4A 77 74 EC 8E 38 8C 11 5C AE 67 03 1E 85 EE CE 52 0B D9 11 55 1B 9A E4 D0 43 69 F2 9A 02 62 6C 86 FB C6 74 7C C7 BC 35 26 45 B6 16 1A 2A 42 D4 4E DA 80 A0 8F A8 D6 1B 76 D3 A1 54 AD 8A 5A 51 78 6B 0B C0 71 47 05 78 71 A9 22 21 2C 5F 67 F4 31 73 17 22 36 B7 74 7D 16 71 E6 D6 92 A3 C7 D4 0A 0C 3C 5C E3 97 54 5D 01 5C 17 5E B5 13 05 51 ED BC 2E E5 D4 00
C>T :	7C 81 83 84 81 80 07 56 93 D9 AE 94 18 77 57 3E 63 4B 6E 64 4F 8E 60 AF 17 A0 07 6B 8B 12 3D 92 01 07 4D 36 15 2B D8 B3 A2 13 F5 38 20 C4 2A DC 79 AB 5D 0A EE C3 AE FB 91 39 4D A4 76 BD 97 B9 B1 4D 0A 65 C1 FC 71 A0 E0 19 CB 08 AF 55 E1 F7 29 00 5F BA 7E 3F A5 DC 41 89 92 38 A2 50 76 7A 6D 46 DB 97 40 64 38 6C D4 56 74 35 85 F8 E5 D9 0C C8 B4 00 4B 1F 6D 86 6C 79 CE 05 84 E4 96 87 FF 61 BC 29 AE A1 90 00

Команда		
CLA	10	Составление последовательности команд
INS	86	GENERAL AUTHENTICATE
P1/P2	00 00	Ключи и протокол, известные в виде подразумеваемых
Lc	86	Длина данных

Данные	Тег	Длина	Значение	Замечания
	7C	81 83	-	Динамические аутентификационные данные
	83	81 80	90 7D 89 E2 D4 25 ... 2E E5 D4	Эфемерные открытые ключи терминала
Le	00		Ожидаемая максимальная длина поля данных ответа составляет 256 байтов	
Ответ				
Данные	Тег	Длина	Значение	Замечания
	7C	81 83		Динамические аутентификационные данные
	84	81 80	07 56 93 D9 AE 94 ... 29 AE A1	Эфемерные открытые ключи чипа
Байты состояния	90 00		Нормальная операция	

Сеансовые ключи KS_{Enc} и KS_{MAC} на основе стандарта AES 128 вырабатываются из совместно используемого секретного ключа с применением KDF.

KS_{Enc}	2F7F46AD CC9E7E52 1B45D192 FAFA9126
KS_{MAC}	805A1D27 D45A5116 F73C5446 9462B7D8

Взаимная аутентификация

Аутентификационное маркерное изображение строится на основе следующих вводных данных.

Вводные данные для T_{IFD}	7F49818F 060A0400 7F000702 02040102 84818007 5693D9AE 94187757 3E634B6E 644F8E60 AF17A007 6B8B123D 9201074D 36152BD8 B3A213F5 3820C42A DC79AB5D 0AEEC3AE FB91394D A476BD97 B9B14D0A 65C1FC71 A0E019CB 08AF55E1 F729005F BA7E3FA5 DC418992 38A25076 7A6D46DB 97406438 6CD45674 3585F8E5 D90CC8B4 004B1F6D 866C79CE 0584E496 87FF61BC 29AEA1
Вводные данные для T_{IC}	7F49818F 060A0400 7F000702 02040102 84818090 7D89E2D4 25A178AA 81AF4A77 74EC8E38 8C115CAE 67031E85 EECE520B D911551B 9AE4D043 69F29A02 626C86FB C6747CC7 BC352645 B6161A2A 42D44EDA 80A08FA8 D61B76D3 A154AD8A 5A51786B 0BC07147 057871A9 22212C5F 67F43173 172236B7 747D1671 E6D692A3 C7D40A0C 3C5CE397 545D015C 175EB513 0551EDBC 2EE5D4

Шифрование вводных данных показано ниже:

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***	
7F49	81 8F		ОТКРЫТЫЙ КЛЮЧ	Вводные данные для T_{IFD}	
06	0A	04 00 7F 00 07 02 02 04 01 02	ИДЕНТИФИКАТОР ОБЪЕКТА		PACE с DH, отображение общего типа и сеансовые ключи AES 128
84	81 80	07 56 93 D9 AE ... 29 AE A1	НЕПОДПИСАННОЕ ЦЕЛОЕ ЧИСЛО		Эфемерный открытый ключ чипа

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***	
7F49	81 8F		ОТКРЫТЫЙ КЛЮЧ	Вводные данные для T_{IC}	
06	0A	04 00 7F 00 07 02 02 04 01 02	ИДЕНТИФИКАТОР ОБЪЕКТА		PACE с DH, отображение общего типа и сеансовые ключи AES 128
84	81 80	90 7D 89 E2 D4 ... 2E E5 D4	НЕПОДПИСАННОЕ ЦЕЛОЕ ЧИСЛО		Эфемерный открытый ключ терминала

Вычисленными аутентификационными маркерными изображениями являются:

T_{IFD}	B46DD9BD 4D98381F
T_{IC}	917F37B5 C0E6D8D1

Наконец, производятся обмен указанными маркерными изображениями и их верификация.

T>C :	00 86 00 00 0C 7C 0A 85 08 B4 6D D9 BD 4D 98 38 1F 00
C>T :	7C 1B 86 08 91 7F 37 B5 C0 E6 D8 D1 87 0F 44 45 54 45 53 54 43 56 43 41 30 30 30 30 33

Команда				
CLA	00		Открытая	
INS	86		GENERAL AUTHENTICATE	
P1/P2	00 00		Ключи и протокол, известные в виде подразумеваемых	
Lc	0C		Длина данных	
Данные	Тег	Длина	Значение	Замечания
	7C	0A	-	Динамические аутентификационные данные
	85	08	B4 6D D9 BD 4D 98 38 1F	Аутентификационное маркерное изображение чипа
Le	00		Ожидаемая максимальная длина поля данных ответа составляет 256 байтов	
Ответ				
Данные	Тег	Длина	Значение	Замечания
	7C	0A		Динамические аутентификационные данные
	86	08	91 7F 37 B5 C0 E6 D8 D1	Аутентификационное маркерное изображение чипа
Байты состояния	90 00		Нормальная операция	

— — — — — — — —

Добавление H к части 11

ПРИМЕР С РЕШЕНИЯМИ: PACE – ИНТЕГРИРОВАННОЕ ОТОБРАЖЕНИЕ (ИНФОРМАЦИОННОЕ)

В настоящем добавлении приводятся два примера протокола PACE с интегрированным отображением. Первый основан на алгоритме Диффи-Хеллмана с использованием эллиптической кривой (ECDH), а второй – на алгоритме Диффи-Хеллмана (DH). Применяется ключ *K*, выработанный из MC3 в предыдущем примере.

