INTERNATIONAL STANDARD

ISO/IEC 25185-1

First edition
2016-01-15

Identification cards — Integrated circuit card authentication protocols —

Part 1: Protocol for Lightweight Authentication of Identity

Cartes d'identification — Integrated circuit protocoles d'authentification par carte —

Partie 1: Protocole pour l'authentification de l'identité léger

Reference number
ISO/IEC 25185-1:2016(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT), see the following URL: Foreword — Supplementary information.

The committee responsible for this document is ISO/IEC JTC 1, *Information technology*.

ISO/IEC 25185-1 was prepared by Standards Australia under the JTC1 Fast Track process from the existing AS-5185 Australian standard as a submission to ISO/IEC JTC 1, *Information technology*.

ISO/IEC 25185 consists of the following parts, under the general title *Identification cards — Integrated circuit card authentication protocols*:

— *Part 1: Protocol for Lightweight Authentication of Identity*

Introduction

PLAID (Protocol for Lightweight Authentication of IDentity) is an ICC (smartcard) authentication protocol, which is designed to expressly support contactless applications. The protocol is designed to fill the gap in standardized protocols between tag and RFID based technologies which do not utilize cryptography but are fast, and PKI based authentication, which can be very strong cryptographically, but slower, and unsuitable for many contactless use-cases.

It is based on a cryptographic method, which uses both symmetric and asymmetric cryptography in a hybrid protocol to protect the communications between ICCs and terminal devices. This is done in such a way that strong authentication of the ICC and credentials is possible in a fast, highly secure and private fashion without the exposure of card or cardholder identifying information or any other information which is useful to an attacker.

PLAID uses standards-based cryptography commonly available on ICCs, computer systems and embedded devices and is consequently highly portable to a wide range of ICC cards and IFD devices.

ISO/IEC draws attention to the fact that it is claimed that compliance with this International Standard may involve the use of intellectual property concerning PLAID.

ISO/IEC takes no position concerning the evidence, validity and scope of such an intellectual property right.

The holder of the right has assured ISO/IEC that they are willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect the licence provided is perpetual, irrevocable, world-wide, non-exclusive, royalty free and no-charge. The statement of the holder of this intellectual property right is registered with ISO/IEC. Information may be obtained from:

The Commonwealth of Australia, acting through the Commonwealth Services Delivery Agency, also known as “Human Services” or such other agency as may, from time to time, administer the PLAID Licence on behalf of the Commonwealth of Australia.

Address: Attn: PLAID; Human Services; PO Box 7788, Canberra M.C. ACT 2910, Australia

Email: PLAID@humanservices.gov.au

Licence: https://www.plaid.gov.au

ISO/IEC wishes to thank the Australian Commonwealth for their support of the development of PLAID and the provision of the associated intellectual property in a royalty free and no-charge licence.

Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights other than those identified above. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

This is the first ISO/IEC edition, the previous Australian Standard, AS 5185:2010 is technically identical other than for references where ISO/IEC standards are required to differ due to ISO convention including compliance to nominated normative standards and where cipher strengths have been updated.

Identification cards — Integrated circuit card authentication protocols —

Part 1: Protocol for Lightweight Authentication of Identity

1 Scope

This International Standard provides an authentication protocol suitable for use in physical and logical access control systems based on ICCs and related systems which support standards based AES-128 and RSA-2048 ciphers and the SHA-256 hashing algorithm.

The standard specifies PLAID and its implementation in sufficient detail to allow any two or more implementations to be interoperable.

This International Standard does not address how implementations share cryptographic keys, access control system credential records (including revocation) or manage payload entities such as PIN, PINHash, or biometric templates or other payload objects.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 7816-4, *Identification cards — Integrated circuit cards — Part 4: Organization, security and commands for interchange*

ISO/IEC 7816-5, *Identification cards — Integrated circuit cards — Part 5: Registration of application providers*

ISO/IEC 8824-1, *Information technology — Abstract Syntax Notation One (ASN.1) — Part 1: Specification of basic notation*

ISO/IEC 9797-1, *Information technology — Security techniques — Message Authentication Codes (MACs) — Part 1: Mechanisms using a block cipher*

ISO/IEC 10116, *Information technology — Security techniques — Modes of operation for an n-bit block cipher*

ISO/IEC 10118-3, *Information technology — Security techniques — Hash-functions — Part 3: Dedicated hash-functions*

ISO/IEC 18033-2, *Information technology — Security techniques — Encryption algorithms — Part 2: Asymmetric ciphers*

ISO/IEC 18033-3, *Information technology — Security techniques — Encryption algorithms — Part 3: Block ciphers*

IETF RFC 3447, *Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1*

3 Terms and definitions

For the purposes of this document, the terms and definitions in ISO/IEC 7816-4, ISO/IEC 7816-5, ISO/IEC 8824-1, ISO/IEC 10116, ISO/IEC 10118-3, ISO/IEC 18033-2, ISO/IEC 18033-3, IETF RFC 3447, and the following apply.

