Archived NIST Technical Series Publication

The attached publication has been archived (withdrawn), and is provided solely for historical purposes. It may have been superseded by another publication (indicated below).

Archived Publication

Series/Number:	NIST Special Publication 800-73-2
Title:	Interfaces for Personal Identity Verification
Publication Date(s):	September 2008
Withdrawal Date:	February 2010
Withdrawal Note:	SP 800-73-2 is superseded in its entirety by the publication of SP 800-73-3 (February 2010).

Superseding Publication(s)

The attached publication has been **superseded by** the following publication(s):

Series/Number:	NIST Special Publication 800-73-3
Title:	Interfaces for Personal Identity Verification
Author(s):	Ramaswamy Chandramouli, David Cooper, James F. Dray, Hildegard Ferraiolo, Scott B. Guthery, William MacGregor, Ketan Mehta
Publication Date(s):	February 2010
URL/DOI:	http://dx.doi.org/10.6028/NIST.SP.800-73-3

Additional Information (if applicable)

Contact:	Computer Security Division (Information Technology Lab)
Latest revision of the attached publication:	SP 800-73-4 (as of August 6, 2015)
Related information:	http://csrc.nist.gov/groups/SNS/piv/
Withdrawal announcement (link):	N/A

NIST
National Institute of
Standards and Technology
U.S. Department of Commerce

Date updated: August 6, 2015

NIST Special Publication 800-73-2

National Institute of Standards and Technology
U.S. Department of Commerce

Interfaces for Personal Identity Verification – Part 1: End-Point PIV Card Application, Namespace, Data Model and Representation

Ramaswamy Chandramouli
James F. Dray
Hildegard Ferraiolo
Scott B. Guthery
William MacGregor
Ketan Mehta

INFORMATION SECURITY

Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD, 20899-8930

September 2008

U.S. Department of Commerce
Carlos M. Gutierrez, Secretary

National Institute of Standards and Technology
James M. Turner, Deputy Director

Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the Nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analyses to advance the development and productive use of information technology. ITL's responsibilities include the development of management, administrative, technical, and physical standards and guidelines for the cost-effective security and privacy of non-national security-related information in Federal information systems. This special publication 800-series reports on ITL's research, guidelines, and outreach efforts in information system security, and its collaborative activities with industry, government, and academic organizations.

National Institute of Standards and Technology Special Publication 800-73-2, Part 1, 41 pages, September 2008)

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

NIST makes no representation as to whether or not one or more implementations of SP 800-73-2 is/are covered by existing patents.

Acknowledgements

The authors (Ramaswamy Chandramouli, James Dray, Hildegard Ferraiolo, William MacGregor of NIST, Ketan Mehta of Mehta Inc. and Scott Guthery of Mobile Mind Inc.) wish to thank their colleagues who reviewed drafts of this document and contributed to its development. Special thanks to the Government Smart Card Interagency Advisory Board (GSC-IAB) and InterNational Committee for Information Technology Standards (INCITS) for providing detailed technical inputs to the SP 800-73-2 development process. The authors also gratefully acknowledge and appreciate the many contributions from the public and private sectors whose thoughtful and constructive comments improved the quality and usefulness of this publication.

TABLE OF CONTENTS

List of Appendices

List of Tables

List of Figures

1. Introduction

The Homeland Security Presidential Directive 12 (HSPD-12) called for a common identification standard to be adopted governing the interoperable use of identity credentials to allow physical and logical access to Federal government locations and systems. The Personal Identity Verification (PIV) of Federal Employees and Contractors, Federal Information Processing Standard 201 (FIPS 201) [1] was developed to establish standards for identity credentials. Special Publication 800-73-2 (SP 800-73-2) contains technical specifications to interface with the smart card (PIV Card[1]) to retrieve and use the identity credentials.

1.1 Authority

This document has been developed by the National Institute of Standards and Technology (NIST) in furtherance of its statutory responsibilities under the Federal Information Security Management Act (FISMA) of 2002, Public Law 107-347.

NIST is responsible for developing standards and guidelines, including minimum requirements, for providing adequate information security for all agency operations and assets, but such standards and guidelines shall not apply to national security systems. This recommendation is consistent with the requirements of the Office of Management and Budget (OMB) Circular A-130, Section 8b(3), Securing Agency Information Systems, as analyzed in A-130, Appendix IV: Analysis of Key Sections. Supplemental information is provided A-130, Appendix III.

This recommendation has been prepared for use by federal agencies. It may be used by non-governmental organizations on a voluntary basis and is not subject to copyright though attribution is desirable. Nothing in this document should be taken to contradict standards and guidelines made mandatory and binding on Federal agencies by the Secretary of Commerce under statutory authority. Nor should this recommendation be interpreted as altering or superseding the existing authorities of the Secretary of Commerce, Director of the Office of Management and Budget (OMB), or any other Federal official.

1.2 Purpose

FIPS 201 defines procedures for the PIV lifecycle activities including identity proofing, registration, PIV Card issuance, and PIV Card usage. FIPS 201 also specifies that the identity credentials must be stored on a smart card. SP 800-73-2 contains the technical specifications to interface with the smart card to retrieve and use the identity credentials. The specifications reflect the design goals of interoperability and PIV Card functions. The goals are addressed by specifying a PIV data model, card edge interface, and application programming interface. Moreover, SP 800-73-2 enumerates requirements where the standards include options and branches. The specifications go further by constraining implementers' interpretations of the normative standards. Such restrictions are designed to ease implementation, facilitate interoperability, and ensure performance, in a manner tailored for PIV applications.

[1] A physical artifact (e.g., identity card, "smart" card) issued to an individual that contains stored identity credentials (e.g., photograph, cryptographic keys, biometric data) so that the claimed identity of the cardholder can be verified against the stored credentials by another person (human readable and verifiable) or an automated process (computer readable and verifiable).

1.3 Scope

SP 800-73-2 specifies the PIV data model, Application Programming Interface (API) and card interface requirements necessary to comply with the use cases, as defined in Section 6 of FIPS 201 and further described in this document. Interoperability is defined as the use of PIV identity credentials such that client-application programs, compliant card applications, and compliant integrated circuits cards (ICC) can be used interchangeably by all information processing systems across Federal agencies. SP 800-73-2 defines the PIV data elements identifiers, structure and format. SP 800-73-2 also describes the client application programming interface and card command interface for use of the PIV Card.

This part, SP 800-73-2, Part 1 – *End-Point PIV Card Application Namespace, Data Model and Representation*, specifies the End-Point PIV Card Application Namespace, the PIV Data Model and its logical representation on the PIV Card and is a companion document to FIPS 201.

1.4 Audience and Assumptions

This document is targeted at Federal agencies and implementers of PIV systems. Readers are assumed to have a working knowledge of smart card standards and applications.

1.5 Document Overview and Structure

All sections in this document are *normative* (i.e., mandatory for compliance) unless specified as *informative* (i.e., non-mandatory). Following is the structure of this document:

+ Section 1, *Introduction*, provides the purpose, scope, audience, and assumptions of the document and outlines its structure.
+ Section 2, *PIV Card Application Namespace,* defines the three NIST managed namespaces used by the PIV Card Application.
+ Section 3, *End-Point PIV Data Model Elements*, describes the PIV Data Model elements in detail.
+ Section 4, *End-Point PIV Data Objects Representation*, describes the format and coding of the PIV data structures used by the PIV client-application programming interface and the PIV Card Application.
+ Section 5, *End-Point Data Types and Their Representation*, provides the details of the data types found on the PIV client-application programming interface and the PIV Card Application card command interface.

1.5.1 Appendices

The appendices contain material that needs special formatting together with illustrative material to aid in understanding information in the body of the document.

2. PIV Card Application Namespaces

2.1 Namespaces of the PIV Card Application

Names used on the PIV interfaces are drawn from three namespaces managed by NIST:

- Proprietary Identifier extension (PIX) of the NIST Registered Application Provider Identifier (RID)
- ASN.1 object identifiers (OIDs) in the personal verification subset of the OIDs managed by NIST
- Basic Encoding Rules – Tag Length Value (BER-TLV) tags of the NIST PIV coexistent tag allocation scheme

All unspecified names in these managed namespaces are reserved for future use.

All interindustry tags defined in ISO/IEC 7816, *Information Technology – Identification Cards – Integrated Circuit(s) Card with Contacts* [2], and used in the NIST coexistent tag allocation scheme without redefinition have the same meaning in the NIST PIV coexistent tag allocation scheme as they have in [2].

All unspecified values in the following identifier and value namespaces are reserved for future use:

- algorithm identifiers
- key reference values
- cryptographic mechanism identifiers

2.2 PIV Card Application AID

The Application Identifier (AID) of the Personal Identity Verification Card Application (PIV Card Application) shall be:

'A0 00 00 03 08 00 00 10 00 01 00'

The AID of the PIV Card Application consists of the NIST RID ('A0 00 00 03 08') followed by the application portion of the NIST PIX indicating the PIV Card Application ('00 00 10 00') and then the version portion of the NIST PIX ('01 00') for the first version of the PIV Card Application. All other PIX sequences on the NIST RID including the trailing five bytes of the PIV Card Application AID are reserved for future use.

The PIV Card Application can be selected as the current application by providing the full AID as listed above or by providing the right-truncated version; that is, without the two-byte version, as follows:

'A0 00 00 03 08 00 00 10 00'

3. End-Point PIV Data Model Elements

This section contains the description of the data elements for personal identity verification, the PIV data model.

A PIV Card Application shall contain five mandatory interoperable data objects and may contain six optional interoperable data objects. The five mandatory data objects for interoperable use are as follows:

1. Card Capability Container
2. Cardholder Unique Identifier
3. X.509 Certificate for PIV Authentication
4. Cardholder Fingerprints
5. Security Object

The six optional data objects for interoperable use are as follows:

1. Cardholder Facial Image
2. Printed Information
3. X.509 Certificate for Digital Signature
4. X.509 Certificate for Key Management
5. X.509 Certificate for Card Authentication
6. Discovery Object

3.1 Mandatory Data Elements

The five mandatory data objects support FIPS 201 minimum mandatory compliance.

3.1.1 Card Capability Container

The Card Capability Container (CCC) is mandatory for compliance with the Government Smart Card Interoperability Specification (GSC-IS) [3]. It supports minimum capabilities for retrieval of data model and application information.

The data model of the PIV Card Application shall be identified by data model number 0x10. Deployed applications use 0x00 through 0x04. This enables the GSC-IS application domain to correctly identify a new data model name space and structure as defined in this document.

3.1.2 Cardholder Unique Identifier

The Cardholder Unique Identifier (CHUID) data object is defined in accordance with the Technical Implementation Guidance: Smart Card Enabled Physical Access Control Systems (TIG SCEPACS) [4]. For this specification, the CHUID is common between the contact and contactless chips. For dual chip implementations, the CHUID is copied in its entirety between the two chips.

In addition to the requirements specified in TIG SCEPACS, the CHUID on the PIV Card shall meet the following requirements:

- The Buffer Length field is an optional TLV element. This element is the length in bytes of the entire CHUID, excluding the Buffer Length element itself, but including the

CHUID's Asymmetric Signature element. The calculation of the asymmetric signature must exclude the Buffer Length element if it is present.

+ The Federal Agency Smart Credential Number (FASC-N) shall be in accordance with TIG SCEPACS[4]. A subset of FASC-N, the FASC-N Identifier, shall be the unique identifier as described in [4, 6.6]: “The combination of an Agency Code, System Code, and Credential Number is a fully qualified number that is uniquely assigned to a single individual”. The Agency Code is assigned to each Department or Agency by Special Publication 800-87 (SP 800-87) *Codes for the Identification of Federal and Federally-Assisted Organizations* [5]. The subordinate System Code and Credential Number value assignment is subject to Department or Agency policy, provided that the FASC-N identifier (i.e. the concatenated Agency Code, System Code, and Credential Number) is unique for each card. The same FASC-N value shall be used in all the PIV data objects that include the FASC-N. To eliminate unnecessary use of the SSN[2], the FASC-N’s Person Identifier (PI) field should not encode the SSN. TIG SCEPACS also specifies PACS interoperability requirements in section 2.1, 10th paragraph of [4, 2.1]: “For full interoperability of a PACS it must at a minimum be able to distinguish fourteen digits (i.e., a combination of an Agency Code, System Code, and Credential Number) when matching FASC-N based credentials to enrolled card holders.”

+ The Global Unique Identifier (GUID) field must be present, and may include either an issuer assigned IPv6 address or be coded as all zeros. The GUID is included to enable future migration away from the FASC-N into a robust numbering scheme for all issued credentials.

+ The DUNS and Organizational Code fields are optional.

+ The Authentication Key Map[3] is specified as an optional field which enables the application to discover the key reference.

+ The Expiration Date is mapped to the reserved for future use (RFU) tag 0x35, keeping that within the existing scope of the TIG SCEPACS specification. This field shall be 8 bytes in length and shall be encoded as YYYYMMDD.

+ The CHUID is signed in accordance with FIPS 201. The card issuer’s digital signature key shall be used to sign the CHUID and the associated certificate shall be placed in the signature field of the CHUID.

3.1.3 X.509 Certificate for PIV Authentication

The X.509 Certificate for PIV Authentication and its associated private key, as defined in FIPS 201, is used to authenticate the card and the cardholder. The read access control rule for the X.509 Certificate for PIV Authentication is “Always,” meaning the certificate can be read without access control restrictions. The Public Key Infrastructure (PKI) cryptographic function (see Table 3) is protected with a "PIN" access rule. In other words, private key operations using the PIV Authentication Key require the Personal Identification Number (PIN) to be submitted, but a successful PIN submission enables multiple private key operations without additional cardholder consent.

[2] See the attachment to OMB M-07-16, Section 2: “Reduce the Use of Social Security Numbers”.
[3] The Authentication Key Map is deprecated. It will be eliminated in a future revision of SP 800-73.

3.1.4 Cardholder Fingerprints

The fingerprint data object specifies the primary and secondary fingerprints in accordance with the FIPS 201. The Common Biometric Exchange Formats Framework (CBEFF) header shall contain the FASC-N and shall require the Integrity Option. The header shall not require the Confidentiality Option.

3.1.5 Security Object

The Security Object is in accordance with Appendix C of PKI for Machine Readable Travel Documents (MRTD) Offering ICC Read-Only Access Version 1.1 [6]. Tag 0xBA is used to map the ContainerIDs in the PIV data model to the 16 Data Groups specified in the MRTD. The mapping enables the Security Object to be fully compliant for future activities with identity documents.

The “DG-number-to-Container-ID” mapping object TLV in tag 0xBA encapsulates a series of three byte triples - one for each PIV data object included in the Security Object. The first byte is the Data Group (DG) number, and the second and third bytes are the most and least significant bytes (respectively) of the Container ID value. The DG number assignment is arbitrary; however, the same number assignment applies to the DataGroupNumber(s) in the DataGroupHash(es). This will ensure that the ContainerIDs in the mapping object refers to the correct hash value in the Security Object (0xBB).

The 0xBB Security Object is formatted according to the MRTD [5, Appendix C]. The LDS Security Object itself must be in ASN.1 DER format, formatted as specified in [5, Appendix C.2]. This structure is then inserted into the encapContentInfo field of the Cryptographic Message Syntax (CMS) object specified in [5, Appendix C.1].

The card issuer's digital signature key used to sign the CHUID shall also be used to sign the Security Object. The signature field of the Security Object, Tag 0xBB shall omit the issuer’s certificate, since it is included in the CHUID. At a minimum, unsigned data objects, such as the Printed Information data object, shall be included in the Security Object if present. For maximum protection against credential splicing attacks (credential substitution), it is recommended, however, that all PIV data objects, except the PIV X.509 certificates, be included in the Security Object.

3.2 Optional Data Elements

The six optional data elements of FIPS 201, when implemented, shall conform to the specifications provided in this document.

3.2.1 Cardholder Facial Image

The photo on the chip supports human verification only. It is not intended to support facial recognition systems for automated identity verification.

3.2.2 Printed Information

All FIPS 201 mandatory information printed on the card is duplicated on the chip in this data object. The Security Object enforces integrity of this information according to the issuer. This provides specific protection that the card information must match the printed information, mitigating alteration risks on the printed media.

3.2.3 X.509 Certificate for Digital Signature

The X.509 Certificate for Digital Signature and its associate private key, as defined in FIPS 201, support the use of digital signatures for the purpose of document signing. The read access control rule for the X.509 Certificate is "Always", meaning the certificate can be read without access control restrictions. The Public Key Infrastructure (PKI) cryptographic function is protected with a "PIN Always" access rule. In other words, the PIN must be submitted every time immediately before a *Digital Signature Key* operation. This ensures cardholder participation every time the private key is used for digital signature generation.

3.2.4 X.509 Certificate for Key Management

The X.509 Certificate for Key Management and its associate private key, as defined in FIPS 201, support the use of encryption for the purpose of confidentiality. This key pair is escrowed by the issuer for key recovery purposes. The read access control rule for the X.509 Certificate is "Always", meaning the certificate can be read without access control restrictions. The PKI cryptographic function is protected with a "PIN" access rule. In other words, once the PIN is submitted, subsequent *Key Management Key* operations can be performed without requiring the PIN again. This enables multiple private key operations without additional cardholder consent.

3.2.5 X.509 Certificate for Card Authentication

FIPS 201 specifies the optional Card Authentication Key (CAK) as an asymmetric or symmetric key that is used to support additional physical access applications. For an asymmetric CAK, the read access control rule of the corresponding X.509 Certificate for Card Authentication is "Always", meaning the certificate can be read without access control restrictions. Private (asymmetric) key operations or secret symmetric cryptographic operation is defined as "Always". In other words, the private or secret key can be used without access control restrictions. With extremely high probability, each PIV Card shall contain a unique CAK.

3.2.6 Discovery Object

The Discovery Object, if implemented, is the 0x7E interindustry ISO/IEC 7816-6 template that nests interindustry data objects. For the Discovery Object, the 0x7E template nests two BER-TLV structured interindustry data elements: 1) tag 0x4F contains the AID of the PIV Card Application and 2) tag 0x5F2F lists the PIN Usage Policy.

+ Tag 0x4F encodes the PIV Card Application AID as follows:
  {'4F 0B A0 00 00 03 08 00 00 10 00 01 00'}

+ Tag 0x5F2F encodes the PIN Usage Policy as follows:

  First byte: 0x40 indicates that the PIV Card Application PIN alone satisfies the PIV Access Control Rules (ACRs) for command execution[4] and object access.

  0x60 indicates that both the PIV Card Application PIN and Global PIN satisfy the PIV ACRs for command execution and PIV data object access.

[4] Command execution pertains to the VERIFY APDU and optionally to the CHANGE REFERENCE DATA APDU.

Bits 5 through 1 of the first byte are RFU.

The second byte of the PIN Usage Policy encodes the cardholder's PIN preference for PIV Cards with both the PIV Card Application PIN and the Global PIN enabled:

Second byte: 0x10 indicates that the PIV Card Application PIN is the primary PIN used to satisfy the PIV ACRs for command execution and object access.

0x20 indicates that the Global PIN is the primary PIN to satisfy the PIV ACRs for command execution and object access.

Note: If the first byte is set to 0x40, then second byte is RFU and shall be set to 0x00.

PIV Card Application that satisfy the PIV ACRs for PIV data object access and command execution [5]with both PIV Card Application PIN and Global PIN shall implement the Discovery Object with the PIN Usage Policy set to 0x60 zz where zz is set to either 0x10 or 0x20.

The encoding of the 0x7E Discovery Object is as follows:

> {'7E 12' {{'4F 0B A0 00 00 03 08 00 00 10 00 01 00'} {'5F 2F 02 xx yy'}}}, where xx and yy encode the first and second byte of the PIN Usage Policy as described in this section.

The Security Object enforces integrity of Discovery Object according to the issuer.

3.3 Data Object Containers and associated Access Rules and Interface Modes

Table 1 defines a high level view of the data model. Each on-card storage container is labeled either as Mandatory (M) or Optional (O). This data model is designed to enable and support dual interface cards. Note that access conditions based on the interface mode (contact vs. contactless) take precedence over all Access Rules defined in Table 1, Column 3.

Table 1. Data Model Containers

Container Name	Container ID	Access Rule for Read	Contact / Contactless[6]	M/O
Card Capability Container	0xDB00	Always	Contact	M
Cardholder Unique Identifier	0x3000	Always	Contact and Contactless	M
X.509 Certificate for PIV Authentication	0x0101	Always	Contact	M
Cardholder Fingerprints	0x6010	PIN	Contact	M
Security Object	0x9000	Always	Contact	M
Cardholder Facial Image	0x6030	PIN	Contact	O
Printed Information	0x3001	PIN	Contact	O
X.509 Certificate for Digital Signature	0x0100	Always	Contact	O
X.509 Certificate for Key	0x0102	Always	Contact	O

[5] 5 Command execution pertains to the VERIFY APDU and optionally to the CHANGE REFERENCE DATA APDU.