H.1 ПРИМЕР НА ОСНОВЕ ECDH

Данный пример основан на применении эллиптической кривой стандарта BrainpoolP256r1. Используемым в этом примере блочным шифром является AES-128. В качестве напоминания ниже приводятся параметры кривой:

Простое p	A9FB57DB A1EEA9BC 3E660A90 9D838D72 6E3BF623 D5262028 2013481D 1F6E5377
Параметр a	7D5A0975 FC2C3057 EEF67530 417AFFE7 FB8055C1 26DC5C6C E94A4B44 F330B5D9
Параметр b	26DC5C6C E94A4B44 F330B5D9 BBD77CBF 95841629 5CF7E1CE 6BCCDC18 FF8C07B6
Координата x генератора группы G	8BD2AEB9 CB7E57CB 2C4B482F FC81B7AF B9DE27E1 E3BD23C2 3A4453BD 9ACE3262
Координата y генератора группы G	547EF835 C3DAC4FD 97F8461A 14611DC9 C2774513 2DED8E54 5C1D54C7 2F046997
Порядок группы n	A9FB57DB A1EEA9BC 3E660A90 9D838D71 8C397AA3 B561A6F7 901E0E82 974856A7
Сомножитель f	01

Ключом для шифрования является:

K_{π}	591468CD A83D6521 9CCCB856 0233600F

Зашифрованный одноразовый идентификатор

Чип выбирает случайным образом одноразовый идентификатор s и кодирует его с использованием K_{π}. После этого одноразовый идентификатор z посылается на терминал.

Дешифрованный одноразовый идентификатор s	2923BE84 E16CD6AE 529049F1 F1BBE9EB
Зашифрованный одноразовый идентификатор z	143DC40C 08C8E891 FBED7DED B92B64AD

Одноразовый идентификатор отображения

Случайным образом выбирается одноразовый идентификатор t и посылается в открытом формате. После этого t и s используются для вычислений интегрированного отображения. Вначале к идентификаторам s и t применяется псевдослучайная функция R_p, полученная из AES. Затем к результату применяется кодирование точки f_G для вычисления отображаемого генератора $\hat{G}=f_G(R_p(s,t))$.

Одноразовый идентификатор t	5DD4CBFC 96F5453B 130D890A 1CDBAE32
Псевдослучайная функция R(s,t)	E4447E2D FB3586BA C05DDB00 156B57FB B2179A39 49294C97 25418980 0C517BAA 8DA0FF39 7ED8C445 D3E421E4 FEB57322
$R_p(s,t)$	A2F8FF2D F50E52C6 599F386A DCB595D2 29F6A167 ADE2BE5F 2C3296AD D5B7430E
Координата x отображаемого генератора $\hat{G}$	8E82D315 59ED0FDE 92A4D049 8ADD3C23 BABA94FB 77691E31 E90AEA77 FB17D427
Координата y отображаемого генератора $\hat{G}$	4C1AE14B D0C3DBAC 0C871B7F 36081693 64437CA3 0AC243A0 89D3F266 C1E60FAD

Выполнение согласования ключей

Чип и терминал выполняют анонимное согласование ключей Диффи-Хеллмана, используя свои секретные ключи и отображаемый генератор $\hat{G}$. Совместно используемый секретный K является координатой x согласования.

Закрытый ключ SK_{IC} чипа	107CF586 96EF6155 053340FD 633392BA 81909DF7 B9706F22 6F32086C 7AFF974A
Открытый ключ PK_{IC} чипа	67F78E5F 7F768608 2B293E8D 087E0569 16D0F74B C01A5F89 57D0DE45 691E51E8 932B69A9 62B52A09 85AD2C0A 271EE6A1 3A8ADDDC D1A3A994 B9DED257 F4D22753

Закрытый ключ SK_{IFD} терминала	A73FB703 AC1436A1 8E0CFA5A BB3F7BEC 7A070E7A 6788486B EE230C4A 22762595
Открытый ключ PK_{IFD} терминала	89CBA23F FE96AA18 D824627C 3E934E54 A9FD0B87 A95D1471 DC1C0ABF DCD640D4 6755DE9B 7B778280 B6BEBD57 439ADFEB 0E21FD4E D6DF4257 8C13418A 59B34C37
Совместно используемый секретный K	4F150FDE 1D4F0E38 E95017B8 91BAE171 33A0DF45 B0D3E18B 60BA7BEA FDC2C713

Используя спецификации, указанные в [1], из K вырабатываются сеансовые ключи K_{Enc} и K_{MAC} путем применения хэш-функции SHA-1: K_{Enc}=SHA-1(K||0x00000001) и K_{MAC}=SHA-1(K||0x00000002). Затем используются только первые 16 октетов краткой формы сообщения с получением следующих результатов:

K_{Enc}	0D3FEB33 251A6370 893D62AE 8DAAF51B
K_{MAC}	B01E89E3 D9E8719E 586B50B4 A7506E0B

Взаимная аутентификация

Аутентификационные маркерные изображения вычисляются с помощью CMAC с применением ключа K_{MAC} к следующим вводным данным.

Вводные данные для T_{IC}	7F494F06 0A04007F 00070202 04040286 410489CB A23FFE96 AA18D824 627C3E93 4E54A9FD 0B87A95D 1471DC1C 0ABFDCD6 40D46755 DE9B7B77 8280B6BE BD57439A DFEB0E21 FD4ED6DF 42578C13 418A59B3 4C37
Вводные данные для T_{IFD}	7F494F06 0A04007F 00070202 04040286 410467F7 8E5F7F76 86082B29 3E8D087E 056916D0 F74BC01A 5F8957D0 DE45691E 51E8932B 69A962B5 2A0985AD 2C0A271E E6A13A8A DDDCD1A3 A994B9DE D257F4D2 2753

Соответствующими аутентификационными маркерными изображениями являются:

T_{IC}	75D4D96E 8D5B0308
T_{IFD}	450F02B8 6F6A0909

Н.2 ПРИМЕР НА ОСНОВЕ DH

Данный пример основан на применении 1024-битной группы MODP с 160-битной подгруппой с простым порядком. Используемым в этом примере блочным шифром является AES-128.