4 Symbols and abbreviated terms

For the purposes of this document the following symbols and abbreviated terms apply.

||	Logical concatenation of bit strings
ACS	Access Control System
AES	Advanced Encryption Standard (as defined in ISO/IEC 18033-3)
$AES_{Decrypt}^{Key}$	Perform AES Decryption using Key
$AES_{Encrypt}^{Key}$	Perform AES Encryption using Key
AID	Application IDentifier (as defined in ISO/IEC 7816-4)
AP	Authentication Protocol (as defined in ISO/IEC 24727-3)
APDU	Application Protocol Data Unit (as defined is ISO/IEC 7816-4)
ASN.1	Abstract Syntax Notation Number 1 (as defined in ISO/IEC 8824-1)
BER	Basic Encoding Rules of ASN.1 (as defined in ISO/IEC 8825-1)
CBC	Cipher Block Chaining (as defined in ISO/IEC 10116)
CLA	Class Byte (as defined in ISO/IEC 7816-4)
CRT	Chinese Remainder Theorem
DivData	Diversification Data — Seed data used in cryptographic operations
^{e}STR	Encrypted version of data object (STR in this case)
FA	Final Authenticate
IA	Initial Authenticate
ICC	Integrated Circuit Card, logically equivalent in this International Standard to PICC
IFD	InterFace Device
INS	Instruction Byte (as defined in ISO/IEC 7816-4)
IV	Initialisation Vector
$Key^{(DIV)}$	Diversified version of key
KeySetID	A 2 byte value specifying which keyset the protocol will negotiate or use
LACS	Logical Access Control System
OpModeID	A 2 byte value specifying which operational mode the protocol will use
PACS	Physical Access Control System
PICC	Proximity Integrated Circuit Card, logically equivalent to ICC in this International Standard

PIN	Personal Identification Number
PKI	Public-Key Infrastructure
PKCS1.5	RSA padding method (as defined in IETF RFC 3447)
PLAID	Protocol for Lightweight Authentication of IDentity
RNG	Random Number Generator
RSA	Asymmetric cryptographic cipher (as defined in ISO/IEC 18033-2)
$RSA_{Decrypt}^{Key}$	Perform RSA Decryption using Key
$RSA_{Encrypt}^{Key}$	Perform RSA Encryption using Key
SHA	Secure Hash Algorithm (as defined in ISO/IEC 10118-3)
SW1-SW2	Status Bytes (as defined in ISO/IEC 7816-4)
TLV	Tag, Length, Value
UID	Unique IDentifier
UUID	Open credential numbering system (as defined in IETF RFC 4122)
Wiegand	PACS credential numbering system based on Wiegand effect card readers from the 1980s

5 Data dictionary

Table 1 defines the size and details of PLAID data objects.

Table 1 — Data dictionary

Object name	Purpose	Size	Data type	Comments
ACSRecord	Access control system record	Variable. NOTE ACSRecord plus Payload should not exceed 64 bits unless a secondary transmission error check such as CMAC is implemented.	Open	An access control system record for each supported Operational Mode Identifier for the purpose of authorization and revocation by back office PACS or LACS access control systems. This record is mapped by the OpModeID to the particular back office numbering system the protocol is supporting. This record is returned by the final authenticate command response.

Table 1 *(continued)*

Object name	Purpose	Size	Data type	Comments
DivData	Symmetric key diversification data	128 bits	Binary	A seed value which is set during PLAID enablement for use in the key diversification algorithm to ensure that loss of an individual ICC symmetric key cannot result in a breach of the system master keys. This seed is determined by the owner of the key and should preferably be random or unique per PLAID ICC issued AND per system.
FAKey	Undiversified final authenticate key (AES-128)	128 bits	Binary	The undiversified AES master key.
$FAKey^{(DIV)}$	Diversified final authenticate key (AES-128)	128 bits	Binary	The key derived from FAKey by the specified diversification process.
IV	Initialisation Vector	16 bytes	Binary	The IV shall be set to all zero bits.
IAKey	Initial authenticate key (RSA-2048)	2 048 bits	Binary	An instance of an initial authenticate RSA key pair.
KeySetID	Uniquely identifies a keyset that consists of an $FAKey^{(div)}$ and IAKey pair	2 bytes	Binary	One or more two byte identifiers sent in a list to the ICC in the initial authenticate command so as to determine and/or negotiate the keyset to be used for authentication.
OpModeID	Operating mode identifier	2 bytes	Binary	An identifier sent to the ICC in the final authenticate command that determines which ACSRecord and Payload is passed in the final authentication response from the ICC.
Payload	Transfer of extensible user related data	Variable. NOTE In the instance of the Final Authenticate Response, the ACSRecord plus Payload should not exceed 64 bits unless a secondary transmission error check such as CMAC is implemented.	Open	Container for carriage of optional data. Payload is always linked to the OpModeID value.
RND1	Random number one	16 bytes	Binary	Random number generated by the ICC using its RNG.
RND2	Random number two	16 bytes	Binary	Random number generated by the IFD or back office system using a RNG.
STR1	First APDU body object	50 bytes	Binary	Internal to ICC only.