[6] Contact interface mode means the container is accessible through contact interface only. Contact and contactless interface mode means the container can be accessed from either interface.

Management				
X.509 Certificate for Card Authentication	0x0500	Always	Contact and Contactless	O
Discovery Object	0x6050	Always	Contact and Contactless	O

Appendix A provides a detailed spreadsheet for the data model. ContainerIDs and Tags within the containers for each data object are defined by this data model and in accordance with SP 800-73-2 naming conventions.

4. End-Point PIV Data Objects Representation

4.1 Data Objects Definition

A *data object* is an item of information seen on the card command interface for which are specified a name, a description of logical content, a format, and a coding. Each data object has a globally unique name called its *object identifier* (OID), as defined in ISO/IEC 8824-2:2002, *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification.* [7]

A data object whose data content is encoded as a BER-TLV data structure as in ISO/IEC 8825—1:2002 – ASN.1 encoding rules [8] is called *BER-TLV data object.*

4.1.1 Data Object Content

The *content* of a data object is the sequence of bytes that are said to be *contained in* or to be the *value of* the data object. The number of bytes in this byte sequence is referred to as the *length* of the data content and also as the *size* of the data object. The first byte in the sequence is regarded as being at *byte position* or *offset* zero in the content of the data object.

The data content of a BER-TLV data object may consist of other BER-TLV data objects. In this case the tag of the data object indicates that the data object is a *constructed data object*. A BER-TLV data object that is not a constructed data object is called a *primitive data object*.

The PIV End-Point Data objects are BER-TLV objects encoded as per [8], except that Tag values (T-values) of the PIV data object's inner tag assignments do not conform to BER-TLV requirements.[7] This is due to the need to accommodate legacy tags inherited from the GSC-IS.

4.2 OIDs and Tags of PIV Card Application Data Objects

Table 2 lists the ASN.1 object identifiers and BER-TLV tags of the eleven PIV Card Application data objects for interoperable use. For the purpose of constructing PIV Card Application data object names in the CardApplicationURL in CCC of the PIV Card Application, the NIST RID ('A0 00 00 03 08') shall be used and the card application type shall be set to '00'.

4.3 Object Identifiers

Each of the data objects in the PIV Card Application has been provided with a three-byte BER-TLV tag and an ASN.1 OID from the NIST personal verification arc. These object identifier assignments are given in Table 2.

A data object shall be identified on the PIV client-application programming interface using its OID. An object identifier on the PIV client-application programming interface shall be a dot delimited string of the integer components of the OID. For example, the representation of the OID of the CHUID on the PIV client-application programming interface is “2.16.840.1.101.3.7.2.48.0”.

[7] The exception does not apply to the Discovery Object, nor the Application Property Template (APT), since these objects use interindustry tags from ISO/IEC 7816-6.

A data object shall be identified on the PIV Card Application card command interface using its BER-TLV tag. For example, the CHUID is identified on the card command interface to the PIV Card Application by the three-byte identifier '5FC102'.

Table 1 lists the ACRs of the eleven PIV Card Application data objects for interoperable use. See table 6-3 in Special Publication 800-78 (SP 800-78) *Cryptographic Algorithms and Key Sizes for Personal Identity Verification* [9], for the key references and permitted algorithms associated with these authenticatable entities.

Table 2. Object Identifiers of the PIV Data Objects for Interoperable Use

Data Object for Interoperable Use	ASN.1 OID	BER-TLV Tag	M/O
Card Capability Container	2.16.840.1.101.3.7.1.219.0	'5FC107'	M
Cardholder Unique Identifier	2.16.840.1.101.3.7.2.48.0	'5FC102'	M
X.509 Certificate for PIV Authentication	2.16.840.1.101.3.7.2.1.1	'5FC105'	M
Cardholder Fingerprints	2.16.840.1.101.3.7.2.96.16	'5FC103'	M
Security Object	2.16.840.1.101.3.7.2.144.0	'5FC106'	M
Cardholder Facial Image	2.16.840.1.101.3.7.2.96.48	'5FC108'	O
Printed Information	2.16.840.1.101.3.7.2.48.1	'5FC109'	O
X.509 Certificate for Digital Signature	2.16.840.1.101.3.7.2.1.0	'5FC10A'	O
X.509 Certificate for Key Management	2.16.840.1.101.3.7.2.1.2	'5FC10B'	O
X.509 Certificate for Card Authentication	2.16.840.1.101.3.7.2.5.0	'5FC101'	O
Discovery Object	2.16.840.1.101.3.7.2.96.80	'7E'	O

5. End-Point Data Types and Their Representation

This section provides a description of the data types used in the PIV Client-Application Programming Interface (SP 800-73-2, Part 3) and PIV Card Command Interface (SP 800-73-2, Part 2). Unless otherwise indicated, the representation shall be the same on both interfaces.

The data types are defined in Part 1, rather than in Parts 2 and 3 in order to achieve smart card platform independence from Part 1. Thus, non-government smart card programs can readily adopt the interface specifications in Parts 2 and 3 while customizing Part 1 to their own data model, data types, and namespaces.

5.1 Key References

A key reference is a one-byte identifier that specifies a cryptographic key according to its PIV Key Type. SP 800-78, Table 6-1, defines the key reference values that shall be used on the PIV interfaces. The key reference values are used in a cryptographic protocol such as an authentication or a signing protocol. Key references are only assigned to private and secret (symmetric) keys. All other PIV Card Application key reference values are reserved for future use.

Table 3. PIV Card Application Authentication and Key References

Key Reference Value	PIV Key Type	Authenticatable Entity / Administrator	Security Condition for Use	Retry Reset Value	Number of Unblocks
'00'	Global PIN	Cardholder	Always	Platform Specific	Platform Specific
'80'	Application PIN	Cardholder	Always	Issuer Specific	Issuer Specific
'81'	PIN Unblock Key	PIV Card Application Administrator	Always	Issuer Specific	Issuer Specific
See Table 6-1 in SP 800-78	*PIV Authentication Key*	PIV Card Application Administrator	PIN	N/A	N/A
See Table 6-1 in SP 800-78	*Card Management Key*[8]	PIV Card Application Administrator	Always	N/A	N/A
See Table 6-1 in SP 800-78	*Digital Signature Key*	PIV Card Application Administrator	PIN Always	N/A	N/A
See Table 6-1 in SP 800-78	*Key Management Key*	PIV Card Application Administrator	PIN	N/A	N/A
See Table 6-1 in SP 800-78	*Card Authentication Key*	PIV Card Application Administrator	Always	N/A	N/A

[8] Note: The Card Management key is the PIV Card Application Administration Key used for managing the PIV card application.

When represented as a byte, the key reference occupies bits b8 and b5-b1, while b7 and b6 shall be set to 0. If b8 is 0 then the key reference names global reference data. If b8 is 1, then the key reference names application-specific reference data.

The access control rules for PIV data object access shall reference the PIV Card Application PIN and may optionally reference the cardholder Global PIN. If the Global PIN is used by the PIV Card Application, then the Global PIN format shall follow the PIV Card Application PIN format defined in section 2.4.3 of Part 2.

PIV Card Applications with the discovery object, and the first byte of the PIN Usage Policy value set to 0x60 as per section 3.2.6, shall reference the PIV Card Application PIN as well as the cardholder Global PIN in the access control rules for PIV data object access. Additionally, the PIV Card Application card commands can change the status of the Global PIN, and may change its reference data while the PIV Card Application is the currently selected application.

Note: The rest of the document uses “PIN” to mean either the PIV Application PIN or the Global PIN.

5.2 PIV Algorithm Identifier

A PIV algorithm identifier shall be a one-byte identifier of a cryptographic algorithm. The identifier specifies a cryptographic algorithm and key size. For symmetric cryptographic operations, the algorithm identifier also specifies a mode of operation (i.e., CBC or ECB). SP 800-78, Table 6-2 lists the PIV algorithm identifiers for the cryptographic algorithms that may be recognized on the PIV interfaces.

5.3 Cryptographic Mechanism Identifiers

Cryptographic Mechanism Identifiers are defined in Table 4. These identifiers serve as data field inputs to the SP 800-73-2 Part 2 GENERATE ASYMMETRIC KEY PAIR card command and the SP 800-73-2 Part 3 pivGenerateKeyPair() client API function call, which initiates the generation and storing of the asymmetric key pair.

Table 4. Cryptographic Mechanism Identifiers

Cryptographic Mechanism Identifier	Description	Parameter
'00'-'05'	RFU	
See Table 6-2 in SP 800-78	RSA 1024	Optional public exponent encoded big-endian
See Table 6-2 in SP 800-78	RSA 2048	Optional public exponent encoded big-endian
'08'-'10'	RFU	
See Table 6-2 in SP 800-78	ECC: Curve P-256	None
'12'-'13'	RFU	
See Table 6-2 in SP 800-78	ECC: Curve P-384	None

All other cryptographic mechanism identifier values are reserved for future use.

5.4 Status Words

A Status Word (SW) shall be a 2-byte value returned by an entry point on the client-application programming interface or a card command at the card edge. The first byte of a status word is referred to as SW1 and the second byte of a status word is referred to as SW2.

Recognized values of all SW1-SW2 pairs used as return values on both the client-application programming and card command interfaces and their interpretation are given in Table 5. The description of individual client-application programming interface entry points or card commands provide additional information for interpreting returned status words.

Table 5. Status Words

SW1	SW2	Meaning
'61'	'xx'	Successful execution where SW2 encodes the number of response data bytes still available
'63'	'CX'	Verification failed, X indicates the number of further allowed retries or resets
'69'	'82'	Security condition not satisfied
'69'	'83'	Authentication method blocked
'6A'	'80'	Incorrect parameter in command data field
'6A'	'81'	Function not supported
'6A'	'84'	Not enough memory
'6A'	'86'	Incorrect parameter in P1 or P2
'6A'	'88'	Referenced data or reference data not found
'90'	'00'	Successful execution

Appendix A—PIV Data Model

The PIV data model number is 0x10, and the data model version number is 0x01.

The SP800-73-2 End-Point specification does not provide mechanisms to read partial contents of a PIV data object. Individual access to the TLV elements within a container is not supported. For each container, End-Point compliant cards shall return all TLV elements of the container in the order listed in this Appendix.

Both single-chip/dual-interface and dual-chip implementations shall be feasible. In the single-chip/dual-interface configuration, the PIV Card Application shall be provided the information regarding which interface is in use. In the dual-chip configuration, a separate PIV Card Application shall be loaded on each chip.

Table 6. PIV Data Containers

Container Description	Container ID	BER-TLV Tag	Container Minimum Capacity (Bytes)*	Access Rule for Read	Contact / Contactless	M/O
Card Capability Container	0xDB00	'5FC107'	297	Always	Contact	M
Cardholder Unique Identifier	0x3000	'5FC102'	3414	Always	Contact and Contactless	M
X.509 Certificate for PIV Authentication	0x0101	'5FC105'	2005	Always	Contact	M
Cardholder Fingerprints	0x6010	'5FC103'	4006	PIN	Contact	M
Security Object	0x9000	'5FC106'	1031	Always	Contact	M
Cardholder Facial Image	0x6030	'5FC108'	12710	PIN	Contact	O
Printed Information	0x3001	'5FC109'	164	PIN	Contact	O
X.509 Certificate for Digital Signature	0x0100	'5FC10A'	2005	Always	Contact	O
X.509 Certificate for Key Management	0x0102	'5FC10B'	2005	Always	Contact	O
X.509 Certificate for Card Authentication	0x0500	'5FC101'	2005	Always	Contact and Contactless	O
Discovery Object	0x6050	'7E'	20	Always	Contact and Contactless	O

* The values in this column denote the guaranteed minimum capacities, in bytes, of the on-card storage containers. Cards may be produced and determined conformant with larger containers.

Note that all data elements of the following data objects are mandatory unless specified as optional.

Table 7. Card Capability Container

Card Capability Container		0xDB00	
Data Element (TLV)	**Tag**	**Type**	**Max. Bytes***
Card Identifier	0xF0	Fixed	21
Capability Container version number	0xF1	Fixed	1
Capability Grammar version number	0xF2	Fixed	1
Applications CardURL	0xF3	Variable	128
PKCS#15	0xF4	Fixed	1
Registered Data Model number	0xF5	Fixed	1
Access Control Rule Table	0xF6	Fixed	17
Card APDUs	0xF7	Fixed	0
Redirection Tag	0xFA	Fixed	0
Capability Tuples (CTs)	0xFB	Fixed	0
Status Tuples (STs)	0xFC	Fixed	0
Next CCC	0xFD	Fixed	0
Extended Application CardURL (optional)	0xE3	Fixed	48
Security Object Buffer (optional)	0xB4	Fixed	48
Error Detection Code	0xFE	LRC	0

Table 8. Cardholder Unique Identifier

Cardholder Unique Identifier		0x3000	
Data Element (TLV)	**Tag**	**Type**	**Max. Bytes***
Buffer Length (Optional)	0xEE	Fixed	2
FASC-N	0x30	Fixed Text	25
Organization Identifier (Optional)	0x32	Fixed	4
DUNS (Optional)	0x33	Fixed	9
GUID	0x34	Fixed Numeric	16
Expiration Date	0x35	Date (YYYYMMDD)	8
Authentication Key Map (Optional)	0x3D	Variable	512
Issuer Asymmetric Signature	0x3E	Variable	2816**
Error Detection Code	0xFE	LRC	0

The Error Detection Code is the same element as the Longitudinal Redundancy Code (LRC) in TIG SCEPACS. Because TIG SCEPACS makes the LRC mandatory, it is present in the CHUID. However, this document makes no use of the Error Detection Code, and therefore the length of the TLV value is set to 0 bytes, (i.e., no value will be supplied).

* The values in the “Max. Bytes” columns denote the lengths of the value (V) fields of BER-TLV elements.

** Recommended length: The signer certificate may cause the “Max. Bytes” value in the Issuer Asymmetric Signature field to be exceeded.

Table 9. X.509 Certificate for PIV Authentication

X.509 Certificate for PIV Authentication		0x0101	
Data Element (TLV)	**Tag**	**Type**	**Max. Bytes***
Certificate	0x70	Variable	1856**
CertInfo	0x71	Fixed	1
MSCUID (Optional)	0x72	Variable	38
Error Detection Code	0xFE	LRC	0

Table 10. Cardholder Fingerprints

Cardholder Fingerprints		0x6010	
Data Element (TLV)	**Tag**	**Type**	**Max. Bytes***
Fingerprint I & II	0xBC	Variable	4000***
Error Detection Code	0xFE	LRC	0

Table 11. Security Object

Security Object		0x9000	
Data Element (TLV)	**Tag**	**Type**	**Max. Bytes***
Mapping of DG to ContainerID	0xBA	Variable	100
Security Object	0xBB	Variable	900
Error Detection Code	0xFE	LRC	0

Table 12. Cardholder Facial Image

Cardholder Facial Image		0x6030	
Data Element (TLV)	**Tag**	**Type**	**Max. Bytes***
Image for Visual Verification	0xBC	Variable	12704****
Error Detection Code	0xFE	LRC	0

* The values in the “Max. Bytes” columns denote the lengths of the value (V) fields of BER-TLV elements.

** Recommended length. Certificate size can exceed indicated length value.

*** Recommended length. The certificate that signed the Fingerprint I and II data element in the Cardholder Fingerprint data object can either be stored in the CHUID or in the Fingerprint I and II data element itself. If the latter, the “Max. Bytes” value quoted is a recommendation and the signer certificate in CBEFF_SIGNATURE_BLOCK can exceed the “Max. bytes”.

****Recommended length. The certificate that signed the Facial Image data element (tag 0xBC) can be stored in the CHUID or in the Facial Image data object itself. If the latter, the “Max. Bytes” value quoted is a recommendation and the signer certificate in CBEFF_SIGNATURE_BLOCK can exceed the “Max. bytes”.

Table 13. Printed Information

Printed Information		0x3001	
Data Element (TLV)	**Tag**	**Type**	**Max. Bytes***
Name	0x01	Fixed Text	32
Employee Affiliation (Line 1)	0x02	Fixed Text	20
Employee Affiliation (Line 2)	0x03	Fixed Text	20
Expiration date	0x04	Date (YYYYMMMDD)	9
Agency Card Serial Number	0x05	Fixed Text	10
Issuer Identification	0x06	Fixed Text	15
Organization Affiliation (Line 1) (Optional)	0x07	Fixed Text	20
Organization Affiliation (Line 2) (Optional)	0x08	Fixed Text	20
Error Detection Code	0xFE	LRC	0

Note: The Organization Affiliation fields (tags 0x07 and 0x08) are new optional data elements in the Printed Information data object. Employee Affiliation Line 2 (tag 0x03) is deprecated and will be eliminated in a future revision, as it does not have a corresponding text field on the face of the card. In order to successfully match the printed information for verification on Zone 8 (Employee Affiliation) and Zone 10 (Organization Affiliation) on the face of the card with the printed information represented stored electronically on card, agencies should use tags 0x02, 0x07 and 0x08.

Table 14. X.509 Certificate for Digital Signature

X.509 Certificate for Digital Signature		0x0100	
Data Element (TLV)	**Tag**	**Type**	**Max. Bytes***
Certificate	0x70	Variable	1856**
CertiInfo	0x71	Fixed	1
MSCUID (Optional)	0x72	Variable	38
Error Detection Code	0xFE	LRC	0

Table 15. X.509 Certificate for Key Management

X.509 Certificate for Key Management		0x0102	
Data Element (TLV)	**Tag**	**Type**	**Max. Bytes***
Certificate	0x70	Variable	1856**
CertiInfo	0x71	Fixed	1
MSCUID (Optional)	0x72	Variable	38
Error Detection Code	0xFE	LRC	0

* The values in the "Max. Bytes" columns denote the lengths of the value (V) fields of BER-TLV elements.
** Recommended length. Certificate size can exceed indicated length value.

Table 16. X.509 Certificate for Card Authentication

X.509 Certificate for Card Authentication		0x0500	
Data Element (TLV)	**Tag**	**Type**	**Max. Bytes***
Certificate	0x70	Variable	1856**
CertiInfo	0x71	Fixed	1
MSCUID (Optional)	0x72	Variable	38
Error Detection Code	0xFE	LRC	0

Table 17. Discovery Object

Discovery Object (Tag '7E')		0x6050	
Data Element (TLV)	**Tag**	**Length**	**Max. Bytes***
PIV Card Application AID	0x4F	Fixed	12
PIN Usage Policy	0x5F2F	Fixed	3

The CertInfo byte in certificates identified above shall be encoded as follows:

```
CertInfo::= BIT STRING {
                    CompressionTypeMsb(0),  // 0 = no compression and 1 = gzip[10]
                                            // compression.
                    CompressionTypeLsb(1),  // shall be set to '0' for PIV Applications
                    IsX509(2),              // shall be set to '0' for PIV Applications
                    RFU3(3),
                    RFU4(4),
                    RFU5(5),
                    RFU6(6),
                    RFU7(7)
                    }
```

* The values in the "Max. Bytes" columns denote the lengths of the value (V) fields of BER-TLV elements.

** Recommended length. Certificate size can exceed indicated length value.

[10] Gzip formats are specified in RFC 1951 and RFC 1952

Appendix B—PIV Authentication Mechanisms

To provide guidelines on the usage and behavior supported by the PIV Card, PIV authentication mechanisms and application scenarios are described in this section. FIPS 201 describes PIV authentication as the "process of establishing confidence in the identity of the cardholder presenting a PIV Card." The fundamental goal of using the PIV Card is to authenticate the identity of the cardholder to a system or person that is controlling access to a protected resource or facility. This end goal may be reached by various combinations of one or more of the validation steps described below:

Card Validation (CardV) — This is the process of verifying that a PIV Card is authentic (i.e., not a counterfeit card). Card validation mechanisms include:

+ Visual inspection of the tamper-proofing and tamper-resistant features of the PIV Card as per Section 4.1.2 of FIPS 201,
+ Use of cryptographic challenge-response schemes with symmetric keys,
+ Use of asymmetric authentication schemes to validate private keys embedded within the PIV Card.

Credential Validation (CredV) — This is the process of verifying the various types of credentials (such as visual credentials, CHUID, biometrics, PIV keys and certificates) held by the PIV Card. Credential validation mechanisms include:

+ Visual inspection of PIV Card visual elements (such as the photo, the printed name, and rank, if present),
+ Verification of certificates on the PIV Card,
+ Verification of signatures on the PIV biometrics and the CHUID,
+ Checking the expiration date,
+ Checking the revocation status of the credentials on the PIV Card.