Параметры группы:

Простое p	B10B8F96 A080E01D DE92DE5E AE5D54EC 52C99FBC FB06A3C6 9A6A9DCA 52D23B61 6073E286 75A23D18 9838EF1E 2EE652C0 13ECB4AE A9061123 24975C3C D49B83BF ACCBDD7D 90C4BD70 98488E9C 219A7372 4EFFD6FA E5644738 FAA31A4F F55BCCC0 A151AF5F 0DC8B4BD 45BF37DF 365C1A65 E68CFDA7 6D4DA708 DF1FB2BC 2E4A4371
Генератор подгруппы g	A4D1CBD5 C3FD3412 6765A442 EFB99905 F8104DD2 58AC507F D6406CFF 14266D31 266FEA1E 5C41564B 777E690F 5504F213 160217B4 B01B886A 5E91547F 9E2749F4 D7FBD7D3 B9A92EE1 909D0D22 63F80A76 A6A24C08 7A091F53 1DBF0A01 69B6A28A D662A4D1 8E73AFA3 2D779D59 18D08BC8 858F4DCE F97C2A24 855E6EEB 22B3B2E5
Простой порядок q в рамках g	F518AA87 81A8DF27 8ABA4E7D 64B7CB9D 49462353

Используется следующий ключ шифрования:

K_{π}	591468CD A83D6521 9CCCB856 0233600F

Зашифрованный одноразовый идентификатор

Чип случайным образом выбирает одноразовый идентификатор s и кодирует его с использованием K_{π}. После этого одноразовый идентификатор z посылается на терминал.

Дешифрованный одноразовый идентификатор s	FA5B7E3E 49753A0D B9178B7B 9BD898C8
Зашифрованный одноразовый идентификатор z	9ABB8864 CA0FF155 1E620D1E F4E13510

Одноразовый идентификатор отображения

Случайным образом выбирается одноразовый идентификатор t и посылается в открытом формате. После этого t и s используются для вычисления интегрированного отображения. Вначале к идентификаторам s и t применяется псевдослучайная функция R_p, полученная из AES. Затем к результату применяется кодирование точки f_g.

Одноразовый идентификатор t	B3A6DB3C 870C3E99 245E0D1C 06B747DE
Псевдослучайная функция R(s,t)	EAB98D13 E0905295 2AA72990 7C3C9461 84DEA0FE 74AD2B3A F506F0A8 3018459C 38099CD1 F7FF4EA0 A078DB1F AC136550 5E3DC855 00EF95E2 0B4EEF2E 88489233 BEE0546B 472F994B 618D1687 02406791 DEEF3CB4 810932EC 278F3533 FDB860EB 4835C36F A4F1BF3F A0B828A7 18C96BDE 88FBA38A 3E6C35AA A1095925 1EB5FC71 0FC18725 8995944C 0F926E24 9373F485
$R_p(s,t)$	A0C7C50C 002061A5 1CC87D25 4EF38068 607417B6 EE1B3647 3CFB800D 2D2E5FA2 B6980F01 105D24FA B22ACD1B FA5C8A4C 093ECDFA FE6D7125 D42A843E 33860383 5CF19AFA FF75EFE2 1DC5F6AA 1F9AE46C 25087E73 68166FB0 8C1E4627 AFED7D93 570417B7 90FF7F74 7E57F432 B04E1236 819E0DFE F5B6E77C A4999925 328182D2
Отображаемый генератор $\hat{g}= f_g(R_p(s,t))$	1D7D767F 11E333BC D6DBAEF4 0E799E7A 926B9697 3550656F F3C83072 6D118D61 C276CDCC 61D475CF 03A98E0C 0E79CAEB A5BE2557 8BD4551D 0B109032 36F0B0F9 76852FA7 8EEA14EA 0ACA87D1 E91F688F E0DFF897 BBE35A47 2621D343 564B262F 34223AE8 FC59B664 BFEDFA2B FE7516CA 5510A6BB B633D517 EC25D4E0 BBAA16C2

Выполнение согласования ключей

Чип и терминал выполняют анонимное согласование ключей Диффи-Хеллмана, используя свои секретные ключи и отображаемый генератор $\hat{g}$.

Закрытый ключ SK_{IC} чипа	020F018C 7284B047 FA7721A3 37EFB7AC B1440BB3 0C5252BD 41C97C30 C994BB78 E9F0C5B3 2744D840 17D21FFA 6878396A 6469CA28 3EF5C000 DAF7D261 A39AB886 0ED4610A B5343390 897AAB5A 7787E4FA EFA0649C 6A94FDF8 2D991E8E 3FC332F5 142729E7 040A3F7D 5A4D3CD7 5CBEE1F0 43C1CAD2 DD484FEB 4ED22B59 7D36688E

Открытый ключ PK_{IC} чипа	928D9A0F 9DBA450F 13FC859C 6F290D1D 36E42431 138A4378 500BEB4E 0401854C FF111F71 CB6DC1D0 335807A1 1388CC8E AA87B079 07AAD9FB A6B169AF 6D8C26AF 8DDDC39A DC3AD2E3 FF882B84 D23E9768 E95A80E4 746FB07A 9767679F E92133B4 D379935C 771BD7FB ED6C7BB4 B1708B27 5EA75679 524CDC9C 6A91370C C662A2F3
Закрытый ключ SK_{IFD} терминала	4BD0E547 40F9A028 E6A515BF DAF96784 8C4F5F5F FF65AA09 15947FFD 1A0DF2FA 6981271B C905F355 1457B7E0 3AC3B806 6DE4AA40 6C1171FB 43DD939C 4BA16175 103BA3DE E16419AA 248118F9 0CC36A3D 6F4C3736 52E0C3CC E7F0F1D0 C5425B36 00F0F0D6 A67F004C 8BBA33F2 B4733C72 52445C1D FC4F1107 203F71D2 EFB28161
Открытый ключ PK_{IFD} терминала	0F0CC629 45A80292 51FB7EF3 C094E12E C68E4EF0 7F27CB9D 9CD04C5C 4250FAE0 E4F8A951 557E929A EB48E5C6 DD47F2F5 CD7C351A 9BD2CD72 2C07EDE1 66770F08 FFCB3702 62CF308D D7B07F2E 0DA9CAAA 1492344C 85290691 9538C98A 4BA4187E 76CE9D87 832386D3 19CE2E04 3C3343AE AE6EDBA1 A9894DC5 094D22F7 FE1351D5
Совместно используемый секретный K	419410D6 C0A17A4C 07C54872 CE1CBCEB 0A2705C1 A434C8A8 9A4CFE41 F1D78124 CA7EC52B DE7615E5 345E48AB 1ABB6E7D 1D59A57F 3174084D 3CA45703 97C1F622 28BDFDB2 DA191EA2 239E2C06 0DBE3BBC 23C2FCD0 AF12E0F9 E0B99FCF 91FF1959 011D5798 B2FCBC1F 14FCC24E 441F4C8F 9B08D977 E9498560 E63E7FFA B3134EA7

Используя хэш-функцию SHA-1, из K вырабатываются сеансовые ключи K_{Enc} и K_{MAC}: K_{Enc}= SHA-1(K||0x00000001) и K_{MAC}=SHA-1(K||0x00000002). Затем используются только первые 16 октетов краткой формы сообщения с получением следующих результатов:

K_{Enc}	01AFC10C F87BE36D 8179E873 70171F07
K_{MAC}	23F0FBD0 5FD6C7B8 B88F4C83 09669061

Взаимная аутентификация

Аутентификационное маркерное изображение вычисляется с помощью CMAC с применением ключа K_{MAC} к следующим вводным данным:

Вводные данные для T_{IC}	7F49818F 060A0400 7F000702 02040302 8481800F 0CC62945 A8029251 FB7EF3C0 94E12EC6 8E4EF07F 27CB9D9C D04C5C42 50FAE0E4 F8A95155 7E929AEB 48E5C6DD 47F2F5CD 7C351A9B D2CD722C 07EDE166 770F08FF CB370262 CF308DD7 B07F2E0D A9CAAA14 92344C85 29069195 38C98A4B A4187E76 CE9D8783 2386D319 CE2E043C 3343AEAE 6EDBA1A9 894DC509 4D22F7FE 1351D5
Вводные данные для T_{IFD}	7F49818F 060A0400 7F000702 02040302 84818092 8D9A0F9D BA450F13 FC859C6F 290D1D36 E4243113 8A437850 0BEB4E04 01854CFF 111F71CB 6DC1D033 5807A113 88CC8EAA 87B07907 AAD9FBA6 B169AF6D 8C26AF8D DDC39ADC 3AD2E3FF 882B84D2 3E9768E9 5A80E474 6FB07A97 67679FE9 2133B4D3 79935C77 1BD7FBED 6C7BB4B1 708B275E A7567952 4CDC9C6A 91370CC6 62A2F3

Соответствующими аутентификационными маркерными изображениями являются:

T_{IC}	C2F04230 187E1525
T_{IFD}	55D61977 CBF5307E

— — — — — — — —

Добавление I к части 11

ПРИМЕР С РЕШЕНИЯМИ: PACE – PACE С ОТОБРАЖЕНИЕМ ДЛЯ АУТЕНТИФИКАЦИИ ЧИПА (ИНФОРМАЦИОННОЕ)

В настоящем добавлении приводится пример протокола PACE с отображением для аутентификации чипа, основанный на алгоритме Диффи-Хеллмана с использованием эллиптической кривой (ECDH). Все цифры в таблицах записаны в шестнадцатеричной системе.

MC3 используется в качестве пароля. Соответствующие поля данных MC3, включая контрольные цифры, следующие:

- Номер документа: C11T002JM4;
- Дата рождения: 9608122;
- Дата истечения срока действия: 2310314.

Таким образом, кодировкой K зоны MC3 и выработанным ключом кодировки K_π являются:

K	894D03F1 48C6265E 89845B21 8856EA34 D00EF8E8
K_π	4E6F6FBF 7BE748B9 32C7B741 61BBA9DF

I.1 ПРИМЕР НА ОСНОВЕ ECDH

Этот пример основан на ECDH с применением стандартизированных по BrainpoolP256r1 параметров домена (см. [RFC 5639]).

В первом разделе приводится соответствующая информация PACE. Впоследствии перечисляются и рассматриваются обмениваемые блоки APDU, включая все генерированные одноразовые идентификаторы и эфемерные ключи.

Параметры эллиптической кривой

При использовании стандартизированных параметров домена вся информация, необходимая для осуществления PACE, содержится в структуре данных "Информация PACE". В частности, никакой информации о параметрах домена PACE не требуется.

Информация PACE	3012060A 04007F00 07020204 06020201 0202010D

Подробная структура информации PACE разбита по пунктам в следующей таблице.

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***	
30	12		ПОСЛЕДОВА-ТЕЛЬНОСТЬ	Информация PACE	
06	0A	04 00 7F 00 07 02 02 04 06 02	ИДЕНТИФИКА-ТОР ОБЪЕКТА		PACE с ECDH, отображение для аутентификации чипа и сеансовые ключи AES 128
02	01	02	ЦЕЛОЕ ЧИСЛО		Версия 2
02	01	0D	ЦЕЛОЕ ЧИСЛО		Стандартизированные параметры домена по Brainpool P256r1

Для удобства ниже приводится основанное на ASN.1 кодирование параметров домена по BrainpoolP256r1.

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***		
30	81 EC		ПОСЛЕДОВА-ТЕЛЬНОСТЬ	Параметры домена		
06	07	2A 86 48 CE 3D 02 01	ИДЕНТИФИКА-ТОР ОБЪЕКТА		Алгоритм id-ecPublicKey	
30	81 E0		ПОСЛЕДОВА-ТЕЛЬНОСТЬ		Параметры домена	
02	01	01	ЦЕЛОЕ ЧИСЛО			Версия
30	2C		ПОСЛЕДОВА-ТЕЛЬНОСТЬ			Исходное поле
06	07	2A 86 48 CE 3D 01 01	ИДЕНТИФИКА-ТОР ОБЪЕКТА			Простое поле
02	21	00 A9 FB 57 DB A1 EE A9 BC 3E 66 0A 90 9D 83 8D 72 6E 3B F6 23 D5 26 20 28 20 13 48 1D 1F 6E 53 77	ЦЕЛОЕ ЧИСЛО			Простое p
30	44		ПОСЛЕДОВА-ТЕЛЬНОСТЬ		Уравнение кривой	
04	20	7D 5A 09 75 FC 2C 30 57 EE F6 75 30 41 7A FF E7 FB 80 55 C1 26 DC 5C 6C E9 4A 4B 44 F3 30 B5 D9	ОКТЕТНАЯ СТРОКА			Параметр a
04	20	26 DC 5C 6C E9 4A 4B 44 F3 30 B5 D9 BB D7 7C BF 95 84 16 29 5C F7 E1 CE 6B CC DC 18 FF 8C 07 B6	ОКТЕТНАЯ СТРОКА			Параметр b

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***
04	41		ОКТЕТНАЯ СТРОКА	Генератор группы G
		04	-	Несжатая точка
		8B D2 AE B9 CB 7E 57 CB 2C 4B 48 2F FC 81 B7 AF B9 DE 27 E1 E3 BD 23 C2 3A 44 53 BD 9A CE 32 62	-	Координата x
		54 7E F8 35 C3 DA C4 FD 97 F8 46 1A 14 61 1D C9 C2 77 45 13 2D ED 8E 54 5C 1D 54 C7 2F 04 69 97	-	Координата y
02	21	00 A9 FB 57 DB A1 EE A9 BC 3E 66 0A 90 9D 83 8D 71 8C 39 7A A3 B5 61 A6 F7 90 1E 0E 82 97 48 56 A7	ЦЕЛОЕ ЧИСЛО	Порядок группы n
02	01	01	ЦЕЛОЕ ЧИСЛО	Сомножитель f

Блок-схема приложения применительно к примеру, основанному на ECDH

Для инициализации PACE терминал посылает на чип команду MSE:AT.