Table 1 *(continued)*

Object name	Purpose	Size	Data type	Comments
^{e}STR1	RSA-2048 Encrypted version of First APDU string	256 bytes	RSA-2048 encrypted binary	Transmitted RSA is encrypted.
STR(2,3)	2nd and 3rd APDU body objects	Variable	Binary	Internal to ICC/IFD only.
^{e}STR(2,3)	AES encrypted version of 2nd and 3rd APDU body objects	Variable	AES encrypted binary	Transmitted AES is encrypted.
KeysHash	Key Hash	128 bits	Binary	String generated by the IFD and ICC separately calculating SHA-256[RND1|RND2]. Additional trailing output bits shall be truncated to the required length.
SessionKey	Session key	128 bits	Binary	Key generated by the IFD and ICC separately calculating KeysHash.
ShillKey	Shill key (RSA/AES)	2 048/128 bits respectively	Binary	A Shill key is randomly generated by the ICC and is only known to the ICC application. A shill key is generated for both the initial authenticate (RSA) and the final authenticate (AES) commands. Shill key is used by the ICC in place of the actual key when an error in the response is detected, thereby removing any indication to a potential attacker that an error has been detected.

NOTE The stream byte order of all data objects is big-endian.

6 Authentication Protocol Description

This Clause discusses the steps involved in the PLAID mutual authentication involving a Physical or Logical access control use-case. Figure 1 illustrates this process.