Cardholder Validation (HolderV) — This is the process of establishing that the PIV Card is in the possession of the individual to whom the card has been issued. Classically, identity authentication is achieved using one or more of these factors: a) something you have, b) something you know, and c) something you are. The assurance of the authentication process increases with the number of factors used. In the case of the PIV Card, these three factors translate as follows: a) something you have – possession of a PIV Card, b) something you know – knowledge of the PIN, and c) something you are – the visual characteristics of the cardholder, and the live fingerprint samples provided by the cardholder. Thus, mechanisms for PIV cardholder validation include:

+ Presentation of a PIV Card by the cardholder,
+ Matching the visual characteristics of the cardholder with the photo on the PIV Card,
+ Matching the PIN provided with the PIN on the PIV Card,

+ Matching the live fingerprint samples provided by the cardholder, with the biometric information embedded within the PIV Card.

B.1 Authentication Mechanism Diagrams

This section describes the activities and interactions involved in interoperable usage and authentication of the PIV Card. The authentication mechanisms represent how a relying party will authenticate the cardholder (regardless of which agency issued the card) in order to provide access to its systems or facilities. These activities and interactions are represented in functional authentication mechanism diagrams. These diagrams are not intended to provide syntactical commands or API function names.

Each of the PIV authentication mechanisms described in this section can be broken into a sequence of one or more validation steps where Card, Credential, and Cardholder validation is performed. In the illustrations, the validation steps are marked as CardV, CredV and HolderV to signify Card, Credential, and Cardholder validation respectively.

Depending upon the assurance provided by the actual sequence of validation steps in a given PIV authentication mechanism, relying parties can make appropriate decisions for granting access to protected resources based on a risk analysis.

B.1.1 Authentication using PIV Visual Credentials

This is the authentication mechanism where a human guard authenticates the cardholder using the visual credentials held by the PIV Card, and is illustrated in Figure B-1.

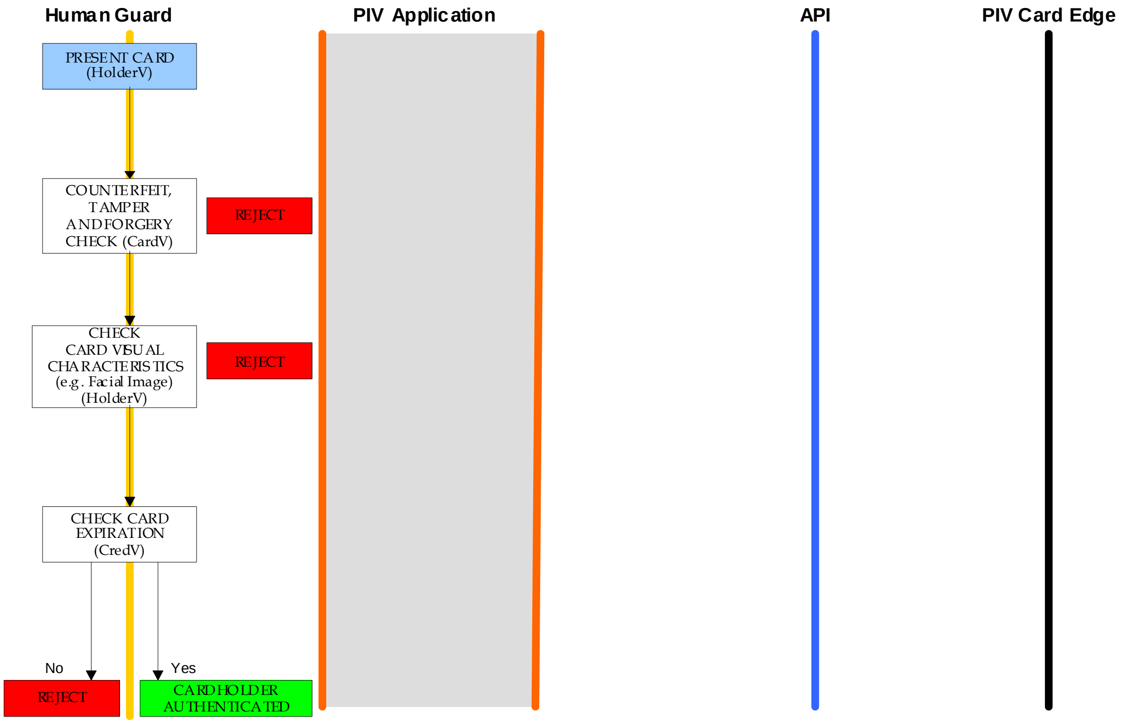
Human Guard
PIV Application
API
PIV Card Edge
PRESENT CARD (HolderV)
COUNTERFEIT, TAMPER AND FORGERY CHECK (CardV)
REJECT
CHECK CARD VISUAL CHARACTERISTICS (e.g. Facial Image) (HolderV)
REJECT
CHECK CARD EXPIRATION (CredV)
No
Yes
REJECT
CARDHOLDER AUTHENTICATED

Figure B-1. Authentication using PIV Visual Credentials

B.1.2 Authentication using PIV *CHUID*

The PIV CHUID may be used for authentication in several variations. The use of the PIV Card to implement the CHUID authentication mechanism is illustrated in Figure B-2. The minimum set of data that must be transmitted from the PIV Application on the Local System to the host is application dependent and therefore not defined in this Specification.

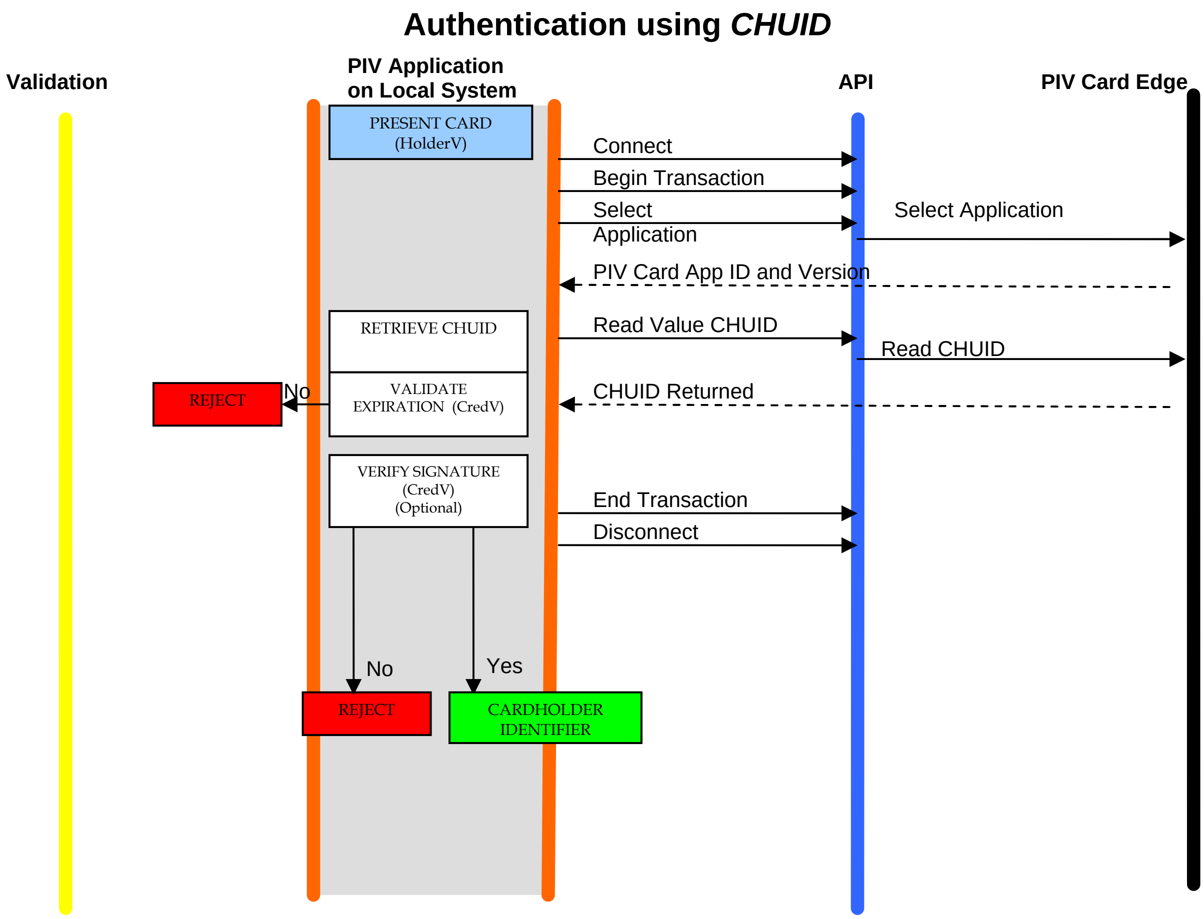

Figure B-2. Authentication using PIV *CHUID*

B.1.3 Authentication using PIV *Biometrics (BIO)*

The general authentication mechanism using the PIV biometrics is illustrated in Figure B-3.

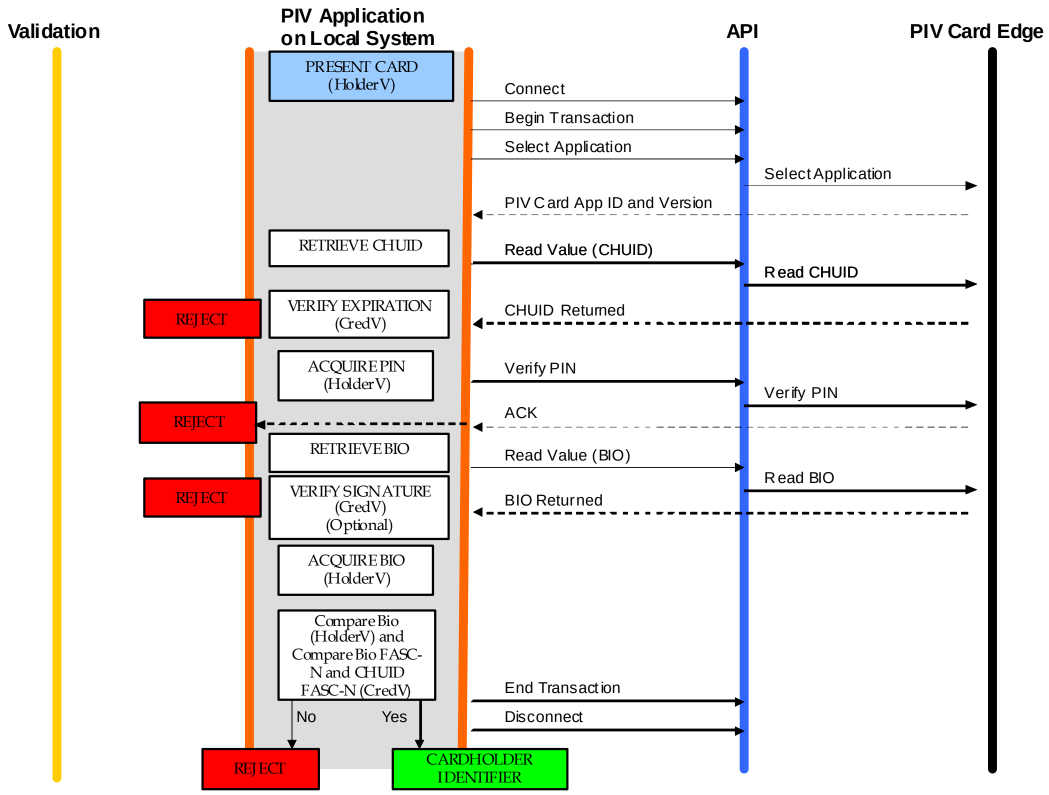

Figure B-3. Authentication using *PIV Biometrics (BIO)*

The assurance of authentication using the *PIV biometric* can be further increased if the live biometric sample is collected in an attended environment, with a human overseeing the process. The attended biometric authentication mechanism (BIO-A) is illustrated in Figure B-4.

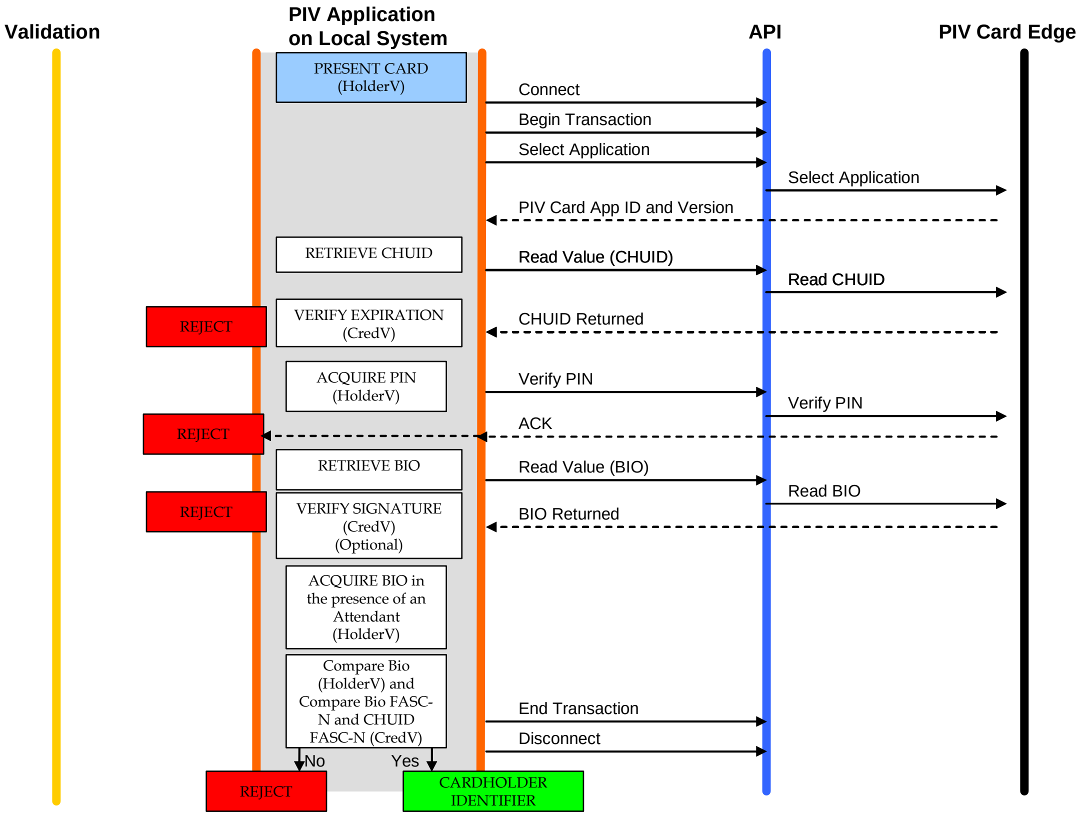

Figure B-4. Authentication using *PIV Biometrics Attended (BIO-A)*

B.1.4 Authentication using PIV Authentication Key

The authentication mechanism using the *PIV Authentication Key* is illustrated in Figure B-5.

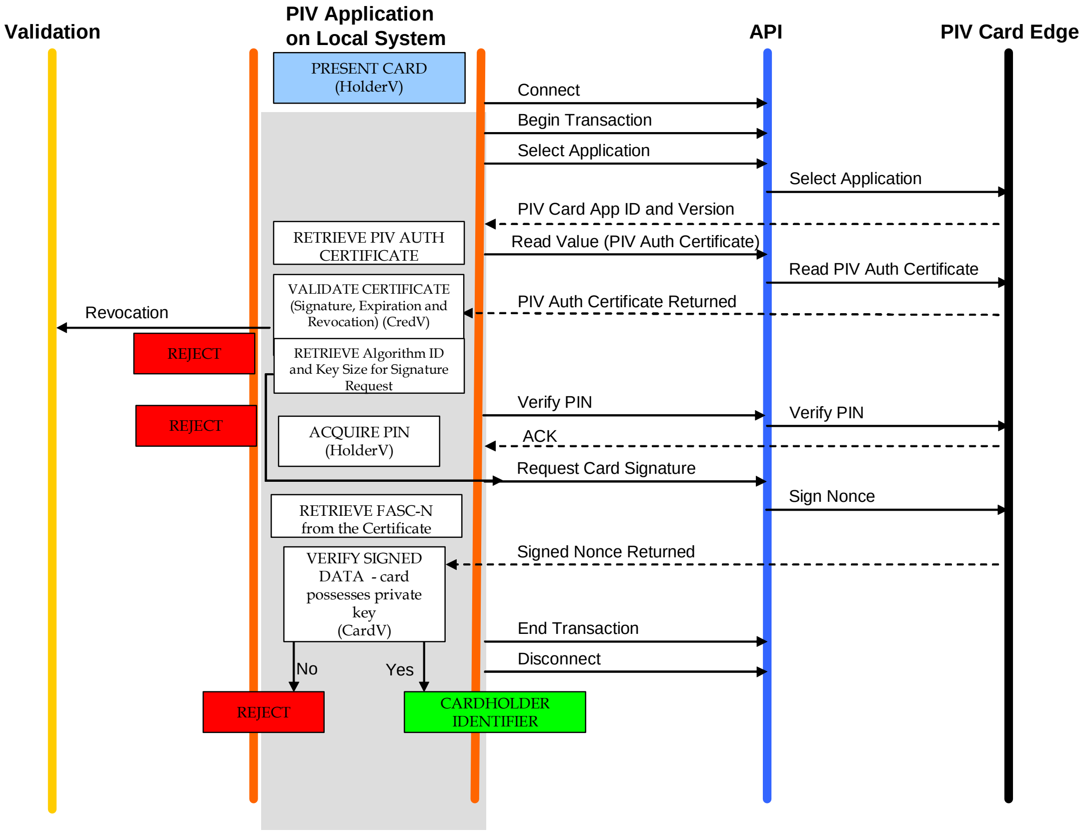

Figure B-5. Authentication using *PIV Authentication Key*

B.1.5 Authentication using Card Authentication Key

Authentication mechanisms using the *Card Authentication Key* are illustrated in Figures B-6 and B-7. Figure B-6 illustrates the use of an asymmetric *Card Authentication Key*, while figure B-7 uses a symmetric *Card Authentication* Key for the authentication mechanism. Both mechanisms provide "SOME" confidence in the assurance of the identity.