T>C :	`00 22 C1 A4 0F 80 0A 04 00 7F 00 07 02 02 04 06 02 83 01 01`
C>T :	`90 00`

Здесь T>C является сокращением для APDU, посланного с терминала на чип, в то время как C>T обозначает соответствующий ответ, посланный чипом на терминал. Шифрование команды поясняется в ниже-следующей таблице.

Команда				
CLA	00		Открытое	
INS	22		Управление средствами защиты	
P1/P2	C1 A4		Установка шаблона аутентификации для взаимной аутентификации	
Lc	0F		Длина поля данных	
Данные	Тег	Длина	Значение	Замечания
	80	0A	04 00 7F 00 07 02 02 04 06 02	Криптографический механизм: PACE с ECDH, отображение для аутентификации чипа и сеансовые ключи AES128

	83	01	01	Пароль: МСЗ
Ответ				
Байты состояния	90 00		Нормальная операция	

Зашифрованный одноразовый идентификатор

На следующем этапе чип случайным образом генерирует одноразовый идентификатор s и кодирует его с помощью K_π.

Дешифрованный одноразовый идентификатор s	`658B860B C94DF6F0 44FCE6D5 C82CF8E5`
Зашифрованный одноразовый идентификатор z	`CB60E8E0 D85B76A9 BD304747 C2AD42E2`

Зашифрованный одноразовый идентификатор запрашивается терминалом.

T>C :	`10 86 00 00 02 7C 00 00`
C>T :	`7C 12 80 10 CB 60 E8 E0 D8 5B 76 A9 BD 30 47 47 C2 AD 42 E2 90 00`

Кодировка APDU команды и соответствующий ответ указываются в следующей таблице.

Команда				
CLA	10		Составление последовательности команд	
INS	86		GENERAL AUTHENTICATE	
P1/P2	00 00		Ключи и протокол, известные в виде подразумеваемых	
Lc	02		Длина данных	
Данные	Тег	Длина	Значение	Замечания
	7C	00	-	Отсутствуют
Le	00		Ожидаемая максимальная длина поля данных ответа составляет 256 байтов	
Ответ				
Данные	Тег	Длина	Значение	Замечания
	7C	12		Динамические данные аутентификации

	80	10	CB60E8E0 D85B76A9 BD304747 C2AD42E2	Зашифрованный одноразовый идентификатор
Байты состояния	90 00		Нормальная операция	

Одноразовый идентификатор отображения

Этот одноразовый идентификатор отображается в генераторе эфемерной группы с помощью отображения общего типа. В нижеследующей таблице также собраны требуемые эфемерные ключи, выбираемые случайным образом.

Закрытый ключ терминала	5D8BB87B D74D985A 4B7D4325 B9F7B976 FE835122 77340079 8914AA22 738135CC
Открытый ключ терминала	7F1D410A DB7DDB3B 84BF1030 800981A9 105D7457 B4A3ADE0 02384F30 86C67EDE 1AB88910 4A27DB6D 842B0190 20FBF3CE ACB0DC62 7F7BDCAC 29969E19 D0E553C1
Закрытый ключ чипа	9E56A6B5 9C95D06E CE5CD10F 983BB2F4 F1943528 E577F238 81D89D8C 3BBEE0AA
Открытый ключ чипа	A234236A A9B9621E 8EFB73B5 245C0E09 D2576E52 77183C12 08BDD552 80CAE8B3 04F36571 3A356E65 A451E165 ECC9AC0A C46E3771 342C8FE5 AEDD0926 85338E23
Совместно используемый секретный H	2C1DCC17 73346492 C6636A36 EE4B965E 292E9AAE 7EE37736 EF58B9D0 A043F348 403A8CF3 3CA7DC0D 9DF61D08 89CE2442 4FF97C1A AD48A5CA 2A554B07 1EF7638D
Отображаемый генератор $\hat{G}$	89F0B5EA BF3BE293 C75903A3 98613192 5C9F5B51 5CA95AF4 85DC7E88 6F03245D 44BEFB2D D3A0DBD7 1CB5E618 971CF474 7F12B79E 548379A4 0E45963B AAF3E829

Для отображения одноразового идентификатора терминал и чип обмениваются следующими блоками APDU.

T>C :	10 86 00 00 45 7C 43 81 41 04 7F 1D 41 0A DB 7D DB 3B 84 BF 10 30 80 09 81 A9 10 5D 74 57 B4 A3 AD E0 02 38 4F 30 86 C6 7E DE 1A B8 89 10 4A 27 DB 6D 84 2B 01 90 20 FB F3 CE AC B0 DC 62 7F 7B DC AC 29 96 9E 19 D0 E5 53 C1 00
C>T :	7C 43 82 41 04 A2 34 23 6A A9 B9 62 1E 8E FB 73 B5 24 5C 0E 09 D2 57 6E 52 77 18 3C 12 08 BD D5 52 80 CA E8 B3 04 F3 65 71 3A 35 6E 65 A4 51 E1 65 EC C9 AC 0A C4 6E 37 71 34 2C 8F E5 AE DD 09 26 85 33 8E 23 90 00

Структуру блоков APDU можно описать следующим образом:

Команда					
CLA	10		Составление последовательности команд		
INS	86		GENERAL AUTHENTICATE		
P1/P2	00 00		Ключи и протокол, известные в виде подразумеваемых		
Lc	45		Длина данных		
Данные	Тег	Длина	Значение	Замечания	
	7C	43	-	Динамические аутентификационные данные	
	81	41		Данные отображения	
			04		Несжатая точка
			7F 1D 41 0A ... 86 C6 7E DE		Координата x
			1A B8 89 10... D0 E5 53 C1		Координата y
Le	00		Ожидаемая максимальная длина поля данных ответа составляет 256 байтов		
Ответ					
Данные	Тег	Длина	Значение	Замечания	
	7C	43		Динамические аутентификационные данные	
	82	41		Данные отображения	
			04		Несжатая точка
			A2 34 23 6A ... 80 CA E8 B3		Координата x
			04 F3 65 71... 85 33 8E 23		Координата y
Байты состояния	90 00		Нормальная операция		

Выполнение согласования ключей

На третьем этапе чип и терминал выполняют анонимное согласование ключей на основе ECDH, используя новые параметры домена, установленные генератором эфемерной группы на предыдущем этапе. В

качестве совместно используемого секретного параметра требуется только координата x, поскольку для выработки сеансовых ключей KDF использует только первую координату.