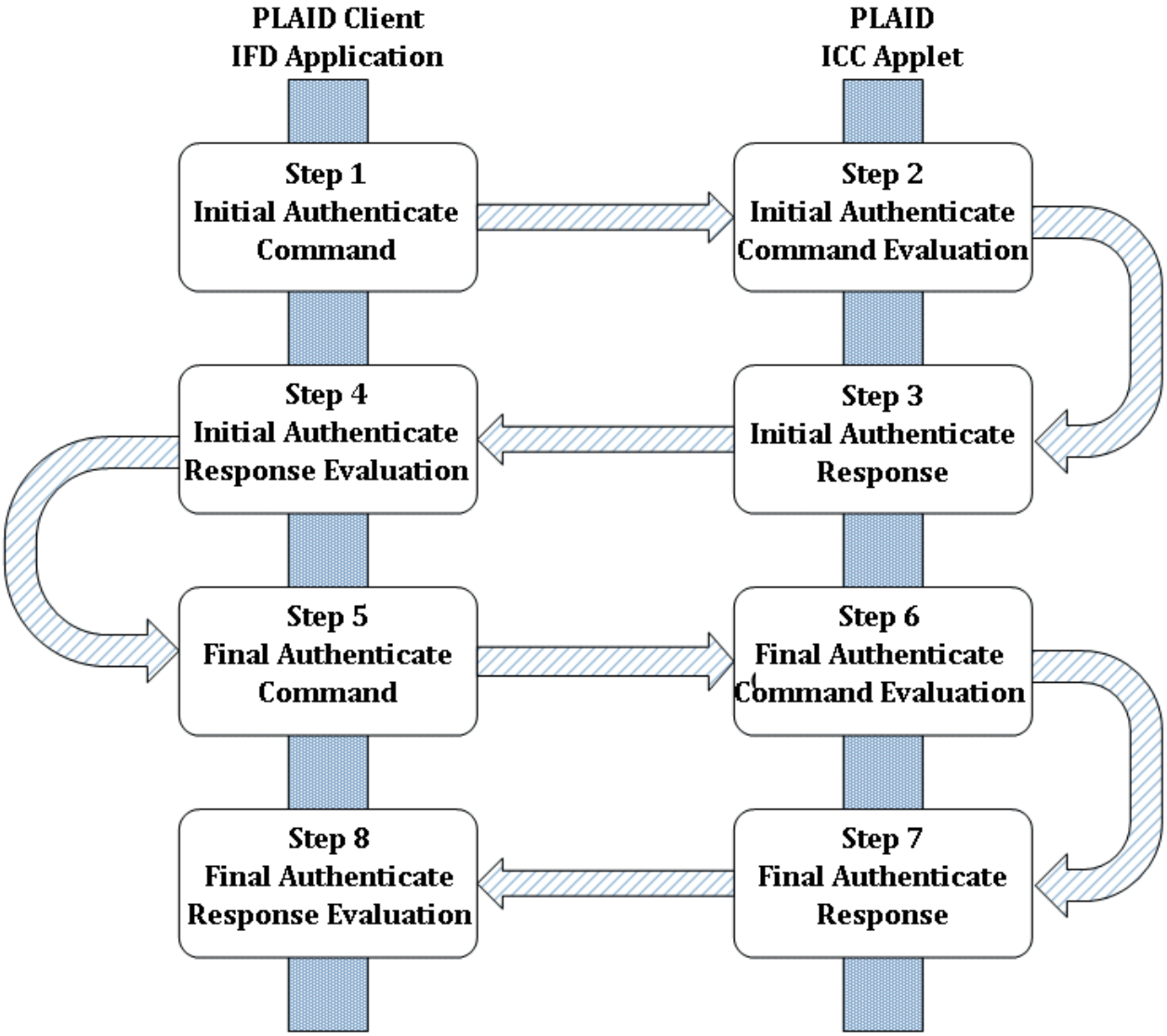

Figure 1 — PLAID Authentication Protocol Overview

The steps required to carry out a mutual authenticate using PLAID shall be as follows:

6.1 Step 1 — INITIAL AUTHENTICATE command

a) The IFD sends an Initial Authenticate APDU request to the ICC in order to obtain the Diversification Data (DivData).

b) The body of the APDU contains the complete list of authorized KeySetID values (BER-TLV encoded) that will be acknowledged by the IFD.

c) This list shall be ordered with the preferred KeySetID first, followed by lesser preferred KeySetID values.

d) The formal ASN.1 representation of the body of the Initial Authenticate APDU is:

```
PLAID {iso(1) standard(1) iccap(25185) part1(1) plaid(1)}
DEFINITIONS IMPLICIT TAGS ::=
BEGIN
KeySetID ::= OCTET STRING (SIZE (2))
KeySetIDSequence ::= SEQUENCE OF
KeySetID
END
```

NOTE 1 BER-TLV encoding is only used in the body of the Initial Authenticate command.

NOTE 2 The KeySetID values are passed in the clear by the IFD and should not be considered to be secured.

6.2 Step 2 — INITIAL AUTHENTICATE command evaluation

a) The ICC parses the BER-TLV listing of KeySetID values and retrieves the first IAKey found which matches a KeySetID supported by the ICC.

b) The ICC should traverse the entire list of KeySetID values irrespective of when a keyset match is found. This serves to prevent potential timing attacks.

c) If none of the KeySetID values identified match a key stored by the ICC then the ICC responds as per step 3 using a random byte string encrypted with the ShillKey, thereby preventing any indication that an error has occurred.

6.3 Step 3 — INITIAL AUTHENTICATE response

a) The ICC generates a random value (RND1) using its RNG. The size of RND1 is identical to the key size of the selected AES-128 cipher (16 bytes).

b) The ICC retrieves the unique diversification data DivData.

c) The ICC creates the bit string STR1: KeySetID || DivData || RND1 || RND1.

d) The ICC computes the bit string ^{e}STR1 where ^{e}STR1 = $RSA_{Encrypt}{}^{IAKey}$ (STR1). This encryption only uses the modulus and public exponent of the IAKey. PKCS1.5 padding shall be incorporated in the encryption.

e) The ICC transmits the string ^{e}STR1 to the IFD.

6.4 Step 4 — INITIAL AUTHENTICATE response evaluation

a) The IFD receives string ^{e}STR1 and calculates STR1 where STR1 = $RSA_{Decrypt}{}^{IAKey}$ (^{e}STR1) using the KeySetID values identified in the list.

b) The IFD compares the two copies of RND1 for each value to confirm that decryption was successful.

c) The IFD should traverse the entire list of KeySetID values irrespective of when the first successful decryption is performed and store the successful KeySetID values. This serves to prevent potential timing attacks.

d) Authentication fails if all KeySetID values have been used and decryption fails. Note that the same Asymmetric keys might be used in multiple key sets for large implementations.

e) The IFD extracts the ICC diversification seed data DivData and KeySetID value from the first successful STR1 decryption.

6.5 Step 5 — FINAL AUTHENTICATE command

a) The IFD generates a random value (RND2) using a RNG. The size of RND2 is identical to the key size of the selected AES-128 cipher (16 bytes).

b) The IFD calculates SHA-256 [RND1||RND2]; the result is denoted as KeysHash.

c) The IFD uses the diversification data (DivData) and calculates the diversified final authenticate key where $FAKey^{(Div)} = AES_{Encrypt}{}^{FAKey}$ (DivData). The FAKey to be used is referenced by the KeySetID identified as successful in the earlier IA Response evaluation.

d) The IFD creates the bit string STR2: OpModeID || RND2 || <Payload> || KeysHash

e) If needed, padding shall consist of one mandatory byte set to 0x80 followed, if required, by 0 to k–1 bytes set to 0x00, until the respective data block is filled up to k bytes, complying with ISO/IEC 9797-1 padding method 2.

f) The IFD calculates $^{e}STR2$ where $^{e}STR2 = AES_{Encrypt}{}^{FAKey(Div)}$ (STR2). The cipher mode for this operation is CBC.

g) The IFD transmits the Final Authenticate string $^{e}STR2$ to the ICC.