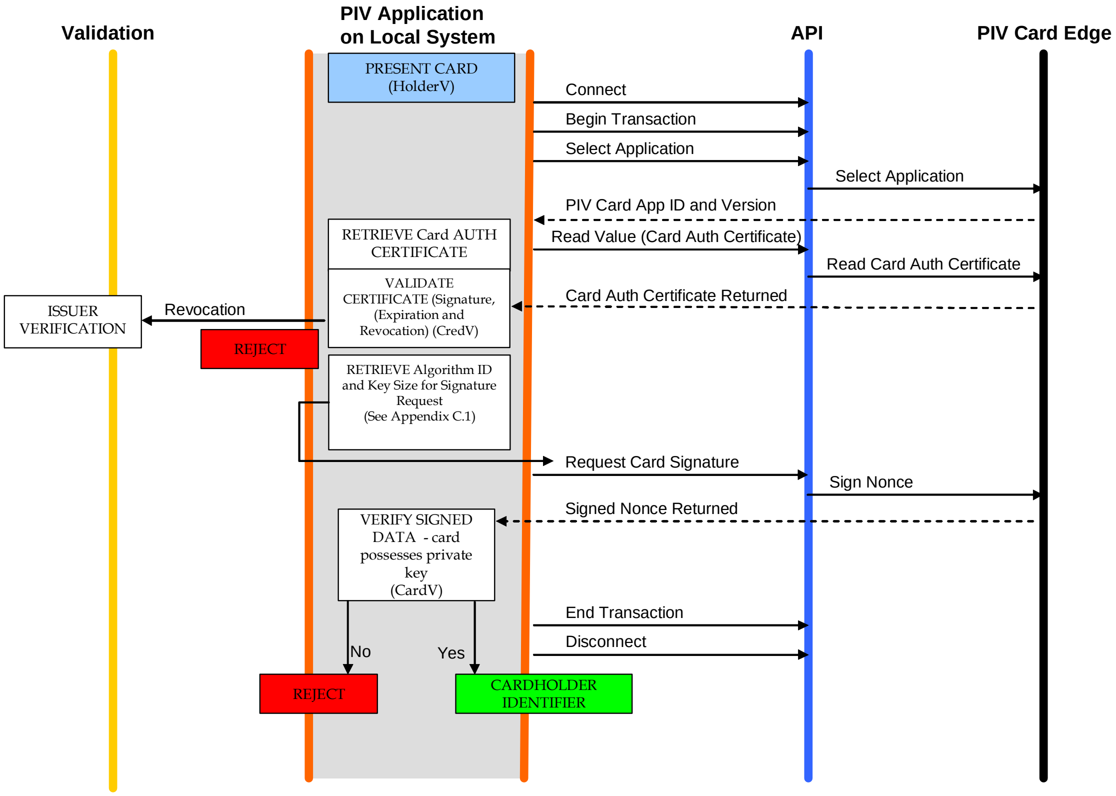
Validation
PIV Application on Local System
API
PIV Card Edge
PRESENT CARD (HolderV)
Connect
Begin Transaction
Select Application
Select Application
PIV Card App ID and Version
RETRIEVE Card AUTH CERTIFICATE
Read Value (Card Auth Certificate)
Read Card Auth Certificate
ISSUER VERIFICATION
Revocation
VALIDATE CERTIFICATE (Signature, (Expiration and Revocation) (CredV)
Card Auth Certificate Returned
REJECT
RETRIEVE Algorithm ID and Key Size for Signature Request (See Appendix C.1)
Request Card Signature
Sign Nonce
VERIFY SIGNED DATA - card possesses private key (CardV)
Signed Nonce Returned
End Transaction
Disconnect
No
Yes
REJECT
CARDHOLDER IDENTIFIER

Figure B-6. Authentication using an asymmetric *Card Authentication Key*

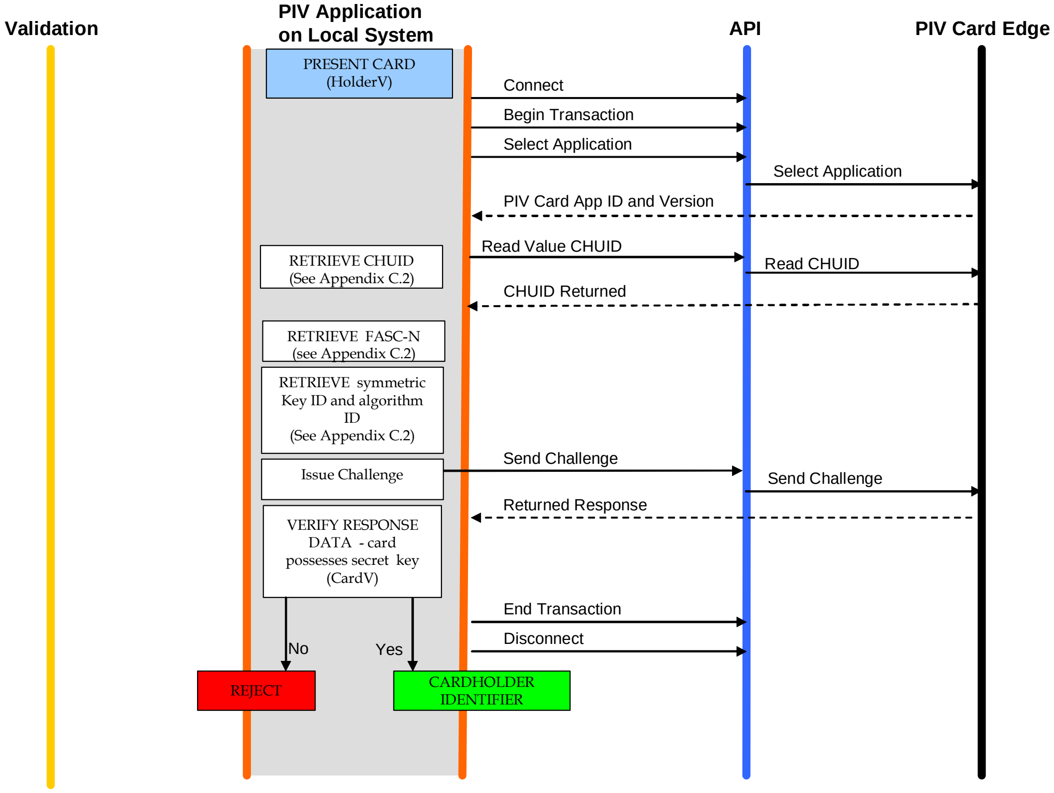

Figure B-7. Authentication using a symmetric *Card Authentication Key*

B.2 Summary Table

The following table summarizes the types of validation activities that are included in each of the PIV authentication mechanisms described earlier in this section.

Table 18. Summary of PIV Authentication Mechanisms

PIV Authentication Mechanism	Card Validation Steps (CardV)	Credential Validation Steps (CredV)	Cardholder Validation Steps (HolderV)
PIV Visual Authentication	Counterfeit, tamper and forgery check	Expiration check	Possession of Card Match of card visual characteristics with cardholder
PIV CHUID		Expiration check CHUID signature check (optional)	Possession of Card
Symmetric *Card Authentication Key*	Perform challenge and response with a PIV symmetric key		Possession of Card
Asymmetric *Card Authentication Key*	Perform challenge and response with a PIV asymmetric Card Authentication key, and validate signature on response	Card expiration check Certificate validation of a PIV certificate	Possession of Card
PIV Authentication Key	Perform challenge and response with a PIV asymmetric key, and validate signature on response	Card expiration check Certificate validation of a PIV certificate	Possession of Card Match PIN provided by Cardholder
PIV Biometric		Expiration check CHUID signature check (optional) PIV Bio signature check (optional) Match CHUID FASC-N with PIV Bio FASC-N	Possession of Card Match PIN provided by Cardholder Match Cardholder bio with PIV bio
PIV Biometric (Attended)		Expiration check CHUID signature check (optional) PIV Bio signature check (optional) Match CHUID FASC-N with PIV Bio FASC-N	Possession of Card Match PIN provided by Cardholder Match of Cardholder bio to PIV bio *in view of attendant*

Appendix C—PIV Algorithm Identifier Discovery

Relying Parties interact with many PIV Cards with the same native key-type implemented by different key sizes and algorithms[11]. For example, a relying party performing the authentication mechanism described in B.1.4 (Authentication using the *PIV Authentication Key*), can expect to perform a challenge and response cryptographic authentication with 1) a PIV Card with RSA 1024 bit *PIV Authentication Key,* 2) a PIV Card with RSA 2048 bit *PIV Authentication Key* or 3) a PIV Card with an elliptic curve key (P-256) *PIV Authentication Key*.

This appendix describes recommended procedures for key size and algorithm discovery (PIV algorithm ID discovery) to facilitate cryptographic authentication initiated by the relying party to make appropriate decisions for granting access to logical networks and systems as well as physical access control systems. The discovery procedure is defined in terms of asymmetric and symmetric cryptographic authentication.

C.1 PIV Algorithm Identifier Discovery for Asymmetric Cryptographic Authentication

As illustrated in the authentication mechanisms in Appendix B, an asymmetric cryptographic authentication involves issuing a challenge (request to sign a nonce) to the PIV Card. The relying party issuing the command provides the nonce to be signed, the key reference, and the PIV algorithm identifier as parameters of the command. The nonce is random data generated by the relying party and the key reference is known. The PIV algorithm identifier, on the other hand, is unknown to the relying party and needs to be identified in order to issue the challenge command. The PIV algorithm identifier can be derived from the previous steps of the authentication mechanism. The relying party, prior to the challenge command, retrieved and parsed the X.509 certificate from the card in order to 1) optionally validate the certificate and 2) extract the public key for the pending decryption and matching of the signed nonce once returned from the card. It is during the parsing of the X.509 certificate that the PIV algorithm identifier can be identified in two steps[12]:

Step 1: Algorithm Type Discovery:
The X.509 certificate stores the public key in the SubjectPublicKeyInfo field. The same field also stores the X.509 AlgorithmIdentifier object identifiers (OIDs). This OID identifies the algorithm (RSA, or ECC) as listed in table 3-5 of SP 800-78.

Step 2: Key Size Discovery:[13]
The public key of the certificate holder is stored in the X.509 SubjectPublicKeyInfo field. By reading the modulus n bit string, in case of a RSA key, or the Curve Point string, in case of an elliptic curve public key, the corresponding private key size is implicitly known since both public and private keys are of the same length.

[11] Table 3.1 , SP 800-78 list the various PIV algorithm identifiers to choose one for each PIV key type

[12] The PIV algorithm identifiers specify both the key and the algorithm for the key references, Thus both values have to be discovered in order to derive the PIV algorithm identifier

[13] If the AlgorithmIdentifier OID indicates an elliptic curve algorithm and its EcpkParameters does not indicate implicit inherited from the issuer's certificate, then the namedCurve field in the EcpkParameters encodes the curve as per table 3.6 of SP 800-78. The associated named curve, indicates the key size x of curve P-xxx. This is an alternative method to discover the key size for an elliptic curve keys.

As a final step, the discovered X.509 algorithm OID and key size are mapped to the PIV Algorithm Identifiers as defined in SP 800-78 table 6-2. The relying party then proceeds to issue the general authenticate command to the card.

C.2 PIV Algorithm Identifier Discovery for Symmetric Cryptographic Authentication

In the absence of a X.509 certificate, as is the case with symmetric cryptography, the PIV algorithm identifier discovery mechanism has to rely on a lookup table residing at the local system. The table maps a unique card identifier and key reference (inputs) to an associated PIV algorithm identifier (output). The unique identifier supplied by the card shall be Agency Code || System Code || Credential Number of the FASC-N.

The optional *card authentication key* can be a symmetric key or an asymmetric key. A relying party has no prior knowledge of 1) the key's existence and 2) the key's symmetric or asymmetric implementation. The following routine discovers the *Card Authentication Key's* native implementation:

1) Attempt to read the X.509 Certificate for Card Authentication.
   + If the first step succeeds, the *Card Authentication Key* is asymmetric. The asymmetric PIV algorithm identifier discovery (C.1) mechanism should be followed.

   + If the first step fails, the *Card Authentication Key* a) does not exist or b) is a symmetric key.

2) Read the CHUID and extract the Agency Code || System code || Credential Number from the CHUID's FASC-N.

3) Attempt to retrieve the PIV algorithm identifier from the local lookup table.
   + If a valid PIV algorithm identifier is returned, the *Card Authentication Key* is symmetric.

   + If no algorithm identifier is returned, the PIV Card does not implement the key.

Appendix D—Terms, Acronyms, and Notation

D.1 Terms

Algorithm Identifier
A PIV algorithm identifier is a one-byte identifier that specifies a cryptographic algorithm and key size. For symmetric cryptographic operations, the algorithm identifier also specifies a mode of operation (i.e., CBC or ECB).

Application Identifier
A globally unique identifier of a card application as defined in ISO/IEC 7816-4.

Application Session
The period of time within a card session between when a card application is selected and a different card application is selected or the card session ends.

Authenticatable Entity
An entity that can successfully participate in an authentication protocol with a card application.

BER-TLV Data Object
A data object coded according to ISO/IEC 8825-2.

Card
An integrated circuit card.

Card Application
A set of data objects and card commands that can be selected using an application identifier.

Client Application
A computer program running on a computer in communication with a card interface device.

Data Object
An item of information seen at the card command interface for which are specified a name, a description of logical content, a format and a coding.

Interface Device
Synonym for card interface device.

Key Reference
A key reference is a one-byte identifier that specifies a cryptographic key according to its PIV Key Type. The identifier is part of cryptographic material used in a cryptographic protocol such as an authentication or a signing protocol.

MSCUID
An optional legacy identifier included for compatibility with Common Access Card and Government Smart Card Interoperability Specifications.

Object Identifier
A globally unique identifier of a data object as defined in ISO/IEC 8824-2.

PIV Key Type
A type of a Key. The PIV Key Types are 1) PIV Authentication Key, 2) PIV Card Authentication Key, 3) PIV Digital Signature Key, 4) The PIV Key Management Key and 5) The Card Application Administration Key.

Relying Party
An entity that relies upon the subscriber’s credentials, typically to process a transaction or grant access to information or a system.

Status Word Two bytes returned by an integrated circuit card after processing any command that signify the success of or errors encountered during said processing.

D.2 Acronyms

ACR	Access Control Rule
AID	Application Identifier
APDU	Application Protocol Data Unit
API	Application Programming Interface
ASN.1	Abstract Syntax Notation
BER	Basic Encoding Rules
CBC	Cipher Block Chaining
CBEFF	Common Biometric Exchange Formats Framework
CCC	Card Capability Container
CHUID	Cardholder Unique Identifier
DER	Distinguished Encoding Rules
DES	Data Encryption Standard
ECB	Electronic Code Book
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
FASC-N	Federal Agency Smart Credential Number
FIPS	Federal Information Processing Standards
FISMA	Federal Information Security Management Act
GSC-IAB	Government Smart Card Interagency Advisory Board
GSC-IS	Government Smart Card Interoperability Specification
GUID	Global Unique Identification Number
GSC-IAB	Government Smart Card Interagency Advisory Board
HSPD	Homeland Security Presidential Directive

ICC	Integrated Circuit Card
IEC	International Electrotechnical Commission
INCITS	InterNational Committee for Information Technology Standards
ISO	International Standards Organization
ITL	Information Technology Laboratory
LSB	Least Significant Bit
LRC	Longitudinal Redundancy Code
MRTD	Machine Readable Travel Document
MSB	Most Significant Bit
NIST	National Institute of Standards and Technology
OID	Object Identifier
OMB	Office of Management and Budget
PACS	Physical Access Control System
PIN	Personal Identification Number
PI	Person Identifier, a field in the FASC-N
PIV	Personal Identity Verification
PIX	Proprietary Identifier Extension
PKCS	Public Key Cryptography Standard
PKI	Public Key Infrastructure
PUK	PIN Unblocking Key
RFU	Reserved for Future Use
RID	Registered application provider IDentifier
RSA	Rivest, Shamir, Aldeman
SCEPACS	Smart Card Enabled Physical Access Control System
SCP	ETSI Smart Card Project
SP	Special Publication

SW1	First byte of a two-byte status word
SW2	Second byte of a two-byte status word
TIG	Technical Implementation Guidance
TLV	Tag-Length-Value
URL	Uniform Resource Locator

D.3 Notation

The sixteen hexadecimal digits shall be denoted using the alphanumeric characters 0, 1, 2…, A, B, C, D, E, and F. A byte consists of two hexadecimal digits, for example, '2D'. A sequence of bytes may be enclosed in single quotation marks, for example 'A0 00 00 01 16' rather than given as a sequence of individual bytes, 'A0' '00' '00' '01' '16'.

A byte can also be represented by bits b8 to b1, where b8 is the most significant bit (MSB) and b1 is the least significant bit (LSB) of the byte. In textual or graphic representations, the leftmost bit is the MSB. Thus, for example, the most significant bit, b8, of '80' is 1 and the least significant bit, b1, is 0.

All bytes specified as RFU shall be set to '00' and all bits specified as reserved for future use shall be set to 0.

All lengths shall be measured in number of bytes unless otherwise noted.

Data objects in templates are described as being mandatory (M), optional (O) or conditional (C). 'Mandatory' means the data object shall appear in the template. 'Optional' means the data object may appear in the template. In the case of 'conditional' data objects, the conditions under which they are required are provided in a footnote to the table.

In other tables the M/O column identifies properties of the PIV Card Application that shall be present (M) or may be present (O).

BER-TLV data object tags are represented as byte sequences as described above. Thus, for example, '4F' is the interindustry data object tag for an application identifier and '7F 60' is the interindustry data object tag for the biometric information template.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this standard are to be interpreted as described in IETF RFC 2119, Key Words for Use in RFCs to Indicate Requirement Levels [10].

Appendix E—References

[1] Federal Information Processing Standard 201-1, Change Notice 1, *Personal Identity Verification (PIV) of Federal Employees and Contractors*, March 2006. (See http://csrc.nist.gov)

[2] ISO/IEC 7816 (Parts 4, 5, 6, 8, and 9), *Information technology — Identification cards — Integrated circuit(s) cards with contacts.*

[3] *Government Smart Card Interoperability Specification, Version 2.1*, NIST Interagency Report 6887 – 2003 Edition, July 16, 2003.

[4] PACS v2.2, *Technical Implementation Guidance: Smart Card Enabled Physical Access Control Systems*, Version 2.2, The Government Smart Card Interagency Advisory Board's Physical Security Interagency Interoperability Working Group, July 27, 2004. (see http://www.smart.gov/information/TIG_SCEPACS_v2.2.pdf)

[5] NIST Special Publication 800-87, *Codes for the Identification of Federal and Federally-Assisted Organizations*, April 2008. (See http://csrc.nist.gov)

[6] *PKI for Machine Readable Travel Documents Offering ICC Read-Only Access Version - 1.1* Date - October 01, 2004. Published by authority of the Secretary General, International Civil Aviation Organization.

[7] ISO/IEC 8824-2:2002, *Information technology -- Abstract Syntax Notation One (ASN.1): Information object specification.*

[8] ISO/IEC 8825-1:2002, *Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).*

[9] NIST Special Publication 800-78-1, *Cryptographic Algorithms and Key Sizes for Personal Identity Verification*, May 2007. (See http://csrc.nist.gov)

[10] IETF RFC 2119, "Key Words for Use in RFCs to Indicate Requirement Levels," March, 1997.

NIST Special Publication 800-73-2

Interfaces for Personal Identity Verification – Part 2: End-Point PIV Card Application Card Command Interface

National Institute of Standards and Technology
Technology Administration
U.S. Department of Commerce

Ramaswamy Chandramouli
James F. Dray
Hildegard Ferraiolo
Scott B. Guthery
William MacGregor
Ketan Mehta

I N F O R M A T I O N   S E C U R I T Y

Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD, 20899-8930

September 2008

U.S. Department of Commerce
Carlos M. Gutierrez, Secretary

National Institute of Standards and Technology
James M. Turner, Deputy Director

Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the Nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analyses to advance the development and productive use of information technology. ITL's responsibilities include the development of management, administrative, technical, and physical standards and guidelines for the cost-effective security and privacy of non-national security-related information in Federal information systems. This special publication 800-series reports on ITL's research, guidelines, and outreach efforts in information system security, and its collaborative activities with industry, government, and academic organizations.

National Institute of Standards and Technology Special Publication 800-73-2, Part 2,
29 pages (September 2008)

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

NIST makes no representation as to whether or not one or more implementations of SP 800-73-2 is/are covered by existing patents.

Acknowledgements

The authors (Ramaswamy Chandramouli, James Dray, Hildegard Ferraiolo, William MacGregor of NIST, Ketan Mehta of Mehta Inc. and Scott Guthery of Mobile Mind Inc.) wish to thank their colleagues who reviewed drafts of this document and contributed to its development. Special thanks to the Government Smart Card Interagency Advisory Board (GSC-IAB) and InterNational Committee for Information Technology Standards (INCITS) for providing detailed technical inputs to the SP 800-73-2 development process. The authors also gratefully acknowledge and appreciate the many contributions from the public and private sectors whose thoughtful and constructive comments improved the quality and usefulness of this publication.

Table of Contents

List of Appendices

List of Tables

1. Introduction

The Homeland Security Presidential Directive 12 (HSPD-12) called for a common identification standard to be adopted governing the interoperable use of identity credentials to allow physical and logical access to Federal government locations and systems. The Personal Identity Verification (PIV) of Federal Employees and Contractors, Federal Information Processing Standard 201 (FIPS 201) [1] was developed to establish standards for identity credentials. Special Publication 800-73-2 (SP 800-73-2) specifies interface requirements for retrieving and using the identity credentials from the PIV Card[1] and is a companion document to FIPS 201.

1.1 Authority

This document has been developed by the National Institute of Standards and Technology (NIST) in furtherance of its statutory responsibilities under the Federal Information Security Management Act (FISMA) of 2002, Public Law 107-347.

NIST is responsible for developing standards and guidelines, including minimum requirements, for providing adequate information security for all agency operations and assets, but such standards and guidelines shall not apply to national security systems. This recommendation is consistent with the requirements of the Office of Management and Budget (OMB) Circular A-130, Section 8b(3), Securing Agency Information Systems, as analyzed in A-130, Appendix IV: Analysis of Key Sections. Supplemental information is provided A-130, Appendix III.

This recommendation has been prepared for use by federal agencies. It may be used by non-governmental organizations on a voluntary basis and is not subject to copyright though attribution is desirable. Nothing in this document should be taken to contradict standards and guidelines made mandatory and binding on Federal agencies by the Secretary of Commerce under statutory authority. Nor should this recommendation be interpreted as altering or superseding the existing authorities of the Secretary of Commerce, Director of the Office of Management and Budget (OMB), or any other Federal official.