Закрытый ключ терминала	76ECFDAA 9841C323 A3F5FC5E 88B88DB3 EFF7E35E BF57A7E6 946CB630 006C2120
Открытый ключ терминала	446C9340 84D9DAB8 63944F21 9520076C 29EE3F7A E6722B11 FF319EC1 C7728F95 5483400B FF60BF0C 59292700 09277DC2 A515E125 75010AD9 BA916CF1 BF86FEFC
Закрытый ключ чипа	CD626EF3 C256E235 FE8912CA C28279E6 26008EDA 6B3A05C4 CF862A3B DAB79E78
Открытый ключ чипа	02AD566F 3C6EC7F9 324509AD 50A51FA5 2030782A 4968FCFE DF737DAE A9933331 11C3B9B4 C2287789 BD137E7F 8AA882E2 A3C633CC D6ECC2C6 3C57AD40 1A09C2E1
Совместно используемый секретный параметр	67950559 D0C06B4D 4B86972D 14460837 461087F8 419FDBC3 6AAF6CEA AC462832

Согласование ключей выполняется следующим образом:

T>C :	10 86 00 00 45 7C 43 83 41 04 44 6C 93 40 84 D9 DA B8 63 94 4F 21 95 20 07 6C 29 EE 3F 7A E6 72 2B 11 FF 31 9E C1 C7 72 8F 95 54 83 40 0B FF 60 BF 0C 59 29 27 00 09 27 7D C2 A5 15 E1 25 75 01 0A D9 BA 91 6C F1 BF 86 FE FC 00
C>T :	7C 43 84 41 04 02 AD 56 6F 3C 6E C7 F9 32 45 09 AD 50 A5 1F A5 20 30 78 2A 49 68 FC FE DF 73 7D AE A9 93 33 31 11 C3 B9 B4 C2 28 77 89 BD 13 7E 7F 8A A8 82 E2 A3 C6 33 CC D6 EC C2 C6 3C 57 AD 40 1A 09 C2 E1 90 00

Шифрование согласования ключей рассматривается в следующей таблице:

Команда				
CLA	10		Составление последовательности команд	
INS	86		GENERAL AUTHENTICATE	
P1/P2	00 00		Ключи и протокол, известные в виде подразумеваемых	
Lc	45		Длина данных	
Данные	Тег	Длина	Значение	Замечания
	7C	43	-	Динамические аутентификационные данные

	83	41		Эфемерный открытый ключ терминала	
			04		Несжатая точка
			44 6C 93 40 ... C7 72 8F 95		Координата x
			54 83 40 0B ... BF 86 FE FC		Координата y
Le	00		Ожидаемая максимальная длина поля данных ответа составляет 256 байтов		
Ответ					
Данные	Тег	Длина	Значение	Замечания	
	7C	43		Динамические аутентификационные данные	
	84	41		Эфемерный открытый ключ чипа	
			04		Несжатая точка
			02 AD 56 6F ... A9 93 33 31		Координата x
			11 C3 B9 B4 ... 1A 09 C2 E1		Координата y
Байты состояния	90 00		Нормальная операция		

С помощью KDF и совместно используемого секретного параметра вырабатываются сеансовые ключи KS_{Enc} и KS_{MAC} на основе AES 128. Таковыми являются:

KS_{Enc}	0A9DA4DB 03BDDE39 FC5202BC 44B2E89E
KS_{MAC}	4B1C0649 1ED5140C A2B537D3 44C6C0B1

Взаимная аутентификация

Аутентификационные маркерные изображения вырабатываются с помощью ключа KS_{MAC}, используя

Вводные данные для T_{IFD}	7F494F06 0A04007F 00070202 04060286 410402AD 566F3C6E C7F93245 09AD50A5 1FA52030 782A4968 FCFEDF73 7DAEA993 333111C3 B9B4C228 7789BD13 7E7F8AA8 82E2A3C6 33CCD6EC C2C63C57 AD401A09 C2E1

Вводные данные для T_{IC}	7F494F06 0A04007F 00070202 04060286 4104446C 934084D9 DAB86394 4F219520 076C29EE 3F7AE672 2B11FF31 9EC1C772 8F955483 400BFF60 BF0C5929 27000927 7DC2A515 E1257501 0AD9BA91 6CF1BF86 FEFC

в качестве вводных параметров. Шифрование вводных данных показано ниже.

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***
7F49	4F		ОТКРЫТЫЙ КЛЮЧ	Вводные данные для T_{IFD}
06	0A	04 00 7F 00 07 02 02 04 06 02	ИДЕНТИФИКАТОР ОБЪЕКТА	PACE с ECDH, отображение для аутентификации чипа и сеансовые ключи AES 128
86	41		ТОЧКА ЭЛЛИПТИЧЕСКОЙ КРИВОЙ	Эфемерная открытая точка чипа
		04		Несжатая точка
		02 AD 56 6F... A9 93 33 31		Координата x
		11 C3 B9 B4 ... 1A 09 C2 E1		Координата y

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***
7F49	4F		ОТКРЫТЫЙ КЛЮЧ	Вводные данные для T_{IC}
06	0A	04 00 7F 00 07 02 02 04 06 02	ИДЕНТИФИКАТОР ОБЪЕКТА	PACE с ECDH, отображение для аутентификации чипа и сеансовые ключи AES 128
86	41		ТОЧКА ЭЛЛИПТИЧЕСКОЙ КРИВОЙ	Эфемерная открытая точка терминала
		04		Несжатая точка
		44 6C 93 40 ... C7 72 8F 95		Координата x
		54 83 40 0B ... BF 86 FE FC		Координата y

Вычисленными аутентификационными маркерными изображениями являются:

T_{IFD}	E86BD060 18A1CD3B
T_{IC}	8596CF05 5C67C1A3

Наконец, производятся обмен этими маркерными изображениями и их верификация.

T>C :	00 86 00 00 0C 7C 0A 85 08 E8 6B D0 60 18 A1 CD 3B 00
C>T :	7C 3C 86 08 85 96 CF 05 5C 67 C1 A3 8A 30 1E EA 96 4D AA E3 72 AC 99 0E 3E FD E6 33 33 53 BF C8 9A 67 04 D9 3D A8 79 8C F7 7F 5B 7A 54 BD 10 CB A3 72 B4 2B E0 B9 B5 F2 8A A8 DE 2F 4F 92 90 00

Кодирование взаимной аутентификации рассматривается в следующей таблице:

Команда				
CLA	00		Составление последовательности команд отсутствует (последняя команда в последовательности)	
INS	86		GENERAL AUTHENTICATE	
P1/P2	00 00		Ключи и протокол, известные в виде подразумеваемых	
Lc	0C		Длина данных	
Данные	Тег	Длина	Значение	Замечания
	7C	0A	-	Динамические аутентификационные данные
	85	08		Аутентификационное маркерное изображение терминала
			E8 6B D0 60 18 A1 CD 3B	T_{IFD}
Le	00		Ожидаемая максимальная длина поля данных ответа составляет 256 байтов	
Ответ				
Данные	Тег	Длина	Значение	Замечания
	7C	3C		Динамические данные аутентификации
	86	08		Аутентификационное маркерное изображение чипа
			85 96 CF 05 5C 67 C1 A3	T_{IC}

	8A	30			Координата x
			1E EA 96 4D … DE 2F 4F 92		*Зашифрованные данные аутентификации чипа*
Байты состояния	90 00		Нормальная операция		

Аутентификация чипа

Получение ChipAuthenticationPublicKeyInfo из файла EF.CardSecurity

ChipAuthenticationPublicKeyInfo	30620609 04007F00 07020201 02305230 0C060704 007F0007 01020201 0D034200 04187270 9494399E 7470A643 1BE25E83 EEE24FEA 568C2ED2 8DB48E05 DB3A610D C884D256 A40E35EF CB59BF67 53D3A489 D28C7A4D 973C2DA1 38A6E7A4 A08F68E1 6F02010D

Подробная структура ChipAuthenticationPublicKeyInfo разбита по пунктам в следующей таблице.