6.6 Step 6 — FINAL AUTHENTICATE command evaluation

a) The ICC calculates STR2 where $STR2 = AES_{Decrypt}{}^{FAKey(Div)}$ ($^{e}STR2$). The $FAKey^{(Div)}$ to be used is referenced by the KeySetID used in the earlier IA Response.

b) The ICC calculates KeysHash as SHA-256 [RND1||RND2] using RND1 generated in the previous IA Command step and RND2 extracted from STR2.

c) The ICC compares KeysHash with the KeysHash extracted from STR2. If a mismatch occurs then the ICC responds using a random byte string encrypted with the ShillKey, thereby preventing any indication that an error has occurred.

d) If the optional payload is sent then it is decrypted and processed as required by the implementation rules.

6.7 Step 7 — FINAL AUTHENTICATE response

a) The ICC retrieves the appropriate fields, based on the OpModeID extracted from STR2. These would normally be the appropriate Wiegand, ID or UUID numbers.

b) The ICC creates the bit string STR3: ACSRecord || <Payload>|| DivData

c) If needed, padding shall consist of one mandatory byte set to 0x80 followed, if required, by 0 to k–1 bytes set to 0x00, until the respective data block is filled up to k bytes, complying with ISO/IEC 9797-1 padding method 2.

d) The ICC calculates $^{e}STR3$ where $^{e}STR3 = AES_{Encrypt}{}^{KeysHash}$ (STR3). The cipher mode for this operation shall be CBC.

e) The ICC transmits the Final Authenticate string $^{e}STR3$ to the IFD.

6.8 Step 8 — FINAL AUTHENTICATE response evaluation

a) The IFD calculates STR3 where $STR3 = AES_{Decrypt}{}^{KeysHash}$ ($^{e}STR3$).

b) The IFD compares the transmitted DivData with the IFD copy received in the IA Response. Authentication fails if they do not match.

c) The ACSRecord and Payload is extracted from STR3 and can now be considered to be authenticated.

d) The optional payload may now be processed as required by the implementation rules.

e) The ACSRecord may now be passed to whichever back office system is appropriate to check for revocation and then open a door or to be part of some further process. The exact means of securing the record beyond this point is outside the scope of this standard.

f) Further authentication protocols or card access protocols may optionally use the generated session key KeysHash as a secure messaging, session or encryption key in subsequent sessions. The cipher mode for this operation shall be CBC.

7 Application identification

The PLAID application shall be selected by either

a) calling the generic PLAID default AID directly at "E0 28 81 C4 61 01", or

b) setting PLAID to be the default application, or

c) the scheme manager registering a specific AID for a specific scheme according to ISO/IEC 7816-5, using the appropriate ISO/IEC registration authority listed here: http://www.iso.org/iso/standards_development/maintenance_agencies.htm.

Although there can only be one default application on an ICC more than one implementation may be supported per card or reader as long as the appropriate AID is explicitly called.

8 Command set

The following are the specific commands from ISO/IEC 7816-4, which are required to comply with this International Standard. These commands use the General Authenticate command in two modes. The IA and FA commands denote related INS values representing the RSA and AES ciphers found in ISO/IEC 7816-4 General Authenticate operation, respectively.

Table 2 — ISO/IEC 7816-4 command set

Operation	CLA	INS	P1 value	P2 value	LC value	Body
Initial Authenticate	0x00	0x87	0x00	0x00	0x00	BER-TLV
Final Authenticate	0x00	0x86	0x00	0x00	Variable	Binary encrypted data

NOTE Standard length APDU response of 256 bytes length is supported by short length LC value of 0x00 under ISO/IEC 7816-4; see the notes to Table 1. This removes the need for an additional APDU.

9 Status bytes and error handling

Error codes shall be in accordance with ISO/IEC 7816-4. In order to protect from identity leakage, and to minimize useful information available to an attacker, an error during a PLAID operation shall not generate status byte error codes according to ISO/IEC 7816-4. Such status bytes would indicate that an inconsistency or failed attempt has occurred. Instead, the ICC shall use a ShillKey to complete the operation and return a status byte of SW_OK.