1.2 Purpose

FIPS 201 defines procedures for the PIV lifecycle activities including identity proofing, registration, PIV Card issuance, and PIV Card usage. FIPS 201 also specifies that the identity credentials must be stored on a smart card. SP 800-73-2 contains technical specifications to interface with the smart card to retrieve and use the identity credentials. The specifications reflect the design goals of interoperability and PIV Card functions. The goals are addressed by specifying a PIV data model, card edge interface, and application programming interface. Moreover, the specifications enumerate requirements where the standards include options and branches. SP 800-73-2 goes further by constraining implementers' interpretations of the normative standards. Such restrictions are designed to ease implementation, facilitate interoperability, and ensure performance, in a manner tailored for PIV applications.

[1] A physical artifact (e.g., identity card, “smart” card) issued to an individual that contains stored identity credentials (e.g., photograph, cryptographic keys, biometric data) so that the claimed identity of the cardholder can be verified against the stored credentials by another person (human readable and verifiable) or an automated process (computer readable and verifiable).

1.3 Scope

SP 800-73-2 specifies the PIV data model, Application Programming Interface (API) and card interface requirements necessary to comply with the use cases, as defined in Section 6 of FIPS 201 and further elaborated in Appendix B of SP 800-73-2 Part 1. Interoperability is defined as the use of PIV identity credentials such that client-application programs, compliant card applications, and compliant integrated circuits cards (ICC) can be used interchangeably by all information processing systems across Federal agencies.

This Part, SP 800-73-2 Part 2 – *End-Point PIV Card Application Interface* contains the technical specifications of the PIV Card command interface to the PIV Card. The specification defines the set of commands surfaced by the PIV Card Application at the card edge of the ICC.

1.4 Audience and Assumptions

This document is targeted at Federal agencies and implementers of PIV systems. Readers are assumed to have a working knowledge of smart card standards and applications.

1.5 Content and Organization

All sections in this document are *normative* (i.e., mandatory for compliance) unless specified as *informative* (i.e., non-mandatory). Following is the structure of Part 2:

- Section 1, *Introduction*, provides the purpose, scope, audience, and assumptions of the document and outlines its structure.
- Section 2*, Overview: End-Point Concept and Construct,* describes the model of computation of the PIV Card Application and the PIV client-application programming interface including information processing concepts and data representation constructs.
- Section 3, *End-Point PIV Card Application Card Command Interface,* describes the set of commands accessible by the PIV middleware to communicate with the PIV Card Application.
- Appendix A, *Examples of the Use of GENERAL AUTHENTICATE*, demonstrates the GENERAL AUHTENTICATE command. This section is *informative.*
- Appendix B, *Terms, Acronyms, and Notation*, contains the list of Terms and Acronyms used in this document and explains the notation in use. This section is *informative*.
- Appendix C, *References*, contains the lists of documents used as references by this document. This section is *informative.*

2. Overview: End-Point Concepts and Constructs

SP 800-73-2 Part 2 and Part 3 define two interfaces to an ICC that contains the Personal Identity Verification card application: a low-level PIV Card Application card command interface (Part 2, card edge) and a high-level PIV client-API (Part 3).

The information processing concepts and data constructs on both interfaces are identical and may be referred to generically as the information processing concepts and data constructs on the *PIV interfaces* without specific reference to the client-application programming interface or the card command interface.

The client-application programming interface provides task-specific programmatic access to these concepts and constructs and the card command interface provides communication access to concepts and constructs. The client-application programming interface is used by client applications using the PIV Card Application. The card command interface is used by software implementing the client-application programming interface (middleware).

The client-application programming interface is thought of as being at a higher level than the card command interface because access to a single entry point on the client-application programming interface may cause multiple card commands to traverse the card command interface. In other words, it may require more than one card command on the card command interface to accomplish the task represented by a single call on an entry point of the client-application programming interface.

The client-application programming interface is a program execution, call/return style interface whereas the card command interface is a communication protocol, command/response style interface. Because of this difference, the representation of the PIV concepts and constructs as bits and bytes on the client-application program interface may be different from the representation of these same concepts and constructs on the card command interface.

2.1 Unified Card Command Interface

The card command interface of the PIV Card Application is a unification of the two card command interfaces found in Government Smart Card Interoperability Specification (GSC-IS) [2].

This unification is accomplished by adopting the object-oriented model of computation of the GSC-IS virtual machine card edge and realizing its technical details using the data structures and operations found in the international ICC standards [3] underpinning the GSC-IS file system card edge. This brings the PIV Card Application into conformance with those standards with minimal impact on existing GSC-IS deployments.

As a result of this unification, the behavior of the PIV Card Application and the client-applications accessing it is independent of the ICC platform on which the PIV Card Application is installed.

2.1.1 Platform Requirements

The following are the requirements that the PIV Card Application places on the ICC platform on which it is implemented or installed:

+ global security status that includes the security status of a global cardholder PIN

+ application selection using a truncated Application Identifier (AID)

+ ability to reset the security status of an individual application
+ indication to applications as to which physical communication interface – contact versus contactless – is in use
+ support for the default selection of an application upon warm or cold reset

2.2 Namespaces of the PIV Card Application

AID, Names, Tag Length Value (BER-TLV) [4] tags, ASN.1 [5] Object Identifiers (OIDs) and Proprietary Identifier Extensions (PIXes) of the NIST Registered Application Provider Identifier (RID) used on the PIV interfaces are specified in Part 1. Part 1 also specifies the use of all unspecified names, BER-TLV tags, OID, and values of algorithm identifiers, key reference, and cryptographic mechanism identifiers.

2.3 Card Applications

Each command that appears on the card command interface shall be implemented by a *card application* that is resident on the ICC. The card command enables operations on and with the data objects to which the card application has access.

Each card application shall have a globally unique name called its Application Identifier (AID) [3, Part 4]. Except for the default applications, access to the card commands and data objects of a card application shall be gained by selecting the card application using its application identifier[2]. The PIX of the AID shall contain an encoding of the version of the card application. The AID of the Personal Identity Verification card application (PIV Card Application) is defined in Part 1.

The card application whose commands are currently being used is called the *currently selected application*.

2.3.1 Default Selected Card Application

The card platform shall support a default selected card application. In other words, there shall be a currently selected application immediately after a cold or warm reset. This card application is the default selected card application. The default card application may be the PIV Card Application, or it may be another card application.

2.4 Security Architecture

The security architecture of an ICC is the means by which the security policies governing access to each data object stored on the card are represented within the card.

The software in the ICC applies these security policy representations to all card commands thereby ensuring that the prescribed data policies for the card applications are enforced.

The following subsections describe the security architecture of the PIV Card Application.

[2] Access to the default application (and its commands and objects), occurs immediately after a warm or cold card reset without an explicit SELECT command.

2.4.1 Access Control Rule

An *access control rule* shall consist of an *access mode* and a *security condition*. The access mode is an operation that can be performed on a data object. A security condition is a Boolean expression using variables called security statuses that are defined below.

According to an access control rule, the action described by the access mode can be performed on the data object if and only if the security condition evaluates to TRUE for the current values of the security statuses. If there is no access control rule with an access mode describing a particular action, then that action shall never be performed on the data object.

2.4.2 Security Status

Associated with each authenticatable entity shall be a set of one or more Boolean variables each called a *security status indicator* of the authenticatable entity. Each security status indicator, in turn, is associated with a credential that can be used to authenticate the entity. The security status indicator of an authenticatable entity shall be TRUE if the credentials associated with the security status indicator of the authenticatable entity have been authenticated and FALSE otherwise.

A successful execution of an authentication protocol shall set the security status indicator associated with the credential used in the protocol to TRUE. An aborted or failed execution of an authentication protocol shall set the security status indicator associated with the credential used in the protocol to FALSE.

As an example, the credentials associated with two security status indicators of the cardholder might be: PIN and fingerprint. Demonstration of knowledge of the PIN is the authentication protocol for the first security status indicator wherein the PIN is the credential. Comparison of the fingerprint template on the card with a fingerprint acquired from the cardholder is the authentication protocol for the second security status indicator wherein the fingerprint is the credential. A security condition using these two security status indicators might be (PIN AND fingerprint).

A security status indicator shall be said to be a *global* security status indicator if it is not changed when the currently selected application changes from one application to another.

A security status indicator is said to be an *application* security status indicator if it is set to FALSE when the currently selected application changes from one application to another. Every security status indicator is either a global security status indicator or an application security status indicator.

The term *global security status* refers to the set of all global security status indicators. The term *application security status* refers to the set of all application security status indicators for a specific application.

2.4.3 Authentication of an Individual

Knowledge of a PIN is the means by which an individual can be authenticated to the PIV Card Application.

Personal identification numbers presented to the card command interface shall be 8 bytes long. If the actual PIN length is less than 8 bytes it shall be padded to 8 bytes with 'FF'. The 'FF' padding bytes shall be appended to the actual PIN. The bytes comprising the PIN shall not include 'FF'. For example,

+ Actual PIN: “123456” or '31 32 33 34 35 36'

+ Padded PIN presented to the card command interface: '31 32 33 34 35 36 FF FF'

Note that the FIPS 201 PIN requirements only apply to the PIV Application PIN. However, the above length and padding requirement for the PIV card edge interface applies to both the PIV application PIN and Global PIN (if implemented).

2.5 Current State of the PIV Card Application

The elements of the *current state* of the PIV Card Application when the PIV Card Application is the currently selected application are described in Table 1.

Table 1. State of the PIV Card Application

State Name	Always Defined	Comment	Location of State
Global security status	Yes	Contains security status indicators that span all card applications on the platform.	PIV Platform
Currently selected application	Yes	The platform shall support the selection of a card application using the full application identifier or by providing the right-truncated version and there shall always be a currently selected application.	PIV Platform
Application security status	Yes	Contains security status indicators local to the PIV Card Application.	PIV Card Application

3. End-Point PIV Card Application Card Command Interface

Table 2 lists the card commands surfaced by the PIV Card Application at the card edge of the ICC when it is the currently selected card application. All PIV Card Application card commands shall be supported by a PIV Card Application. Card commands indicated with a 'Yes' in the Command Chaining column shall support command chaining for transmitting a data string too long for a single command as defined in ISO/IEC 7816-4 [3].

Table 2. PIV Card Application Card Commands

Type	Name	Contact Interface	Contactless Interface	Security Condition for Use	Command Chaining
PIV Card Application Card Commands for Data Access	**SELECT**	Yes	Yes	Always	No
	GET DATA	Yes	Yes	Data Dependent. See Table 1, Part 1.	No
PIV Card Application Card Commands for Authentication	**VERIFY**	Yes	No	Always	No
	CHANGE REFERENCE DATA	Yes	No	PIN	No
	RESET RETRY COUNTER	Yes	No	PIN Unblocking Key	No
	GENERAL AUTHENTICATE	Yes	Yes (See Note)	Key Dependent. See Table 3, Part 1.	Yes
PIV Card Application Card Commands for Credential Initialization and Administration	**PUT DATA**	Yes	No	PIV Card Application Administrator	Yes
	GENERATE ASYMMETRIC KEY PAIR	Yes	No	PIV Card Application Administrator	Yes

The PIV Card Application shall return the status word of '6A81' (Function not supported) when it receives a card command on the contactless interface marked “No” in the Contactless Interface column in Table 2.

Note: Cryptographic protocols using private/secret keys requiring ”PIN" security condition shall not be used on the contactless interface.

3.1 PIV Card Application Card Commands for Data Access

3.1.1 SELECT Card Command

The SELECT card command sets the currently selected application. The PIV Card Application shall be selected by providing its application identifier (see Part 1, Section 2.2), in the data field of the SELECT command.

There shall be at most one PIV Card Application on any ICC. The PIV Card Application can also be made the currently selected application by providing the right-truncated version (see Part 1, Section 2.2); that is, without the two-byte version number in the data field of the SELECT command.

The complete AID, including the two-byte version, of the PIV Card Application that became the currently selected card application upon successful execution of the SELECT command (using the full or right-truncated PIV AID) shall be returned in the application property template.

If the currently selected application is the PIV Card Application when the SELECT command is given and the AID in the data field of the SELECT command is either the AID of the PIV Card Application or the right-truncated version thereof, then the PIV Card Application shall continue to be the currently selected card application and the setting of all security status indicators in the PIV Card Application shall be unchanged.

If the currently selected application is the PIV Card Application when the SELECT command is given and the AID in the data field of the SELECT command is not the PIV Card Application (nor the right-truncated version thereof), but a valid AID supported by the ICC, then the PIV Card Application shall be deselected and all the PIV Card Application security status indicators in the PIV Card Application shall be set to FALSE.

If the currently selected application is the PIV Card Application when the SELECT command is given and the AID in the data field of the SELECT command is an invalid AID not supported by the ICC, then the PIV Card Application shall remain the current selected card application and all PIV Card Application security status indicators shall remain unchanged.

Command Syntax

CLA	'00'
INS	'A4'
P1	'04'
P2	'00'
L_c	Length of application identifier
Data Field	AID of the PIV Card Application, using the full AID or by providing the right-truncated AID (See Section 2.2, Part 1)
L_e	Length of application property template

Response Syntax

Data Field	Application property template (APT). See Table 3 below
SW1-SW2	Status word

Upon selection, the PIV Card Application shall return the application property template described in Table 3.

Table 3. Data Objects in the PIV Card Application Property Template (Tag '61')

Description	Tag	M/O	Comment
Application identifier of application	'4F'	M	The PIX of the AID includes the encoding of the version of the PIV Card Application. See Section 2.2, Part 1.
Coexistent tag allocation authority	'79'	M	Coexistent tag allocation authority template. See Table 4.
Application label	' '50'	O	Text describing the application; e.g. for use on a man-machine interface.
Uniform resource locator	'5F50'	O	Reference to the specification describing the application.

Table 4. Data Objects in a Coexistent Tag Allocation Authority Template (Tag '79')

Description	Tag	M/O	Comment
Application identifier	'4F'	M	See Section 2.2, Part 1

SW1	SW2	Meaning
'6A'	'82'	Application not found
'90'	'00'	Successful execution

3.1.2 GET DATA Card Command

The GET DATA card command retrieves the data content of the single data object whose tag is given in the data field.

Command Syntax

CLA	'00'
INS	'CB'
P1	'3F'
P2	'FF'
L_c	Length of data field*
Data Field	See Table 5
L_e	Number of data content bytes to be retrieved.

* The L_c value is '05' for all PIV data objects except for the 0x7E interindustry tag (Discovery Object), and the Application Property Template (APT), which have a L_c value of '03'.

Table 5. Data Objects in the Data Field of the GET DATA Card Command

Name	Tag	M/O	Comment
Tag list	'5C'	M	BER-TLV tag of the data object to be retrieved. See Table 2, Part 1.

Response Syntax

For the (optional) 0x7E Discovery Object (if present):

Data Field	BER-TLV of the 0x7E Discovery data object (see Section 3.2.6, Part 1 for an example of the Discovery Object's structure returned in the data field).
SW1-SW2	Status word

For all other PIV data objects:

Data Field	BER-TLV with the tag '53' containing in the value field of the requested data object.
SW1-SW2	Status word

SW1	SW2	Meaning
'61'	'xx'	Successful execution where SW2 encodes the number of response data bytes still available
'69'	'82'	Security status not satisfied
'6A'	'82'	Data object not found
'90'	'00'	Successful execution

3.2 PIV Card Application Card Commands for Authentication

3.2.1 VERIFY Card Command

The VERIFY card command initiates the comparison in the card of the reference data indicated by the key reference with authentication data in the data field of the command.

Key reference '80' specific to the PIV Card Application (i.e., local key references) and, optionally, the Global PIN with key reference '00' are the only key references that may be verified by the PIV Card Application's VERIFY command.

Key reference '80' shall be verified by the PIV Card Application VERIFY command.

If the PIV Card Application contains the Discovery Object as described in Part 1, and the first byte of the PIN Usage Policy value is 0x60, then key reference '00' shall be verified by the PIV Card Application VERIFY command.

If the current value of the retry counter associated with the key reference is zero, then the comparison shall not be made, and the PIV Card Application shall return the status word '69 83'.

If the authentication data in the command data field does not satisfy the criteria in Section 2.4.3, then the card command shall fail, and the PIV Card Application shall return the status word '6A 80'.

If the authentication data in the command data field does not match reference data associated with the key reference then the card command shall fail.

If the card command fails, then the security status of the key reference shall be set to FALSE and the retry counter associated with the key reference shall be decremented by one.

If the card command succeeds, then the security status of the key reference shall be set to TRUE and the retry counter associated with the key reference shall be set to the reset retry value associated with the key reference.

The initial value of the retry counter and the reset retry value associated with the key reference; i.e. the number of successive failures (retries) before the retry counter associated with the key reference reaches zero, are issuer dependent.

Command Syntax

CLA	'00'
INS	'20'
P1	'00'
P2	Key reference. See Part 1, Table 3.
L_c	'00'[3] or '08'
Data Field	Absent[4] or PIN reference data as described in Section 2.4.3
L_e	Empty

Response Syntax

SW1	SW2	Meaning
'63'	'CX'	Verification failed, X indicates the number of further allowed retries
'69'	'83'	Authentication method blocked
'6A'	'80'	Incorrect parameter in command data field
'6A'	'88'	Key reference not found
'90'	'00'	Successful execution

3.2.2 CHANGE REFERENCE DATA Card Command

The CHANGE REFERENCE DATA card command initiates the comparison of the verification data with the current value of the reference data and if this comparison is successful, replaces the reference data with new reference data.

[3] If L_c=0x00 and the command data field is empty, the command can be used to retrieve the number of further retries allowed ('63 CX'), or to check whether verification is not needed ('90 00').

Only reference data associated with key reference '80' and '81' specific to the PIV Card Application (i.e. local key reference) and optionally the Global PIN with key reference '00' may be changed by the PIV Card Application CHANGE REFERENCE DATA command.

Key reference '80' reference data shall be changed by the PIV Card Application CHANGE REFERENCE DATA command.

If the current value of the retry counter associated with the key reference is zero, then the reference data associated with the key reference shall not be changed and the PIV Card Application shall return the status word '69 83'.

If the card command succeeds, then the security status of the key reference shall be set to TRUE and the retry counter associated with the key reference shall be set to the reset retry value associated with the key reference.

If the card command fails, then the security status of the key reference shall be set to FALSE and the retry counter associated with the key reference shall be decremented by one.

The initial value of the retry counter and the reset retry value associated with the key reference; i.e., the number of successive failures (retries) before the retry counter associated with the key reference reaches zero, is issuer dependent.

If either the current reference data or the new reference data in the command field of the command does not satisfy the criteria in Section 2.4.3, the PIV Card Application shall not change the reference data associated with the key reference and shall return the status word '6A 80'.

Command Syntax

CLA	'00'
INS	'24'
P1	'00'
P2	Key reference. See Part 1, Table 3
L_c	'10'
Data Field	Current PIN reference data concatenated without delimitation with the new PIN reference data, both PINs as described in Section 2.4.3
L_e	Empty

Response Syntax

SW1	SW2	Meaning
'63'	'CX'	Reference data change failed, X indicates the number of further allowed retries or resets
'69'	'83'	Reference data change operation blocked
'6A'	'80'	Incorrect parameter in command data field
'6A'	'88'	Key reference not found
'90'	'00'	Successful execution

3.2.3 RESET RETRY COUNTER Card Command

The RESET RETRY COUNTER card command resets the retry counter of the key reference to its initial value and changes the reference data associated with the key reference. The command enables recovery of the PIV card application PIN in the case that the cardholder has forgotten a PIV Card Application PIN.

Only retry counters associated with key references specific to the PIV Card Application; i.e. local key references may be reset by the PIV Card Application RESET RETRY COUNTER command.

If the current value of the reset counter associated with the key reference is zero, then retry counter associated with the key reference shall not be reset and the PIV Card Application shall the status word '69 83'.

If the card command succeeds, then the retry counter associated with the key reference shall be set to the reset retry value associated with the key reference. The security status of the key reference shall not be changed.

If the card command fails, then the security status of the key reference shall be set to FALSE and the reset counter associated with the key reference shall be decremented by one.

The initial reset counter associated with the key reference; i.e. the number of failures of the RESET RETRY COUNTER command before the reset counter associated with the key reference reaches zero, is issuer dependent.

If either the reset retry counter reference data (PUK) or the new reference data (PIN) in the command field of the command does not satisfy the criteria in Section 2.4.3, the PIV Card Application shall not reset the retry counter associated with the key reference and shall return the status word '6A 80'.