Тег	***Длина***	***Значение***	***Тип ASN.1***	***Замечания***	
30	62		ПОСЛЕДОВА-ТЕЛЬНОСТЬ	ChipAuthenticationPublicKeyInfo	
06	09	04 00 7F 00 07 02 02 01 02	ИДЕНТИФИКА-ТОР ОБЪЕКТА		id-PK-ECDH
30	52		ПОСЛЕДОВА-ТЕЛЬНОСТЬ		SubjectPublicKeyInfo
30	0C		ПОСЛЕДОВА-ТЕЛЬНОСТЬ		Стандартизированные параметры домена по Brainpool P256r1
06	07	04 00 7F 00 07 01 02	ИДЕНТИФИКА-ТОР ОБЪЕКТА		standardizedDomainParameters
02	01	0D	ЦЕЛОЕ ЧИСЛО		Brainpool256r1
03	42	00 04 18 72 70 … 8F 68 E1 6F	СТРОКА БИТОВ		Открытый ключ CA
02	01	0D	ЦЕЛОЕ ЧИСЛО		keyID 13

Для аутентификации чипа используются следующие данные:

Зашифрованные данные аутентификации чипа	1EEA964D AAE372AC 990E3EFD E6333353 BFC89A67 04D93DA8 798CF77F 5B7A54BD 10CBA372 B42BE0B9 B5F28AA8 DE2F4F92
Дешифрованные данные аутентификации чипа	85DC3FA9 3D0952BF A82F5FD1 89EE75BD 82F11D1F 0B8ED4BF 5319AC9B 53C426B3
Вектор инициализации IV для де-/шифрования данных CA IV = $E(KS_{ENC}, -1)$	F6A3B75A1 E933941 DD7A13E2 520779DF
Открытый ключ чипа, полученный из одноразового идентификатора отображения с использованием GENERAL AUTHENTICATE $PK_{MAP,IC}$	A234236A A9B9621E 8EFB73B5 245C0E09 D2576E52 77183C12 08BDD552 80CAE8B3 04F36571 3A356E65 A451E165 ECC9AC0A C46E3771 342C8FE5 AEDD0926 85338E23
Открытый ключ чипа CA, полученный из ChipAuthenticationPublicKeyInfo PK_{IC}	18727094 94399E74 70A6431B E25E83EE E24FEA56 8C2ED28D B48E05DB 3A610DC8 84D256A4 0E35EFCB 59BF6753 D3A489D2 8C7A4D97 3C2DA138 A6E7A4A0 8F68E16F

Терминал верифицирует, что $PK_{MAP,IC} = KA(CA_{IC}, PK_{IC}, D_{IC})$.

— — — — — — — —

Добавление J к части 11

ПРОЦЕДУРЫ ПРОВЕРКИ (ИНФОРМАЦИОННЫЕ)

J.1 ПРОЦЕДУРА ПРОВЕРКИ ПРИЛОЖЕНИЯ ЭЛЕКТРОННОГО МСПД

В настоящем разделе приводится описание процедуры проверки, охватывающей только приложение электронного МСПД ("LDS1-документы").

1. Получить доступ к бесконтактной ИС (см. раздел 4.2)

   - Если доступ к ИС защищен, то на этом этапе можно использовать PACE или BAC, хотя по соображениям безопасности рекомендуется использовать PACE. Начиная с 1 января 2018 года, электронные МСПД могут поддерживать только PACE.

   - Если поддерживается ИС и терминалом, то по соображениям технических характеристик следует использовать PACE-CAM.

   - ИС предоставляет доступ к менее конфиденциальным данным в приложении электронного МСПД и к файлу EF.CardSecurity в мастер-файле, если присутствует.

2. Начать аутентификацию данных

   - Считать объект защиты документа и верифицировать подпись, включая верификацию последовательности сертификата лица, подписавшего документ.

3. Аутентификация чипа

   - В зависимости от поддержки, обеспечиваемой ИС, выполнить аутентификацию чипа или активную аутентификацию. О поддержке активной аутентификации свидетельствует наличие файла EF.DG15 в приложении электронного МСПД, а о поддержке аутентификации чипа – наличие соответствующего поля `SecurityInfos` в файле EF.DG14.

   - Этот этап можно также выполнить в рамках этапа 1, если используется PACE с отображением для аутентификации чипа.

   - Аутентификация завершается только совместно с аутентификацией файла, содержащего открытый ключ (файлы EF.CardSecurity, EF.DG14 или EF.DG15), использовавшегося для реализации этого этапа.

4. Дополнительный контроль доступа

   - Выполнение аутентификации терминала необходимо в том случае, когда конфигурация электронного МСПД требует этого для обеспечения доступа к конфиденциальным данным, т. е. к файлам EF.DG3 и/или EF.DG4.

5. Считывание данных

   - Считывание данных может быть начато сразу же после предоставления прав доступа, например менее конфиденциальные данные могут быть считаны после реализации этапа 1.

   - Без аутентификации считанных данных (этап 2) эти данные не должны рассматриваться в качестве подлинных.

J.2 ПРОЦЕДУРА ПРОВЕРКИ ЭЛЕКТРОННЫХ МСПД С НЕСКОЛЬКИМИ ПРИЛОЖЕНИЯМИ

В настоящем разделе приводится описание процедуры проверки, предназначенной для электронных МСПД, содержащих одно или несколько приложений, помимо приложения электронного МСПД ("LDS2-документы"). Эта процедура может также использоваться для обеспечения доступа только к приложению электронного МСПД.

1. Получить доступ к бесконтактной ИС (см. раздел 4.2)

   - В этом случае для получения доступа к ИС имеется только PACE.

   - Если поддерживается ИС и терминалом, то по соображениям технических характеристик следует использовать PACE-CAM.

   - ИС предоставляет доступ к менее конфиденциальным данным в приложении электронного МСПД и к файлу EF.CardSecurity, содержащемуся в мастер-файле.

2. Проверить наличие файла EF.CardSecurity

   - Если файл EF.CardSecurity отсутствует, то электронный МСПД не поддерживает аутентификацию в мастер-файле (подразумевая при этом, что в ИС содержится только приложение электронного МСПД). В этом случае выбрать приложение электронного МСПД и продолжить выполнение этапа 2 процедуры, изложенной в разделе J.1 настоящего добавления.