Table 3 — Status Bytes

Error code name	SW1-SW2 Value	Comment
SW_OK	0x9000	See ISO/IEC 7816-4

10 Key diversification

PLAID utilizes key diversification of the AES symmetric keys to ensure that the system remains secure should an individual ICC be compromised and its secret keys determined. The algorithm used to diversify the FAKey is as follows:

$$FAKey^{(Div)} = AES_{Encrypt}{}^{FAKey}\ (DivData)$$

11 Session key generation

PLAID results in the generation of an AES session key that may optionally be used for subsequent communications with the ICC. The size of this session key is determined by the key size of the AES cipher selected. Currently there are only three legitimate key sizes supported by AES (16, 24 or 32 bytes). Since AES uses 128-bit (16-byte) blocks for encryption/decryption, padding may be required up to the next block. The process used to generate the session key is as follows:

SessionKey (KeysHash) = SHA-256 [RND1 || RND2]

The default hashing algorithm is SHA-256. The hashing algorithm used needs to produce a message digest that matches the key size of the selected AES cipher. Where this is not possible, a hashing algorithm producing a message digest larger than the AES key size is to be used, with the additional trailing output bytes truncated to the required length.

12 Default mode

PLAID may be implemented with a number of options involving differing ciphers and modes of those ciphers or hashing algorithms.

Table 4 provides the default mode of operation and ciphers which should be utilized by implementations using the default AID ‘E0 28 81 C4 61 01’ and seeking interoperability.

Table 4 — Default Mode

Symmetric cipher	Key length (bits)	AES Padding	AES mode	Asymmetric cipher	Key length (bits)	RSA padding	RSA option	Hashing algorithm
AES	128	ISO/IEC 9797-1 padding method 2	CBC (see note)	RSA	2048	PKCS1.5	CRT	SHA-256

NOTE Key Diversification of DivData under AES uses a single block of data and therefore no mode is required

Annex A
(normative)

Test vectors

This Annex defines test vectors for PLAID, using default modes discussed in Clause 12, and based on the numbered transaction steps in Figure 1.

Due to the large size of RSA-2048, AES-128 and SHA-256 binary test vectors, and the likelihood of transposition mistakes in both their publication and transposition for subsequent use, the content of this Annex is provided in a separate .rtf file available at http://standards.iso.org/iso/25185/-1/.

Annex B
(informative)

Key management policy

While specific methodologies for key management are outside the scope of this International Standard, any implementation could be compromised without an adequate key management policy.

PLAID is different to most other APs in that it is a hybrid of symmetric and asymmetric cryptography, and that it performs a mutual authentication rather than just an “external” authenticate (from the card point of view) .

In order to protect from directed and privacy attacks, after the first IFD APDU, no data is ever passed in the clear, and the only data that is obtainable freely is the IFDs list of KeySetID values (data which for most implementations would be obvious from the location of the reader in any case)

The Initial Authenticate steps use an asymmetric or one-way cipher. This protects the ICC diversification seed (DivData) and the first random number challenge (RND1) from exposure in the clear. The Final Authenticate is a symmetric key authentication similar to other common contactless authentication protocols.

This hybrid authentication protocol is therefore very different to the more familiar PKI authentication even though both use Asymmetric ciphers. The role of RSA public and private keys in PLAID is different than a PKI.

This is because for PLAID the Initial Authenticate (first) pass proves the ICC knows secrets known only to the IFD rather than secrets known by the ICC. In the second pass (Final Authenticate) this is reversed.

In effect both the public and private RSA keys, as well as the AES keys need to be secured by both the on-card applet as well as the reader. These keys should therefore be managed as if they were shared symmetric keys.

Key management policy should therefore ensure all keys are stored in secure cryptographic modules, such as certified ICCs, Security Access Modules, or specialist Hardware Security Modules and in the back office with strong Key Management Systems, preferably with hardware based cryptographic support.

NOTE Achieving these levels of security on hardware modules has recently become much easier with low cost ICC, SAM and IFD chips becoming available that are both highly certified and which can natively support both PLAID and its key management. Examples of the code required for these devices can be downloaded from the PLAID reference implementation (refer to Annex D).

Annex C
(informative)

Keyset management

This International Standard allows for up to 65 535 Keysets determined by the 2-byte KeySetID record. The ICC may therefore support a minimum of two and as many other Keysets as is viable, given the memory of the ICC. This International Standard also supports negotiation of Keysets, where the IFD, in the Initial Authenticate command, gives the ICC a list of supported Keysets, in preference order, and the ICC will utilize the most preferred Keyset of the set or sets it has loaded.

This allows that different levels of trust and interoperability may be achieved depending on the business requirements of the implementation. These might be building, role or function based structuring of Keysets or some combination of these or other factors.

For example, an implementation might utilize the following Keysets:

Table 5 — Keyset management

Administration	Keyset = 0	For administration of the PLAID application only.
Shared	Keyset = 1	For authentication to shared public areas of a range of buildings where trusted persons could enter the outer perimeter only.
Physical Access	Keyset = 2	For authentication to PACS systems within the outer perimeter.
Computer room	Keyset = 3	For authentication for computer room access and highly secure areas with separate PACS system.
Other Access	Keyset = 4	For authentication for printer access, etc.

Annex D
(informative)

Reference implementation

A reference implementation for PLAID and PLAID SAM is available to assist in the comprehensive understanding of how to implement this standard.