Command Syntax

CLA	'00'
INS	'2C'
P1	'00'
P2	Key reference. See Part 1, Table 3
L_c	'10'
Data Field	Reset retry counter reference data (PUK) concatenated without delimitation with the new reference data (PIN), both PUK and PIN as described in 2.4.3
L_e	Empty

Response Syntax

SW1	SW2	Meaning
'63'	'CX'	Reset failed, X indicates the number of further allowed resets
'69'	'83'	Reset operation blocked
'6A'	'80'	Incorrect parameter in command data field
'6A'	'88'	Key reference not found
'90'	'00'	Successful execution

3.2.4 GENERAL AUTHENTICATE Card Command

The GENERAL AUTHENTICATE card command performs a cryptographic operation such as an authentication protocol using the data provided in the data field of the command and returns the result of the cryptographic operation in the response data field.

The GENERAL AUTHENTICATE command shall be used to authenticate the card or a card application to the client-application (INTERNAL AUTHENTICATE), to authenticate an entity to the card (EXTERNAL AUTHENTICATE), and to perform a mutual authentication between the card and an entity external to the card (MUTUAL AUTHENTICATE).

The GENERAL AUTHENTICATE command shall be used to realize the signing functionality on the PIV client-application programming interface. Data sent to the card is expected to be hashed off-card.

The GENERAL AUTHENTICATE command supports command chaining to permit the uninterrupted transmission of long command data fields to the PIV Card Application. If a card command other than the GENERAL AUTHENTICATICATE command is received by the PIV Card Application before the termination of a GENERAL AUTHENTICATE chain, the PIV Card Application shall rollback to the state it was in immediately prior to the reception of the first command in the interrupted chain. In other words, an interrupted GENERAL AUTHENTICATE chain has no effect on the PIV Card Application.

Command Syntax

CLA	'00' or '10' indicating command chaining.
INS	'87'
P1	Algorithm reference. See Table 6-2, SP 800-78 [7].
P2	Key reference. See Table 6-1, SP 800-78.
L_c	Length of data field
Data Field	See Table 6.
L_e	Absent or length of expected response

Table 6. Data Objects in the Dynamic Authentication Template (Tag '7C')

Name	Tag	M/O	Description
Witness	'80'	C	Demonstration of knowledge of a fact without revealing the fact. An empty witness is a request for a witness.
Challenge	'81'	C	One or more random numbers or byte sequences to be used in the authentication protocol.
Response	'82'	C	A sequence of bytes encoding a response step in an authentication protocol.

The data objects that appear in the dynamic authentication template (tag '7C') in the data field of the GENERAL AUTHENTICATE card command depend on the authentication protocol being executed. The Witness (80) contains encrypted data (unrevealed fact). This data is decrypted by the card. The Challenge (81) contains clear data (byte sequence) which is encrypted by the card. The Response (tag '82') contains either the decrypted data from tag '80' or the encrypted data from tag '81'. Note that the empty tags (i.e., tags with no data) return the same tag with content (they can be seen as "requests for requests"):

- '80 00' Returns '80 TL <encrypted random>' (as per definition)
- '81 00' Returns '81 TL <random>' (as per external auth example)

Response Syntax

Data Field	Absent or authentication-related data
SW1-SW2	Status word

SW1	SW2	Meaning
'61'	'xx'	Successful execution where SW2 encodes the number of response data bytes still available
'69'	'82'	Security status not satisfied
'6A'	'80'	Incorrect parameter in command data field
'6A'	'86'	Incorrect parameter in P1 or P2
'90'	'00'	Successful execution

3.3 PIV Card Application Card Commands for Credential Initialization and Administration

3.3.1 PUT DATA Card Command

The PUT DATA card command completely replaces the data content of a single data object in the PIV Card Application with new content.

Command Syntax

CLA	'00' or '10' indicating command chaining
INS	'DB'
P1	'3F'
P2	'FF'
L_c	Length of data field
Data Field	See Table 7
L_e	Empty

Table 7. Data Objects in the Data Field of the PUT DATA Card Command for the Discovery Object

For the 0x7E Discovery Object (if present):

Tag	M/O	Description
'7E'	M	BER-TLV of tag '7E' as illustrated in Section 3.2.6, Part 1

Table 8. Data Objects in the Data Field of the PUT DATA Card Command for all other PIV Data Objects

For all other PIV Data objects:

Name	Tag	M/O	Description
Tag list	'5C'	M	Tag of the data object whose data content is to be replaced. See Table 2, Part 1.
Data	'53'	M	Data with tag '53' as an unstructured byte sequence.

Response Syntax

Data Field	Absent
SW1-SW2	Status word

SW1	SW2	Meaning
'69'	'82'	Security status not satisfied
'6A'	'84'	Not enough memory
'90'	'00'	Successful execution

3.3.2 GENERATE ASYMMETRIC KEY PAIR Card Command

The GENERATE ASYMMETRIC KEY PAIR card command initiates the generation and storing in the card of the reference data of an asymmetric key pair, i.e., a public key and a private key. The public key of the generated key pair is returned as the response to the command. If there is reference data currently associated with the key reference, it is replaced in full by the generated data.

Command Syntax

CLA	'00' or '10' indicating command chaining.
INS	'47'
P1	'00'
P2	See SP 800-78 Table 6-1 for a list of the PIV Key References

L_c	Length of data field
Data Field	Control reference template. See Table 9
L_e	Length of public key of data object template

Table 9. Data Objects in the Template (Tag 'AC')

Name	Tag	M/O	Description
Cryptographic mechanism identifier	'80'	M	See Part 1, Table 4
Parameter	'81'	C	Specific to the cryptographic mechanism

Response Syntax

Data Field	Data objects of public key of generated key pair. See Table 10
SW1-SW2	Status word

Table 10. Data Objects in the Template (Tag '7F49')

Name	Tag
Public key data objects for RSA	
Modulus	'81'
Public exponent	'82'
Public key data objects for ECDSA	
Point	'86'

SW1	SW2	Meaning
'61'	'xx'	Successful execution where SW2 encodes the number of response data bytes still available
'69'	'82'	Security status not satisfied
'6A'	'80'	Incorrect parameter in command data field; e.g. unrecognized cryptographic mechanism
'6A'	'86'	Incorrect parameter P2; cryptographic mechanism of reference data to be generated different than cryptographic mechanism of reference data of given key reference
'90'	'00'	Successful execution

Appendix A—Examples of the Use of GENERAL AUTHENTICATE

A.1 Authentication of the PIV Card Application Administrator

The PIV Card Application Administrator is authenticated by the PIV Card Application using a challenge/response protocol. A challenge retrieved from the PIV Card Application is encrypted by the client-application and returned to the PIV Card Application associated with key reference '9B', the key reference to the Card Management Key[4]. The PIV Card Application decrypts the response using this reference data and the algorithm associated with the key reference; that is 3 Key Triple DES – ECB (algorithm identifier '00'). If this decrypted value matches the previously provided challenge, then the security status indicator of the PIV Card Application Administrator is set to TRUE within the PIV Card Application.

Table 11 shows the GENERAL AUTHENTICATE card commands sent to the PIV Card Application to realize this particular challenge/response protocol.

Table 11. Authentication of PIV Card Application Administrator

Command	Response	Comment
'00 87 00 00 04 7C 02 81 00'		Client-application requests a challenge from the PIV Card Application
	'7C 0A 81 08 01 02 03 04 05 06 07 08'	Challenge returned to client-application by the PIV Card Application
'00 87 00 9B 0C 7C 0A 82 08 88 77 66 55 44 33 22 11'		Client-application returns the encryption of the challenge ('88 77 66 55 44 33 22 11') referencing algorithm '00' and key reference '9B'. See Tables 6.1 and 6.2 of SP 800-78.
	'9000'	PIV Card Application indicates successful authentication of PIV Card Application Administrator after decrypting '88 77 66 55 44 33 22 11' using the referenced algorithm and key and getting '01 02 03 04 05 06 07 08'

A.2 Validation of the PIV Card Application

The PIV Card Application is validated by first retrieving the X.509 Certificate of the PIV Authentication Key (OID 2.16.840.1.101.3.7.2.1.1) and verifying the signature on this certificate. Assuming the certificate is valid and current, the client-application requests the PIV Card Application to encrypt a challenge using the private key associated with this certificate; i.e., key reference '9A', algorithm

[4] The Card Management key is the PIV Card Application Administration Key used for managing the PIV card application.

identifier '06'. The response is decrypted using the public key in the certificate. If the decrypted response matches the challenge, then the PIV Card Application is validated.

Table 12 shows the GENERAL AUTHENTICATE card commands sent to the PIV Card Application to realize the validation of the PIV Card Application.

Table 12. Validation of the PIV Card Application Using GENERAL AUTHENTICATE

Command	Response	Comment
'00 87 06 9A 0E 7C 0C 82 00 81 08 01 02 03 04 05 06 07 08'		Client-application sends a challenge to the PIV Card Application indicating the reference data associated with key reference '9A' is to be used with algorithm '06'. See Tables 6.1 and 6.2 in SP 800-78.
	'7C 0A 82 08 88 77 66 55 44 33 22 11'	PIV Card Application returns the encryption of the challenge ('88 77 66 55 44 33 22 11') using the indicated key reference data and algorithm.

The same use of GENERAL AUTHENTICATE can be used to achieve a signing of a byte sequence such as a hash by the PIV Card Application. One needs only to indicate which algorithm and which key are to be used by setting the values of the P1 and the P2 parameters respectively.

Note: For exposition purposes, this example uses only an 8-byte challenge and response with a 1024-bit RSA key. In actual usage, a challenge and response more appropriate for this cryptographic algorithm would be used.

Appendix B—Terms, Acronyms, and Notation

B.1 Terms

Application Identifier A globally unique identifier of a card application as defined in ISO/IEC 7816-4.

Algorithm Identifier A PIV algorithm identifier is a one-byte identifier that specifies a cryptographic algorithm and key size. For symmetric cryptographic operations, the algorithm identifier also specifies a mode of operation (i.e., CBC or ECB).

Authenticatable Entity An entity that can successfully participate in an authentication protocol with a card application.

BER-TLV Data Object A data object coded according to ISO/IEC 8825-2.

Card An integrated circuit card.

Card Application A set of data objects and card commands that can be selected using an application identifier.

Client Application A computer program running on a computer in communication with a card interface device.

Data Object An item of information seen at the card command interface for which are specified a name, a description of logical content, a format and a coding.

Key Reference A PIV key reference is a one-byte identifier that specifies a cryptographic key according to its PIV Key Type. The identifier is part of cryptographic material used in a cryptographic protocol such as an authentication or a signing protocol.

Object Identifier A globally unique identifier of a data object as defined in ISO/IEC 8824-2.

Reference Data Cryptographic material used in the performance a cryptographic protocol such as an authentication or a signing protocol. The reference data length is the maximum length of a password or PIN. For algorithms, the reference data length is the length of a key.

Status Word Two bytes returned by an integrated circuit card after processing any command that signify the success of or errors encountered during said processing.

Template A (constructed) BER-TLV data object whose value field contains specific BER-TLV data objects.

B.2 Acronyms

AID	Application Identifier
APDU	Application Protocol Data Unit
API	Application Programming Interface
ASN.1	Abstract Syntax Notation
BER	Basic Encoding Rules
CLA	Class (first) byte of a card command
DES	Data Encryption Standard
DNS	Domain Name Server
ECDSA	Elliptic Curve Digital Signature Algorithm
FIPS	Federal Information Processing Standards
FISMA	Federal Information Security Management Act
GSC-IAB	Government Smart Card Interagency Advisory Board
GSC-IS	Government Smart Card Interoperability Specification
HSPD	Homeland Security Presidential Directive
ICC	Integrated Circuit Card
IEC	International Electrotechnical Commission
IETF	Internet Engineering Task Force
INS	Instruction (second) byte of a card command
INCITS	InterNational Committee for Information Technology Standards
ISO	International Standards Organization
ITL	Information Technology Laboratory
LSB	Least Significant Bit
MSB	Most Significant Bit
NIST	National Institute of Standards and Technology
OID	Object Identifier

OMB	Office of Management and Budget
P1	First parameter of a card command
P2	Second parameter of a card command
PIN	Personal Identification Number
PIV	Personal Identity Verification
PIX	Proprietary Identifier extension
PUK	PIN Unblocking Key
RFU	Reserved for Future Use
RID	Registered application provider Identifier
RFU	Reserved for Future Use
RSA	Rivest, Shamir, Adleman
SP	Special Publication
SW1	First byte of a two-byte status word
SW2	Second byte of a two-byte status word
TLV	Tag-Length-Value

B.3 Notation

The sixteen hexadecimal digits shall be denoted using the alphanumeric characters 0, 1, 2…, A, B, C, D, E, and F. A byte consists of two hexadecimal digits, for example, '2D'. A sequence of bytes may be enclosed in single quotation marks, for example 'A0 00 00 01 16' rather than given as a sequence of individual bytes, 'A0' '00' '00' '01' '16'.

A byte can also be represented by bits b8 to b1, where b8 is the most significant bit (MSB) and b1 is the least significant bit (LSB) of the byte. In textual or graphic representations, the leftmost bit is the MSB. Thus, for example, the most significant bit, b8, of '80' is 1 and the least significant bit, b1, is 0.

All bytes specified as RFU shall be set to '00' and all bits specified as reserved for future use shall be set to 0.

All lengths shall be measured in number of bytes unless otherwise noted.

Data objects in templates are described as being mandatory (M), optional (O) or conditional (C). 'Mandatory' means the data object shall appear in the template. 'Optional' means the data object may appear in the template. In the case of 'conditional' data objects, the conditions under which they are required are provided in a footnote to the table.

In other tables the M/O column identifies properties of the PIV Card Application that shall be present (M) or may be present (O).

BER-TLV data object tags are represented as byte sequences as described above. Thus, for example, '4F' is the interindustry data object tag for an application identifier and '7F 60' is the interindustry data object tag for the biometric information template.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this standard are to be interpreted as described in IETF RFC 2119, Key Words for Use in RFCs to Indicate Requirement Levels [6].

Appendix C—References

[1] Federal Information Processing Standard 201-1, Change Notice 1, *Personal Identity Verification (PIV) of Federal Employees and Contractors*, March 2006. (See http://csrc.nist.gov)

[2] *Government Smart Card Interoperability Specification*, Version 2.1, NIST Interagency Report 6887 – 2003 Edition, July 16, 2003.

[3] ISO/IEC 7816 (Parts 4, 5, 6, 8, and 9), *Information technology — Identification cards — Integrated circuit(s) cards with contacts.*

[4] ISO/IEC 8825-1:2002, *Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).*

[5] ISO/IEC 8824-2:2002, *Information technology -- Abstract Syntax Notation One (ASN.1): Information object specification.*

[6] IETF RFC 2119, *Key Words for Use in RFCs to Indicate Requirement Levels*, March, 1997.

[7] NIST Special Publication 800-78-1, Cryptographic *Algorithms and Key Sizes for Personal Identity Verification*, August 2007. (See http://csrc.nist.gov)

NIST Special Publication 800-73-2

Interfaces for Personal Identity Verification – Part 3: End-Point PIV Client Application Programming Interface

National Institute of
Standards and Technology
U.S. Department of Commerce

Ramaswamy Chandramouli
James F. Dray
Hildegard Ferraiolo
Scott B. Guthery
William MacGregor
Ketan Mehta

INFORMATION SECURITY

Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD, 20899-8930

September 2008

U.S. Department of Commerce
Carlos M. Gutierrez, Secretary

National Institute of Standards and Technology
James M. Turner, Deputy Director

Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the Nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analyses to advance the development and productive use of information technology. ITL's responsibilities include the development of management, administrative, technical, and physical standards and guidelines for the cost-effective security and privacy of non-national security-related information in Federal information systems. This special publication 800-series reports on ITL's research, guidelines, and outreach efforts in information system security, and its collaborative activities with industry, government, and academic organizations.

**National Institute of Standards and Technology Special Publication 800-73-2, Part 3,
19 pages (September 2008)**

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

NIST makes no representation as to whether or not one or more implementations of SP 800-73-2 is/are covered by existing patents.

Acknowledgements

The authors (Ramaswamy Chandramouli, James Dray, Hildegard Ferraiolo, William MacGregor, of NIST Ketan Mehta of Mehta Inc. and Scott Guthery of Mobile Mind, Inc.) wish to thank their colleagues who reviewed drafts of this document and contributed to its development. Special thanks to the Government Smart Card Interagency Advisory Board (GSC-IAB) and InterNational Committee for Information Technology Standards (INCITS) for providing detailed technical inputs to the SP 800-73-2 development process. The authors also gratefully acknowledge and appreciate the many contributions from the public and private sectors whose thoughtful and constructive comments improved the quality and usefulness of this publication.

Table of Contents

List of Appendices

List of Tables

1. Introduction

The Homeland Security Presidential Directive 12 (HSPD-12) called for a common identification standard to be adopted governing the interoperable use of identity credentials to allow physical and logical access to Federal government locations and systems. The Personal Identity Verification (PIV) of Federal Employees and Contractors, Federal Information Processing Standard 201 (FIPS 201) [1] was developed to establish standards for identity credentials. Special Publication 800-73-2 (SP 800-73-2) specifies interface requirements for retrieving and using the identity credentials from the PIV Card and is a companion document to FIPS 201.

1.1 Authority

This document has been developed by the National Institute of Standards and Technology (NIST) in furtherance of its statutory responsibilities under the Federal Information Security Management Act (FISMA) of 2002, Public Law 107-347.

NIST is responsible for developing standards and guidelines, including minimum requirements, for providing adequate information security for all agency operations and assets, but such standards and guidelines shall not apply to national security systems. This recommendation is consistent with the requirements of the Office of Management and Budget (OMB) Circular A-130, Section 8b(3), Securing Agency Information Systems, as analyzed in A-130, Appendix IV: Analysis of Key Sections. Supplemental information is provided A-130, Appendix III.

This recommendation has been prepared for use by federal agencies. It may be used by non-governmental organizations on a voluntary basis and is not subject to copyright though attribution is desirable. Nothing in this document should be taken to contradict standards and guidelines made mandatory and binding on Federal agencies by the Secretary of Commerce under statutory authority. Nor should this recommendation be interpreted as altering or superseding the existing authorities of the Secretary of Commerce, Director of the Office of Management and Budget (OMB), or any other Federal official.

1.2 Purpose

FIPS 201 defines procedures for the PIV lifecycle activities including identity proofing, registration, PIV Card issuance, and PIV Card usage. FIPS 201 also specifies that the identity credentials must be stored on a smart card. SP 800-73-2 contains technical specifications to interface with the smart card to retrieve and use the identity credentials. The specifications reflect the design goals of interoperability and PIV Card functions. The goals are addressed by specifying a PIV data model, card edge interface, and Application Programming Interface (API). Moreover, SP 800-73-2 enumerates requirements where the standards include options and branches. The specifications go further by constraining implementers' interpretations of the normative standards. Such restrictions are designed to ease implementation, facilitate interoperability, and ensure performance, in a manner tailored for PIV applications.

1.3 Scope

SP 800-73-2 specifies the PIV data model, Application Programming Interface and card interface requirements necessary to comply with the use cases, as defined in Section 6 of FIPS 201 and further elaborated in Appendix B of SP 800-73-2, Part 1. Interoperability is defined as the use of PIV identity credentials such that client-application programs, compliant card applications, and compliant integrated circuits cards (ICC) can be used interchangeable by all information processing system across Federal agencies.

This Part, Special Publication 800-73-2 (SP 800-73-2) Part 3: *End-Point PIV Client Application Programming Interface* contains technical specifications of the PIV client application programming interface to the PIV Card.

1.4 Audience and Assumptions

This document is targeted at Federal agencies and implementers of PIV systems. Readers are assumed to have a working knowledge of smart card standards and applications.

1.5 Content and Organization

All sections in this document are *normative* (i.e., mandatory for compliance) unless specified as *informative* (i.e., non-mandatory). Following is the structure of Part 3:

- Section 1, *Introduction*, provides the purpose, scope, audience and assumptions of the document and outlines its structure.
- Section 2*, Overview: End-Point Concept and Construct,* describes both PIV Card Application and the PIV client-application programming interface. This section is informative.
- Section 3, *End-Point PIV Client Application Programming Interface,* describes the set of entry points accessible by client applications through the PIV middleware to interact with the PIV Card.
- Appendix A, *Terms, Acronyms, and Notation*, contains the list of Terms and Acronyms used in this document and explains the notation in use. This section is *informative*.
- Appendix B, *References*, contains the list of documents used as references by this document. This section is *informative*.

2. Overview: End-Point Concepts and Constructs

SP 800-73-2 Part 2 and Part 3 define two interfaces to an ICC that contains the Personal Identity Verification card application: a low-level PIV Card Application card command interface (Part 2) and a high-level PIV client-API (Part 3).

The information processing concepts and data constructs on both interfaces are identical and may be referred to generically as the information processing concepts and data constructs on the *PIV interfaces* without specific reference to the client-application programming interface or the card command interface.