3. Приступить к аутентификации данных

   - Считать файл EF.CardSecurity и проверить подпись, включая последовательность верификации сертификата лица, подписавшего документ.

   - Данные приложения электронного МСПД защищаются объектом обеспечения защиты документа, который должен верифицироваться при считывании данных приложения. Данные других приложений защищаются подписями данных, которые также должны верифицироваться при считывании этих данных.

4. Аутентификация чипа

   - Выполнить аутентификацию чипа в мастер-файле. Если в поле `SecurityInfos` файла EF.CardSecurity необходимая информация отсутствует, то в мастер-файле ИС аутентификацию не поддерживает. В этом случае выбрать приложение электронного МСПД и продолжить реализацию этапа 2 процедуры, предусмотренной разделом J.1 настоящего добавления.

- Если используется PACE с отображением для аутентификации чипа, то этот этап можно также выполнить в рамках этапа 1.

- Аутентификация завершается только совместно с аутентификацией файла, содержащего открытый ключ (файл EF.CardSecurity), используемого на этом этапе.

5. Дополнительный контроль доступа

   - Выполнить аутентификацию терминала.

   - Если в приложении электронного МСПД требуется доступ только к считыванию менее конфиденциальных данных, то этот этап можно пропустить.

6. Считывание/запись данных

   - Считывание/запись данных предусматривают выбор предложений, содержащих файлы.

   - Считывание данных можно начать сразу же после получения необходимых прав доступа, например, менее конфиденциальные данные приложения МСПД могут быть считаны после реализации этапа 1.

   - Без аутентификации считаных данных (этап 3) данные не должны рассматриваться в качестве подлинных.

— — — — — — — —

Добавление K к части 11

ЕВРОПЕЙСКИЙ РАСШИРЕННЫЙ КОНТРОЛЬ ДОСТУПА (ИНФОРМАЦИОННОЕ)

Аутентификация терминала, определенная в настоящем документе, основана на расширенном контроле доступа, используемом в Европейском союзе (см. документ [TR-03110]) для защиты доступа к отпечаткам пальцев, хранимым в приложении LDS1. В настоящем добавлении обращается внимание на различие между положениями документа [TR-03110] и протоколами, определенными в настоящем документе.

Усовершенствованная процедура проверки, используемая для обеспечения доступа к электронным МСПД, изготовленным в соответствии с требованиями документа [TR-03110], предусматривает реализацию следующих этапов:

1. Выполнить процедуру оценки чипа (см. раздел 4.2) и выбрать приложение электронного МСПД;

2. Выполнить аутентификацию чипа в приложении электронного МСПД (см. раздел 6.2) и приступить к проведению пассивной аутентификации (см. раздел 5.1);

3. Выполнить аутентификацию терминала (см. ниже) в приложении электронного МСПД (см. раздел 7.1).

Примечание. В рамках европейского расширенного контроля доступа выполняются как аутентификация чипа, так и аутентификация терминала. Спецификации, содержащиеся в настоящем документе, в зависимости от контекста, позволяют выполнять эти протоколы либо в приложении электронного МСПД, либо в мастер-файле.

K.1 ПРАВА ДОСТУПА

Таблица K-1. Авторизация систем проверки

7	6	5	4	3	2	1	0	Описание
x	x	-	-	-	-	-	-	Роль (см. часть 12 документа Doc 9303)
-	-	x	x	x	x	x	x	Права доступа
-	-	x	x	x	x	-	-	RFU
-	-	-	-	-	-	1	-	Доступ к считыванию приложения электронного МСПД: DG4 (радужная оболочка глаза)
-	-	-	-	-	-	-	1	Доступ к считыванию приложения электронного МСПД: DG3 (отпечаток пальца)

Права доступа к группам данных в приложениях, не являющихся приложением электронного МСПД, передаются посредством расширений, как определено в частях 12 и 10 документа Doc 9303. Права доступа к отпечаткам пальцев (и радужной оболочке глаза) передаются посредством шаблона авторизации владельца сертификата:

Информация, касающаяся определения эффективных прав доступа, приводится в разделе 7.1.4.3.6.

K.2 ФАЙЛ EF.CVCA

Согласно спецификации объекты доверия (справочные данные сертифицирующего полномочного органа), известные для ИС для проведения верификации терминала, передаются IFD в качестве составной части протокола PACE (см. раздел 4.4.3.5).

Вместо этого в рамках европейского расширенного контроля доступа определяется транспарентный файл EF.CVCA. Ниже приводится спецификация:

Таблица K-2. Элементарный файл EF.CVCA

Имя файла	EF.CVCA
ID файла	0x011C (по умолчанию)
Краткий ID файла	0x1C (по умолчанию)
Доступ для считывания	PACE
Доступ для записи	НИКОГДА (только внутреннее обновление)
Размер	36 байтов (фиксированное значение), заполняемых октетами со значением 0x00
Контент	[CARi][||CARi-1][||0x00..00]

Если ИС поддерживает аутентификацию терминала в приложении электронного МСПД, она ДОЛЖНА делать ссылки на открытые ключи CVCA, приемлемые для систем проверки, которые имеются в транспарентном элементарном файле EF.CVCA в приложении электронного МСПД, как указано в таблице K-2.

Этот файл СОДЕРЖИТ последовательность объектов данных ссылки на сертифицирующий полномочный орган (CAR) (см. часть 12 документа Doc 9303), приемлемую для аутентификации терминала.

- Он СОДЕРЖИТ максимум два объекта данных ссылки на сертифицирующий полномочный орган.
- В этом перечне первым объектом данных ЯВЛЯЕТСЯ ссылка на сертифицирующих полномочный орган.
- Данный файл ДОЛЖЕН заполняться октетами со значением 0x00.

Файл EF.CVCA имеет идентификатор EF по умолчанию и краткий идентификатор EF. Если значения по умолчанию использовать нельзя, то (краткий) идентификатор EF УКАЗЫВАЕТСЯ в ФАКУЛЬТАТИВНОМ параметре `efCVCA` поля `TerminalAuthenticationInfo`. Если параметр `efCVCA` используется для указания подлежащего применению идентификатора EF, то идентификатор EF по умолчанию аннулируется. Если краткий идентификатор EF указывается в `efCVCA`, то файл EF.CVCA ДОЛЖЕН однозначно выбираться с использованием заданного идентификатора EF.

```
TerminalAuthenticationInfo ::= SEQUENCE {
  protocol OBJECT IDENTIFIER(id-TA),
  version  INTEGER, -- MUST be 1
  efCVCA   FileID OPTIONAL
}

FileID ::= SEQUENCE {
  fid  OCTET STRING (SIZE(2)),
  sfid OCTET STRING (SIZE(1)) OPTIONAL
}
```

— КОНЕЦ —

ISBN 978-92-9275-568-3