The reference implementation, test vectors and other useful developer's tools may be downloaded from the following URL: https://www.plaid.gov.au.

Annex E
(informative)

Identity leakage considerations

E.1 General

ID-Leakage may occur in passive or active forms. Passive ID-Leakage is where unique per card or per scheme data is available simply by recording the session between IFD and ICC. Active ID-Leakage is where specific tools or viruses scan the ICC for useful identification or private data at likely addresses on the card.

Passive attacks are relatively easily eliminated by the methods used by PLAID, subject to appropriate initialization of the ICC as discussed below.

Active attacks may be impractical to fully eliminate, since most existing standards require freely available information to be available at predictable ICC addresses, and resolving this is beyond the scope of this Standard. The discussion below provides some options for consideration in order to minimize the impact of ID-leakage.

E.2 Passive ID-Leakage

In order to remove possible passive ID-leakage when implementing PLAID, the following additional checks should be considered as part of the pre-personalization set up of a PLAID ICC.

— In the case of contactless PICC, the UID generated by the PICC for the anti-collision procedure in ISO/IEC 14443-3 should be specified to use the "random" option according to ISO/IEC 14443-3. This generally needs to be set prior to card personalization or in some cases at manufacture.

— In implementations where ID-leakage of any form cannot be tolerated, care may need to be taken to ensure the ATR/ATQ response does not contain unique per card or per scheme identifying data, particularly in the historical bytes found in ISO/IEC 7816-3 and ISO/IEC 7816-4. Any such data which must be set may be best set to null values. This generally needs to be set prior to card personalization or in some cases at manufacture.

— Check for any possible session dialogue with other applications on the ICC, particularly any which are set as the default application.

E.3 Active ID-Leakage

— Check the status and session dialogue of all applications on the card, particularly generic or diagnostic applications from the manufacturer which may be instantiated in the Read Only Memory (ROM) mask, and may or may not be formally documented or normally disclosed by the manufacturer.

— Administrative functions such as card management are generally carried out using a contact reader. Many ICCs can differentiate programmatically between contact and contactless interfaces, and can refuse access to selected applications from the contactless interface. Consider switching off access to administrative applications from contactless interfaces, particularly ones which store unique card identification information such as the GlobalPlatform Card Production Life Cycle (CPLC) data.

Annex F
(informative)

Operational mode management

This International Standard allows for up to 65 535 operational modes determined by the 2-byte ACS record field sent to the ICC in the Final Authenticate command. Different values for the ACSRecord are subsequently authenticated and returned by the Final Authenticate response. This facility allows that a different and distinct ACS record can be passed to the IFD and back-end systems depending on the business requirements for authorization for the implementation.

For example, an implementation might utilize the following operational modes:

Table 6 — Operational mode management

New Buildings	ACSRecord = 1	E.g. RFC 4122 based UUID string being returned for authorization within new building systems.
Old buildings	ACSRecord = 2	E.g. 26 bit Wiegand string being returned for authorization within older building systems.
Logical Access	ACSRecord = 3	E.g. FIPS 201 based CHUID/FASCN string being returned for authorization to system login, printer access, etc.
Computer room	ACSRecord = 4	E.g. RFC 4122 based UUID string being returned for authorization to computer room access and highly secure areas.

NOTE There may or may not be a one-one correspondence between OpModeID and KeySetID in any one implementation. For instance during transition there may be a single KeySetID utilized for building access, but new buildings might use one OpModeID whilst old buildings use another in order to transition from their use of the older Wiegand based numbering.

Annex G
(informative)

PLAID security features

This Annex defines the logic and discusses the purpose of each step of the protocol. The protocol is described in the order and structure illustrated in Figure 1, which should be referred to in consideration of the description below.

G.1 INITIAL AUTHENTICATE — General

— The primary purpose of the IA Command is to protect the privacy and value of the Diversification Data (DivData) which is needed to diversify the keys involved in the subsequent FA operation so as to ensure they are not exposed in the clear.

— Other authentication protocols are known to expose the value of this diversification data in the clear as the card serial number or UID, which results in the potential for directed attacks. By using this step the PLAID AP allows all unique identifying information otherwise available from the ICC to be closed down or secured by keys.

— The secondary purpose of this step is to prevent knowledge of the value of RND1 (which is generated by the ICC for the specific session).

— Any compromise of the IA step is not fatal to the subsequent FA step, since the only actual compromise is to the value of RND1 and DivData.

— This step uses an Asymmetric Cryptographic cipher (RSA).

— The ICC encrypts using the public exponent and the modulus of an RSA key so as to obtain maximum performance in the on-card encryption.

— Only IFDs with access to the private RSA key material are able to decrypt the ICC response.

— Both keys are however protected; they are managed as if they were shared symmetric keys.