The client-application programming interface provides task-specific programmatic access to these concepts and constructs and the card command interface provides communication access to concepts and constructs. The client-application programming interface is used by client applications using the PIV Card Application. The card command interface is used by software implementing the client-application programming interface (middleware).

The client-application programming interface is thought of as being at a higher level than the card command interface because access to a single entry point on the client-application programming interface may cause multiple card commands to traverse the card command interface. In other words, it may require more than one card command on the card command interface to accomplish the task represented by a single call on an entry point of the client-application programming interface.

The client-application programming interface is a program execution, call/return style interface, whereas the card command interface is a communication protocol, command/response style interface. Because of this difference the representation of the PIV concepts and constructs as bits and bytes on the client-application program interface may be different from the representation of these same concepts and constructs on the card command interface.

3. End-Point Client-Application Programming Interface

Table 1 lists the entry points on the PIV client-application programming interface. This section references Object Identifiers (OIDs), which are defined and can be found in Part 1 (Table 2).

Table 1. Entry Points on PIV End-Point Client-Application Programming Interface

Type	Name
Entry Points for Communication	**pivMiddlewareVersion**
	pivConnect
	pivDisconnect
Entry Points for Data Access	**pivSelectCardApplication**
	pivLogIntoCardApplication
	pivGetData
	pivLogoutOfCardApplication
Entry Points for Cryptographic Operations	**pivCrypt**
Entry Points for Credential Initialization and Administration	**pivPutData**
	pivGenerateKeyPair

3.1 Entry Points for Communication

3.1.1 pivMiddlewareVersion

Purpose: Returns the PIV Middeware version string

Prototype:

```
status_word pivMiddlewareVersion(
   OUT   version           versionString
);
```

Parameter: `versionString` For SP 800-73-2 Part 3 conformant PIV middleware, the parameter returns “80073-2 Client API”.

Return Codes: `PIV_OK`

Note: SP 800-73-1 conformant PIV Middleware does not implement the pivMiddlewareVersion Client API function. Therefore, a client application invoking the pivMiddlewareVersion function should expect a "function-not-supported" error from SP 800-73-1 conformant PIV Middleware. For purposes of version determination, failure to obtain a specific version from pivMiddlewareVersion shall be considered equivalent to obtaining a response of "800-73-1 Client API". SP 800-73-2 Part 3 conformant PIV middleware shall implement the new pivMiddlewareVersion function as well as the pivPutData and pivGetData functions for the 0x7E interindustry BER-TLV Discovery object.

3.1.2 pivConnect

Purpose: Connects the client-application programming interface to the PIV Card Application on a specific ICC.

Prototype:

```
status_word pivConnect(
   IN    Boolean           sharedConnection,
   INOUT sequence of bytes connectionDescription,
   IN    LONG              CDLength,
   OUT   handle            cardHandle
);
```

Parameters:

sharedConnection If TRUE other client-applications can establish concurrent connections to the ICC. If FALSE and the connection is established then the calling client-application has exclusive access to the ICC.

connectionDescription A connection description data object (tag '7F 21 '). See Table 2.

If the length of the value field of the '8x' data object in the connection description data object is zero then a list of the card readers of the type indicated by the tag of the '8x' series data object and available at the '9x' location is returned in the connectionDescription.

The connection description BER-TLV [2] used on the PIV client-application programming interface shall have the structure described in Table 2.

Table 2. Data Objects in a Connection Description Template (Tag '7F21')

Description	Tag	M/O/C[1]	Comment
Interface device – PC/SC	'81'	C	Card reader name
Interface device – SCP	'82'	C	Card reader identifier on terminal equipment
Interface device – EMR	'83'	C	Contactless connection using radio transmission
Interface device – IR	'84'	C	Contactless connection using infrared

[1] M = Mandatory, O = Optional, C = Conditional. For the definition of M/O/C see, Appendix A.3

			transmission
Interface device – PKCS#11	'85'	C	PKCS#11 interface
Interface device – CryptoAPI	'86'	C	CryptoAPI interface
Network node – Local	'90'	C	No network between client-application host and card reader host
Network node – IP	'91'	C	IP address of card reader host
Network node – DNS	'92'	C	Internet domain name of card reader host
Network node – ISDN	'93'	C	ISDN dialing number string of terminal equipment containing the card reader

At most one selection from the '8x' series and one selection from the '9x' series shall appear in the connection description template.

For example, '7F 21 0C 82 04 41 63 6D 65 91 04 81 06 0D 17' describes a connection to a generic card reader at Internet address 129.6.13.23. As another example, '7F 21 0B 82 01 00 93 06 16 17 12 34 56 7F' describes a connection to the subscriber identity module in the mobile phone at +1 617 123 4567.

When used as an argument to the pivConnect entry point on the PIV client-application programming interface described in his section, an '8x' series data object with zero length together with a '9x' series data object requests the return of all available card readers of the described type on the described node. Thus, '7F 21 04 81 00 90 00' would request a list of all available PC/SC card readers on the host on which the client-application was running.

CDLength Length of the card description parameter.

cardHandle The returned opaque identifier of a communication channel to a particular ICC and hence of the card itself. cardHandle is used in all other entry points on the PIV client-application programming interface to identify which card the functionality of the entry point is to be applied.

Return Codes:

```
PIV_OK
PIV_CONNECTION_DESCRIPTION_MALFORMED
PIV_CONNECTION_FAILURE
PIV_CONNECTION_LOCKED
```

3.1.3 pivDisconnect

Purpose: Disconnect the PIV application programming interface from the PIV Card Application and the ICC containing the PIV Card Application.

Prototype:

```
status_word pivDisconnect(
   IN handle            cardHandle
);
```

Parameters: **cardHandle** Opaque identifier of the card to be acted upon as returned by pivConnect. The value of cardHandle is undefined upon return from pivDisconnect.

Return Codes:

```
PIV_OK
PIV_INVALID_CARD_HANDLE
PIV_CARD_READER_ERROR
```

3.2 Entry Points for Data Access

3.2.1 pivSelectCardApplication

Purpose: Set the PIV Card Application as the currently selected card application and establish the PIV Card Application's security state.

Prototype:

```
status_word pivSelectCardApplication(
   IN handle            cardHandle,
   IN sequence of byte  applicationAID,
   IN LONG              aidLength,
   OUT sequence of byte applicationProperties,
   OUT LONG             APLength
);
```

Parameters:

cardHandle	Opaque identifier of the card to be acted upon as returned by pivConnect.
aidLength	Length of the Application AID.
applicationAID	The AID of the PIV Card Application that is to become the currently selected card application.
applicationProperties	The application properties of the selected PIV Card Application. See Part 2, Table 3.
APLength	Length of the application properties.

Return Codes:

```
PIV_OK
PIV_INVALID_CARD_HANDLE
PIV_CARD_APPLICATION_NOT_FOUND
PIV_CARD_READER_ERROR
```

3.2.2 pivLogIntoCardApplication

Purpose: Set security state within the PIV Card Application.

Prototype:

```
status_word pivLogIntoCardApplication(
   IN handle            cardHandle,
   IN sequence of byte  authenticators,
   OUT LONG             AuthLength
);
```

Parameters:

cardHandle	Opaque identifier of the card to be acted upon as returned by pivConnect.

authenticators A sequence of zero or more BER-TLV encoded authenticators to be used to authenticate and set security state/status in the PIV Card Application context.

The authenticator BER-TLV used on the PIV client-application programming interface shall have the structure described in Table 3.

AuthLength Length of the authenticator template.

Table 3. Data Objects in an Authenticator Template (Tag '67')

Description	Tag	M/O	Comment
Reference data	'81'	M	E.g. the PIN value or challenge response
Key reference	'83'	M	See table 3, Part 1 for PIN reference values

Return Codes:

```
PIV_OK
PIV_INVALID_CARD_HANDLE
PIV_AUTHENTICATOR_MALFORMED
PIV_AUTHENTICATION_FAILURE
PIV_CARD_READER_ERROR
```

3.2.3 pivGetData

Purpose: Return the entire data content of the named data object.

Prototype:

```
status_word pivGetData(
   IN handle            cardHandle,
   IN string            OID,
   IN LONG              oidLength,
   OUT sequence of byte data,
   OUT LONG             DataLength
);
```

Parameters:

cardHandle Opaque identifier of the card to be acted upon as returned by pivConnect.

OID Object identifier of the object whose data content is to be retrieved coded as a string; for example, '2.16.840.1.101.3.7.1.1.2.2.1' See Part 1 Table 2.

oidLength Length of the object identifier.

data Retrieved data content.

DataLength Length of the data to be retrieved from the PIV Card.

Return Codes:

```
PIV_OK
PIV_INVALID_CARD_HANDLE
PIV_INVALID_OID
PIV_DATA_OBJECT_NOT_FOUND
PIV_SECURITY_CONDITIONS_NOT_SATISFIED
PIV_CARD_READER_ERROR
```

3.2.4 pivLogoutOfCardApplication

Purpose: Reset the application security state/status of the PIV Card Application.

Prototype:

```
status_word pivLogOutOfCardApplication(
   IN handle            cardHandle
);
```

Parameters:

cardHandle Opaque identifier of the card to be acted upon as returned by pivConnect. The cardHandle remains valid after execution of this function.

Return Codes:

```
PIV_OK
PIV_INVALID_CARD_HANDLE
PIV_CARD_READER_ERROR
```

3.3 Entry Points for Cryptographic Operations

3.3.1 pivCrypt

Purpose: Perform a cryptographic operation[2] such as encryption or signing on a sequence of bytes. Appendix C, Part 1, describes recommended procedures for PIV algorithm identifier discovery.

Prototype:

```
status_word pivCrypt(

   IN handle               cardHandle,
   IN byte                 algorithmIdentifier,
   IN byte                 keyReference,
   IN sequence of byte     algorithmInput,
   IN LONG                 inputLength,
   OUT sequence of byte    algorithmOutput,
   OUT LONG                outputLength
);
```

Parameters:

cardHandle Opaque identifier of the card to be acted upon as returned by pivConnect.

algorithmIdentifier Identifier of the cryptographic algorithm to be used for the cryptographic operation. See Table 6-2 and 6-3 in SP 800-78[4].

keyReference Identifier of the on-card key to be used for the cryptographic operation. See Table 6-1 and 6-3 in SP 800-78.

algorithmInput Sequence of bytes used as the input to the cryptographic operation.[3]

[2] The pivCrypt function does not perform any cryptographic operations itself. It provides the interface to the GENERAL AUTHENTICATE command to perform cryptographic operations on card. All cryptographic operations on the client side are performed outside the PIV middleware.

`inputLength`	Length of the algorithm input.
`algorithmOutput`	Sequence of bytes output by the cryptographic operation.
`outputLength`	Length of the algorithm output.

Return Codes:

```
PIV_OK
PIV_INVALID_CARD_HANDLE
PIV_INVALID_KEYREF_OR_ALGORITHM
PIV_SECURITY_CONDITIONS_NOT_SATISFIED
PIV_INPUT_BYTES_MALFORMED
PIV_CARD_READER_ERROR
```

The `PIV_INPUT_BYTES_MALFORMED` error condition indicates that some property of the data to be processed such as the length or padding was inappropriate for the requested cryptographic algorithm or key.

3.4 Entry Points for Credential Initialization and Administration

3.4.1 pivPutData

Purpose: Replace the entire data content of the named data object with the provided data.

Prototype:

```
status_word pivPutData(
   IN handle            cardHandle,
   IN string            OID,
   IN LONG              oidLength,
   IN sequence of byte  data,
   OUT LONG             dataLength
);
```

Parameters:

`cardHandle`	Opaque identifier of the card to be acted upon as returned by pivConnect.
`OID`	Object identifier of the object whose data content is to be replaced coded as a string; for example, “2.16.840.1.101.3.7.1.1.2.2.1”. See Table 2, Part 1.
`oidLength`	Length of the object identifier.
`data`	Data to be used to replace in its entirety the data content of the named data object.
`dataLength`	Length of the data to be retrieved from the PIV Card.

Return Codes:

```
PIV_OK
```

[3] The algorithmInput for RSA algorithms shall be restricted to the range 0 to n-1, where n is the RSA modulus.

```
PIV_INVALID_CARD_HANDLE
PIV_INVALID_OID
PIV_CARD_READER_ERROR
PIV_INSUFFICIENT_CARD_RESOURCE
PIV_SECURITY_CONDITIONS_NOT_SATISFIED
```

3.4.2 pivGenerateKeyPair

Purpose: Generates an asymmetric key pair in the currently selected card application.

If the provided key reference exists and the cryptographic mechanism associated with the reference data identified by this key reference is the same as the provided cryptographic mechanism, then the generated key pair replaces in entirety the key pair currently associated with the key reference.

Prototype:

```
status_word pivGenerateKeyPair(
   IN handle                cardHandle,
   IN byte                  keyReference,
   IN byte                  cryptographicMechanism,
   OUT sequence of byte     publicKey,
   OUT LONG                 KeyLength
);
```

Parameters:

cardHandle	Opaque identifier of the card to be acted upon as returned by pivConnect.
keyReference	The key reference of the generated key pair.
cryptographicMechanism	The type of key pair to be generated. See part 1, Table 4.
publicKey	BER-TLV data objects defining the public key of the generated key pair. See Table Part 2, Table 10.
KeyLength	Length of the public key related data retrieved from the PIV Card.

Return Codes:

```
PIV_OK
PIV_INVALID_CARD_HANDLE
PIV_SECURITY_CONDITIONS_NOT_SATISFIED
PIV_INVALID_KEY_OR_KEYALG_COMBINATION
PIV_UNSUPPORTED_CRYPTOGRAPHIC_MECHANISM
PIV_CARD_READER_ERROR
```

Appendix A—Terms, Acronyms, and Notation

A.1 Terms

Application Identifier A globally unique identifier of a card application as defined in ISO/IEC 7816-4.

Application Session The period of time within a card session between when a card application is selected and a different card application is selected or the card session ends.

Algorithm Identifier An PIV algorithm identifier is a one-byte identifier that specifies a cryptographic algorithm and key size. For symmetric cryptographic operations, the algorithm identifier also specifies a mode of operation (i.e., CBC or ECB).

BER-TLV Data Object A data object coded according to ISO/IEC 8825-2.

Card An integrated circuit card.

Card Application A set of data objects and card commands that can be selected using an application identifier.

Card Interface Device An electronic device that connects an integrated circuit card and the card applications therein to a client application.

Card Reader Synonym for card interface device.

Client Application A computer program running on a computer in communication with a card interface device.

Data Object An item of information seen at the card command interface for which are specified a name, a description of logical content, a format and a coding.

Interface Device Synonym for card interface device.

Key Reference A PIV key reference is a one-byte identifier that specifies a cryptographic key according to its PIV Key Type. The identifier used in cryptographic protocols such as an authentication or a signing protocol.

Object Identifier A globally unique identifier of a data object as defined in ISO/IEC 8824-2.

Reference Data Cryptographic material used in the performance a cryptographic protocol such as an authentication or a signing protocol. The reference data length is the maximum length of a password or PIN. For algorithms, the reference data length is the length of a key.

Status Word Two bytes returned by an integrated circuit card after processing any command that encodes the success of or errors encountered during said processing.

Template A (constructed) BER-TLV data object whose value field contains specific BER-TLV data objects.

A.2 Acronyms

AID	Application Identifier
API	Application Programming Interface
ASN.1	Abstract Syntax Notation
BER	Basic Encoding Rules
FIPS	Federal Information Processing Standards
FISMA	Federal Information Security Management Act
GSC-IS	Government Smart Card Interoperability Specification
HSPD	Homeland Security Presidential Directive
ICC	Integrated Circuit Card
IEC	International Electrotechnical Commission
INCITS	InterNational Committee for Information Technology Standards
ISDN	Integrated Services Digital Network
ISO	International Standards Organization
ITL	Information Technology Laboratory
LSB	Least Significant Bit
MSB	Most Significant Bit
NIST	National Institute of Standards and Technology
OID	Object Identifier
OMB	Office of Management and Budget
PIN	Personal Identification Number
PIV	Personal Identity Verification
PIX	Proprietary Identifier eXtension

PKCS	Public Key Cryptography Standard
PKI	Public Key Infrastructure
RFU	Reserved for Future Use
RID	Registered application provider IDentifier
SP	Special Publication
TLV	Tag-Length-Value

A.3 Notation

The sixteen hexadecimal digits shall be denoted using the alphanumeric characters 0, 1, 2…, A, B, C, D, E, and F. A byte consists of two hexadecimal digits, for example, '2D'. A sequence of bytes may be enclosed in single quotation marks, for example 'A0 00 00 01 16' rather than given as a sequence of individual bytes, 'A0' '00' '00' '01' '16'.

A byte can also be represented by bits b8 to b1, where b8 is the most significant bit (MSB) and b1 is the least significant bit (LSB) of the byte. In textual or graphic representations, the leftmost bit is the MSB. Thus, for example, the most significant bit, b8, of '80' is 1 and the least significant bit, b1, is 0.

All bytes specified as RFU shall be set to '00' and all bits specified as reserved for future use shall be set to 0.

All lengths shall be measured in number of bytes unless otherwise noted.

Data objects in templates are described as being mandatory (M), optional (O) or conditional (C). 'Mandatory' means the data object shall appear in the template. 'Optional' means the data object may appear in the template. In the case of 'conditional' data objects, the conditions under which they are required are provided in a footnote to the table.

In other tables the M/O column identifies properties of the PIV Card Application that shall be present (M) or may be present (O).

BER-TLV data object tags are represented as byte sequences as described above. Thus, for example, '4F' is the interindustry data object tag for an application identifier and '7F 60' is the interindustry data object tag for the biometric information template.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this standard are to be interpreted as described in IETF RFC 2119, Key Words for Use in RFCs to Indicate Requirement Levels [3].

Appendix B—References

[1] Federal Information Processing Standard 201-1, Change Notice 1, *Personal Identity Verification (PIV) of Federal Employees and Contractors*, March 2006. (See http://csrc.nist.gov)

[2] ISO/IEC 8825-1:2002, *Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).*

[3] IETF RFC 2119, *Key Words for Use in RFCs to Indicate Requirement Levels*, March, 1997.

[4] NIST Special Publication 800-78-1, *Cryptographic Algorithms and Key Sizes for Personal Identity Verification*, August 2007. (See http://csrc.nist.gov)

NIST Special Publication 800-73-2

Interfaces for Personal Identity Verification – Part 4: The PIV Transitional Interface and Data Model Specification

NIST

National Institute of Standards and Technology
U.S. Department of Commerce

Ramaswamy Chandramouli
James F. Dray
Hildegard Ferraiolo
Scott B. Guthery
William MacGregor
Ketan Mehta

INFORMATION SECURITY

Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD, 20899-8930

September 2008

U.S. Department of Commerce
Carlos M. Gutierrez, Secretary

National Institute of Standards and Technology
James M. Turner, Deputy Director

Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the Nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analyses to advance the development and productive use of information technology. ITL's responsibilities include the development of management, administrative, technical, and physical standards and guidelines for the cost-effective security and privacy of non-national security-related information in Federal information systems. This special publication 800-series reports on ITL's research, guidelines, and outreach efforts in information system security, and its collaborative activities with industry, government, and academic organizations.

National Institute of Standards and Technology Special Publication 800-73-2, Part 4,
18 pages, (September 2008)

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

NIST makes no representation as to whether or not one or more implementations of SP 800-73-2 is/are covered by existing patents.

Acknowledgements

The authors (Ramaswamy Chandramouli, James Dray, Hildegard Ferraiolo, William MacGregor of NIST, Ketan Mehta of Mehta Inc., and Scott Guthery of Mobile Mind Inc) wish to thank their colleagues who reviewed drafts of this document and contributed to its development. Special thanks to the Government Smart Card Interagency Advisory Board (GSC-IAB) and InterNational Committee for Information Technology Standards (INCITS) for providing detailed technical inputs to the SP 800-73-2 development process. The authors also gratefully acknowledge and appreciate the many contributions from the public and private sectors whose thoughtful and constructive comments improved the quality and usefulness of this publication.

TABLE OF CONTENTS

List of Appendices

List of Tables

1. Introduction

The Homeland Security Presidential Directive 12 (HSPD-12) called for a common identification standard to be adopted governing the interoperable use of identity credentials to allow physical and logical access to Federal government locations and systems. The Personal Identity Verification (PIV) of Federal Employees and Contractors, Federal Information Processing Standard 201 (FIPS 201) [1] was developed to establish standards for identity credentials. Special Publication 800-73-2 (SP 800-73-2) specifies interface requirements for retrieving and using the identity credentials from the PIV Card[1] and is a companion document to FIPS 201.