— This step must be very fast. Selection of the Asymmetric cipher and key lengths needs to consider the capability of operational ICCs and IFDs to achieve this speed, as well as the fact that compromise of this step only impacts the privacy of DivData and RND1, not the entire AP transaction.

G.1.1 Step 1 — INITIAL AUTHENTICATE command

— Initiates the AP.

— All data is in the clear, but of no use to an attacker.

— Passes a list of supported Keysets in preference order from the IFD to the ICC.

— The list of Keysets is of little use to an attacker since it does not contain any key values, and the actual list is usually obvious in any case based on the location or use-case.

— Uses a BER-TLV encoding method for maximum extensibility.

— Data is very small, therefore speed is fast.

G.1.2 Step 2 — INITIAL AUTHENTICATE command evaluation

— Parses the list of Keysets and chooses the Keyset for the remainder of the AP session.

G.1.3 Step 3 — INITIAL AUTHENTICATE response

— DivData is unique per ICC and instantiation and is generated at ICC instantiation, preferably as a random, but unique number. It is used to diversify the AES key used in the FA steps.

— DivData is retrieved from protected memory and RND1 is generated using the ICC RNG. Note that RNG strength is important otherwise an attack might be possible.

— The string 'KeySetID||DivData||RND1||RND1' is generated and encrypted using the chosen Asymmetric cipher and returned to the IFD.

— The repeat of RND1 is used as a checksum.

— An Asymmetric cipher is used to ensure that in the event of a single compromised ICC, an adversary will be unable to decrypt the air traffic from any ICC.

— By using an asymmetric cipher, an attacker who fully compromises an ICC cannot even decrypt the compromised ICCs IA response since they only have the public key and the corresponding private key material that is required to decrypt any card is held in the back office or in SAM devices.

G.1.4 Step 4 — INITIAL AUTHENTICATE response evaluation

— The IFD decrypts the response string using the IAKey and checks that RND1 is repeated.

— If the IFD can successfully decrypt and validate the IA command then the ICC will have proven knowledge of an approved IA key value. The ICC presented Diversification Data are used to validate the FA command.

G.2 FINAL AUTHENTICATE — General

— The FA command passes the IFD generated RND2 and the composite KeysHash to the IFD.

— KeysHash is required in order to link the steps and therefore guarantee that a rogue device cannot place itself in the middle of the communications between the IA and FA steps.

— The FA command uses the symmetric AES-128 cipher. This is in order to obtain performance with an adequate key length. This is achievable with AES-128.

— The FA command also passes the OpModeID, so that the ICC can determine the type of system the IFD is, and send it the correct access control records or payload.

G.2.1 Step 5 — FINAL AUTHENTICATE command

— Generates RND2 and calculates KeysHash based on the ICC generated RND1. KeysHash is therefore a composite of the IFD and ICC generated random numbers and will be as strong as the superior of the two RNGs.

— Passes OpModeID to the ICC so that the ICC only responds with just the credential data required for the particular use-case. This reduces unnecessary data being passed over the interface and reduces the possibility of private data leakage. It also makes transition easier since the ICC can support many different PACS record types.

— Communication is AES-128 encrypted using the key calculated for the ICC based on DivData obtained in the previous step.

G.2.2 Step 6 — FINAL AUTHENTICATE command evaluation

— If the ICC can successfully decrypt and validate the FA command then the ICC will consider the IFD authenticated. A successful decryption is determined by the following characteristics.

  — The ICC calculation for KeysHash matches the IFD calculation for KeysHash. This indicates to the ICC that in the previous step, the IFD successfully decrypted the IA command to retrieve the value of the Diversification Data and RND1.

  — The ICC decryption of the FA command is validated (by KeysHash comparisons). This indicates to the ICC that IFD has used the correct unique per ICC diversified AES key.

G.2.3 Step 7 — FINAL AUTHENTICATE response

— AES encrypt the payload credential information (ACSRecord, < Payload >).

— Also AES encrypt DivData which is returned as a confirmation of the linkage between the IA response and the FA response as a backup method to KeysHash to prevent session hijacking.

— Response uses the composite KeysHash as the key to AES encrypt the string. This ensures continuity of the authentication from the first IA response.

G.2.4 Step 8 — FINAL AUTHENTICATE response evaluation

— If the IFD can successfully decrypt the FA command, then the ICC will consider the ICC authenticated and the data presented authenticated. Successful decryption is determined by confirming that the Diversification Data presented in the FA command matches the Diversification Data presented in the previous IA command. A match confirms the following:

  — decryption was successful;

  — the session has not been hijacked.

Bibliography

[1] ISO/IEC 8824-2, *Information technology — Abstract Syntax Notation One (ASN.1): — Part 2: Information object specification*

[2] ISO/IEC 14443-3, *Identification cards — Contactless integrated circuit cards — Proximity cards — Part 3: Initialization and anti-collision*

ICS 35.240.15

Price based on 20 pages