1.1 Authority

This document has been developed by the National Institute of Standards and Technology (NIST) in furtherance of its statutory responsibilities under the Federal Information Security Management Act (FISMA) of 2002, Public Law 107-347.

NIST is responsible for developing standards and guidelines, including minimum requirements, for providing adequate information security for all agency operations and assets, but such standards and guidelines shall not apply to national security systems. This recommendation is consistent with the requirements of the Office of Management and Budget (OMB) Circular A-130, Section 8b(3), Securing Agency Information Systems, as analyzed in A-130, Appendix IV: Analysis of Key Sections. Supplemental information is provided A-130, Appendix III.

This recommendation has been prepared for use by federal agencies. It may be used by non-governmental organizations on a voluntary basis and is not subject to copyright though attribution is desirable. Nothing in this document should be taken to contradict standards and guidelines made mandatory and binding on Federal agencies by the Secretary of Commerce under statutory authority. Nor should this recommendation be interpreted as altering or superseding the existing authorities of the Secretary of Commerce, Director of the Office of Management and Budget (OMB), or any other Federal official.

1.2 Purpose

FIPS 201 defines procedures for the PIV lifecycle activities including identity proofing, registration, PIV Card issuance, and PIV Card usage. FIPS 201 also specifies that the identity credentials must be stored on a smart card. SP 800-73-2 contains technical specifications to interface with the smart card to retrieve and use the identity credentials. The specifications reflect the design goals of interoperability and PIV Card functions. The goals are addressed by specifying a PIV data model, card edge interface, and application programming interface. Moreover, SP 800-73-2 enumerates requirements where the standards include options and branches. The specifications go further by constraining implementers' interpretations of the normative standards. Such restrictions are designed to ease implementation, facilitate interoperability, and ensure performance, in a manner tailored for PIV applications.

[1] A physical artifact (e.g., identity card, "smart" card) issued to an individual that contains stored identity credentials (e.g., photograph, cryptographic keys, biometric data) so that the claimed identity of the cardholder can be verified against the stored credentials by another person (human readable and verifiable) or an automated process (computer readable and verifiable).

1.3 Scope

SP 800-73-2 specifies the PIV data model, Application Programming Interface (API) and card interface requirements necessary to comply with the use cases, as defined in Section 6 of FIPS 201 and further elaborated in Appendix B of SP 800-73-2 Part 1. Interoperability is defined as the use of PIV identity credentials such that client-application programs, compliant card applications, and compliant integrated circuits cards (ICC) can be used interchangeable by all information processing system across Federal agencies.

Part 2, 3 and 4 of SP 800-73-2 describes two realizations of the client application programming and card command interfaces for personal identity verification: the transitional interfaces (this Part 4) and the end-point interfaces (Part 2 and 3). The transitional interface may be used by agencies with an existing identity card program as an optional intermediate step in evolving to the end-point interfaces.

This part, Special Publication 800-73-2, Part 4 *The PIV Transitional Interfaces and Data Model Specification* contains informative links to specifications of the transitional PIV card command interface and client application programming interface of the transitional PIV card. Part 4 also describes the PIV Data Model that is common between End-Point and transitional interface specifications.

1.4 Audience and Assumptions

This document is targeted at Federal agencies and implementers of PIV systems. Readers are assumed to have a working knowledge of smart card standards and applications.

1.5 Document Overview and Structure

All sections in this document are *normative* (i.e., mandatory for compliance) unless specified as *informative* (i.e., non-mandatory). Following is the structure of this document:

Part 4 is organized as follows:

+ Section 1, *Introduction*, provides the purpose, scope, audience, and assumptions of the document and outlines its structure.

+ Section 2: *Common Data Model and Migration Considerations* provides the specification that is common to both the transitional and end-point interfaces. Section 2 also includes guidelines as to strategies for migrating from the transitional interfaces to the end-point interfaces.

+ Section 3: *The Transitional Interfaces* provides links to transitional interface specification that are implemented today by agencies with legacy GSC-IS based card deployments. This section is informative.

- Appendix A, *Terms, Acronyms, and Notation*, contains the list of Terms and Acronyms used in this document and explains notation in use. This section is informative.

- Appendix B, *References*, contains the list of documents used as references by this document. This section is informative.

2. Overview and Migration Considerations

2.1 Migration Considerations

SP 800-73-2 Parts 1 - 4 provide two interface specifications: 1) a transitional card specification as described in this Part 4; and 2) a FIPS 201 End-Point card specification as described in Parts 1- 3 of SP 800-73-2. Part 4 interface specifications are informative PIV profile derived from the Government Smart Card Interoperability Specification (GSC-IS), Version 2.1 [2]. It presents one possible path that agencies with existing GSC-IS based smart card deployments may choose to follow during the transition to End-Point PIV card deployment. All agencies must ultimately comply with End-Point specifications in accordance with the schedule provided by the Office of Management and Budget (OMB). End-Point deployment is therefore the end state of each agency's transition plan.

Agencies may either elect to implement an approved transitional specification as specified in this document (Part 4), particularly when migrating from currently widely implemented identity card architectures to the End-Point specifications described in Parts 1- 3 of SP 800-73-2, or to implement the End-Point specification directly. NIST supports agency efforts towards government-wide PIV-End-Point interoperability described in the Parts 1 - 3 specification. NIST also supports transition specifications of Part 4 for widely implemented deployments as they migrate towards the End-Point specifications.

The migration path to End-Point implementation is based on continuity of the PIV data model. Exactly the same data appear on both the transitional and end-point interfaces. Therefore, description of the data for personal identity verification, the PIV data model, is duplicated from the Part 1 (Section 3) in Section 2.2 below[2].

Specific considerations associated with this migration path are highlighted below:

+ The transitional specifications present a subset of the dual GSC-IS card edge interfaces. The End-Point specifications present a unified card edge interface that is technology independent and compliant with existing international standards.
+ The End-Point specifications provide limited credential administration functionality. A unified and interoperable card management solution between issuing domains including the loading of new card applications is not provided.
+ Named data objects within the data model may be directly accessed. If a data object is managed by the default application, it can be retrieved directly without selecting the application. This avoids a requirement to search through discovery to get named data objects. Otherwise, the (non-default) application managing the data object is selected and the data object is retrieved from this application. The GET DATA command described in Part 2 retrieves a data object in one command.
+ The data model including the data model namespace is controlled by NIST and hence change management of well known and interoperable data objects will be managed by NIST in the process of managing the overall data model. As a first step in namespace management, the data object identifiers of GSC-IS and transitional systems in the range

[2] Although the same data objects are present on the end-point and transitional interfaces, different representations for the same data objects may be used.

'0000' through '9FFF' will be explicitly managed by NIST and data object identifiers of GSC-IS and transitional systems in the range 'A000' through 'FFFF' are placed under control of the card issuer.

+ Each application managing one or more of the directly addressable data model data objects will have a version number enabling the relying application to determine the level of the information contained within the object. The version of the End-Point PIV Card Application is encoded in its full Application IDentifier (AID) which is returned when this application is selected. This is in addition to the Card Capability Container (CCC) style data model naming facility carried over from GSC-IS.

+ Agency-specific applications can be included on cards containing PIV applications. These applications may define and manage their own namespaces that are used when the application is used. Such applications will have application identifiers outside the application namespace managed by NIST; that is, application identifiers not rooted on the NIST Registered application provider IDentifier (RID).

2.2 PIV Data Model

Table 1 defines a high level view of the data model. Each on-card storage container is labeled either as Mandatory (M) or Optional (O). This data model is designed to enable and support dual interface cards. Note that access conditions based on the interface mode (contact vs. contactless) take precedence over all Access Rules defined in Table 1, Column 3.

Table 1. Data Model Containers

Container Name	Container ID	Access Rule for Read	Contact / Contactless[3]	M/O
Card Capability Container	0xDB00	Always	Contact	M
Cardholder Unique Identifier	0x3000	Always	Contact and Contactless	M
X.509 Certificate for PIV Authentication	0x0101	Always	Contact	M
Cardholder Fingerprints	0x6010	PIN	Contact	M
Security Object	0x9000	Always	Contact	M
Cardholder Facial Image	0x6030	PIN	Contact	O
Printed Information	0x3001	PIN	Contact	O
X.509 Certificate for Digital Signature	0x0100	Always	Contact	O
X.509 Certificate for Key Management	0x0102	Always	Contact	O
X.509 Certificate for Card Authentication	0x0500	Always	Contact and Contactless	O
Discovery Object	0x6050	Always	Contact and Contactless	O

Part 1, Appendix A provides a detailed spreadsheet for the data model. ContainerIDs and Tags within the containers for each data object are defined by this data model and in accordance with SP 800-73-2 naming conventions.

[3] Contact interface mode means the container is accessible through contact interface only. Contact and contactless interface mode means the container can be accessed from either interface.

A PIV Card Application shall contain five mandatory interoperable data objects and may contain six optional interoperable data objects. The five mandatory data objects for interoperable use are as follows:

1. Card Capability Container
2. Cardholder Unique Identifier
3. X.509 Certificate for PIV Authentication
4. Cardholder Fingerprints
5. Security Object

The six optional data objects for interoperable use are as follows:

1. Cardholder Facial Image
2. Printed Information
3. X.509 Certificate for Digital Signature
4. X.509 Certificate for Key Management
5. X.509 Certificate for Card Authentication
6. Discovery Object

2.3 Mandatory Data Elements

The five mandatory data objects support FIPS 201 minimum mandatory compliance.

2.3.1 Card Capability Container

The Card Capability Container (CCC) is mandatory for compliance with GSC-IS. It supports minimum capabilities for retrieval of data model and application information.

The data model of the PIV Card Application shall be identified by data model number 0x10. Deployed applications use 0x00 through 0x04. This enables the GSC-IS application domain to correctly identify a new data model name space and structure as defined in this document.

2.3.2 Cardholder Unique Identifier

The Cardholder Unique Identifier (CHUID) data object is defined in accordance with the Technical Implementation Guidance: Smart Card Enabled Physical Access Control Systems (TIG SCEPACS) [4]. For this specification, the CHUID is common between the contact and contactless chips. For dual chip implementations, the CHUID is copied in its entirety between the two chips.

In addition to the requirements specified in TIG SCEPACS, the CHUID on the PIV Card shall meet the following requirements:

+ The Buffer Length field is an optional TLV element. This element is the length in bytes of the entire CHUID, excluding the Buffer Length element itself, but including the CHUID's Asymmetric Signature element. The calculation of the asymmetric signature must exclude the Buffer Length element if it is present.

+ The Federal Agency Smart Credential Number (FASC-N) shall be in accordance with TIG SCEPACS[4]. A subset of FASC-N, the FASC-N Identifier, shall be the unique identifier as described in [4, 6.6]: “The combination of an Agency Code, System Code, and Credential Number is a fully qualified number that is uniquely assigned to a single individual". The Agency Code is assigned to each Department or Agency by Special Publication 800-87 *(*SP 800-87) *Codes for the Identification of Federal and Federally-Assisted Organizations* [5]. The subordinate System Code and Credential Number value assignment is subject to Department or Agency policy, provided that the FASC-N identifier (i.e. the concatenated Agency Code, System Code, and Credential Number) is unique for each card. The same FASC-N value shall be used in all the PIV data objects that include the FASC-N. To eliminate unnecessary use of the SSN[4], the FASC-N's Person Identifier (PI) field should not encode the SSN. TIG SCEPACS also specifies PACS interoperability requirements in section 2.1, 10th paragraph of [4, 2.1]: "For full interoperability of a PACS it must at a minimum be able to distinguish fourteen digits (i.e., a combination of an Agency Code, System Code, and Credential Number) when matching FASC-N based credentials to enrolled card holders."

+ The Global Unique Identifier (GUID) field must be present, and may include either an issuer assigned IPv6 address or be coded as all zeros. The GUID is included to enable future migration away from the FASC-N into a robust numbering scheme for all issued credentials.

+ The DUNS and Organizational Code fields are optional.

+ The Authentication Key Map[5] is specified as an optional field which enables the application to discover the key reference.

+ The Expiration Date is mapped to the reserved for future use (RFU) tag 0x35, keeping that within the existing scope of the TIG SCEPACS specification. This field shall be 8 bytes in length and shall be encoded as YYYYMMDD.

+ The CHUID is signed in accordance with FIPS 201. The card issuer's digital signature key shall be used to sign the CHUID and the associated certificate shall be placed in the signature field of the CHUID.

2.3.3 X.509 Certificate for PIV Authentication

The X.509 Certificate for PIV Authentication and its associated private key, as defined in FIPS 201, is used to authenticate the card and the cardholder. The read access control rule for the X.509 Certificate for PIV Authentication is "Always," meaning the certificate can be read without access control restrictions. The Public Key Infrastructure (PKI) cryptographic function (see Table 3) is protected with a "PIN" access rule. In other words, private key operations using the PIV Authentication Key require the Personal Identification Number (PIN) to be submitted, but a successful PIN submission enables multiple private key operations without additional cardholder consent.

2.3.4 Cardholder Fingerprints

The fingerprint data object specifies the primary and secondary fingerprints in accordance with the FIPS 201. The Common Biometric Exchange Formats Framework (CBEFF) header shall contain the FASC-N and shall require the Integrity Option. The header shall not require the Confidentiality Option.

[4] See the attachment to OMB M-07-16, Section 2: "Reduce the Use of Social Security Numbers".

[5] The Authentication Key Map is deprecated. It will be eliminated in a future revision of SP 800-73.

2.3.5 Security Object

The Security Object is in accordance with Appendix C of PKI for Machine Readable Travel Documents (MRTD) Offering ICC Read-Only Access Version 1.1 [6]. Tag 0xBA is used to map the ContainerIDs in the PIV data model to the 16 Data Groups specified in the MRTD. The mapping enables the Security Object to be fully compliant for future activities with identity documents.

The “DG-number-to-Container-ID” mapping object TLV in tag 0xBA encapsulates a series of three byte triples - one for each PIV data object included in the Security Object. The first byte is the Data Group (DG) number, and the second and third bytes are the most and least significant bytes (respectively) of the Container ID value. The DG number assignment is arbitrary; however, the same number assignment applies to the DataGroupNumber(s) in the DataGroupHash(es). This will ensure that the ContainerIDs in the mapping object refers to the correct hash value in the Security Object (0xBB).

The 0xBB Security Object is formatted according to the MRTD [5, Appendix C]. The LDS Security Object itself must be in ASN.1 DER format, formatted as specified in [5, Appendix C.2]. This structure is then inserted into the encapContentInfo field of the Cryptographic Message Syntax (CMS) object specified in [5, Appendix C.1].

The card issuer's digital signature key used to sign the CHUID shall also be used to sign the Security Object. The signature field of the Security Object, Tag 0xBB shall omit the issuer’s certificate, since it is included in the CHUID. At a minimum, unsigned data objects, such as the Printed Information data object, shall be included in the Security Object if present. For maximum protection against credential splicing attacks (credential substitution), it is recommended, however, that all PIV data objects, except the PIV X.509 certificates, be included in the Security Object.

2.4 Optional Data Elements

The six optional data elements of FIPS 201, when implemented, shall conform to the specifications provided in this document.

2.4.1 Cardholder Facial Image

The photo on the chip supports human verification only. It is not intended to support facial recognition systems for automated identity verification.

2.4.2 Printed Information

All FIPS 201 mandatory information printed on the card is duplicated on the chip in this data object. The Security Object enforces integrity of this information according to the issuer. This provides specific protection that the card information must match the printed information, mitigating alteration risks on the printed media.

2.4.3 X.509 Certificate for Digital Signature

The X.509 Certificate for Digital Signature and its associate private key, as defined in FIPS 201, support the use of digital signatures for the purpose of document signing. The read access control rule for the X.509 Certificate is “Always”, meaning the certificate can be read without access control restrictions. The Public Key Infrastructure (PKI) cryptographic function is protected with a “PIN Always” access rule. In other words, the PIN must be submitted every time immediately before a

Digital Signature Key operation. This ensures cardholder participation every time the private key is used for digital signature generation.

2.4.4 X.509 Certificate for Key Management

The X.509 Certificate for Key Management and its associate private key, as defined in FIPS 201, support the use of encryption for the purpose of confidentiality. This key pair is escrowed by the issuer for key recovery purposes. The read access control rule for the X.509 Certificate is “Always”, meaning the certificate can be read without access control restrictions. The PKI cryptographic function is protected with a “PIN” access rule. In other words, once the PIN is submitted, subsequent *Key Management Key* operations can be performed without requiring the PIN again. This enables multiple private key operations without additional cardholder consent.

2.4.5 X.509 Certificate for Card Authentication

FIPS 201 specifies the optional Card Authentication Key (CAK) as an asymmetric or symmetric key that is used to support additional physical access applications. For an asymmetric CAK, the read access control rule of the corresponding X.509 Certificate for Card Authentication is “Always”, meaning the certificate can be read without access control restrictions. Private (asymmetric) key operations or secret symmetric cryptographic operation is defined as “Always”. In other words, the private or secret key can be used without access control restrictions. With extremely high probability, each PIV Card shall contain a unique CAK.

2.4.6 Discovery Object

The Discovery Object, if implemented, is the 0x7E interindustry ISO/IEC 7816-6 template that nests interindustry data objects. For the Discovery Object, the 0x7E template nests two BER-TLV structured interindustry data elements: 1) tag 0x4F contains the AID of the PIV Card Application and 2) tag 0x5F2F lists the PIN Usage Policy.

+ Tag 0x4F encodes the PIV Card Application AID as follows:
  {'4F 0B A0 00 00 03 08 00 00 10 00 01 00'}

+ Tag 0x5F2F encodes the PIN Usage Policy as follows:

  First byte: 0x40 indicates that the PIV Card Application PIN alone satisfies the PIV Access Control Rules (ACRs) for command execution[6] and object access.

  0x60 indicates that both the PIV Card Application PIN and Global PIN satisfy the PIV ACRs for command execution and PIV data object access.

  Bits 5 through 1 of the first byte are RFU.

  The second byte of the PIN Usage Policy encodes the cardholder’s PIN preference for PIV Cards with both the PIV Card Application PIN and the Global PIN enabled:

  Second byte: 0x10 indicates that the PIV Card Application PIN is the primary PIN used to satisfy the PIV ACRs for command execution and object access.

[6] Command execution pertains to the VERIFY APDU and optionally to the CHANGE REFERENCE DATA APDU.

0x20 indicates that the Global PIN is the primary PIN to satisfy the PIV ACRs for command execution and object access.

Note: If the first byte is set to 0x40, then second byte is RFU and shall be set to 0x00.

PIV Card Application that satisfy the PIV ACRs for PIV data object access and command execution [7]with both PIV Card Application PIN and Global PIN shall implement the Discovery Object with the PIN Usage Policy set to 0x60 zz where zz is set to either 0x10 or 0x20.

The encoding of the 0x7E Discovery Object is as follows:

> {'7E 12' {{'4F 0B A0 00 00 03 08 00 00 10 00 01 00'} {'5F 2F 02 xx yy'}}}, where xx and yy encode the first and second byte of the PIN Usage Policy as described in this section.

The Security Object enforces integrity of Discovery Object according to the issuer.

[7] [7] Command execution pertains to the VERIFY APDU and optionally to the CHANGE REFERENCE DATA APDU.

3. Transition Card Interfaces

3.1 Middleware Application Programming Interface

Reference [8] is an example of a transitional (GSC-IS) middleware API specification.

3.2 Card Edge Commands

Reference [9] is an example of a transitional (GSC-IS) card edge command specification.

Appendix A—Terms, Acronyms, and Notation

A.1 Terms

Card	An integrated circuit card.
Card Application	A set of data objects and card commands that can be selected using an application identifier.
Data Object	An item of information seen at the card command interface for which are specified a name, a description of logical content, a format and a coding.

A.2 Acronyms

APDU	Application Protocol Data Unit
BSI	Basic Services Interface
CBEFF	Common Biometric Exchange Formats Framework
CCC	Card Capability Container
CHUID	Cardholder Unique IDentifier
FASC-N	Federal Agency Smart Credential Number
FIPS	Federal Information Processing Standards
FISMA	Federal Information Security Management Act
GSC-IAB	Government Smart Card Interagency Advisory Board
GSC-IS	Government Smart Card Interoperability Specification
GUID	Global Unique Identification Number
HSPD	Homeland Security Presidential Directive
ICC	Integrated Circuit Card
IEC	International Electrotechnical Commission
INCITS	InterNational Committee for Information Technology Standards
ISO	International Standards Organization
ITL	Information Technology Laboratory
LSB	Least Significant Bit

MRTD	Machine Readable Travel Document
MSB	Most Significant Bit
NIST	National Institute of Standards and Technology
OMB	Office of Management and Budget
PACS	Physical Access Control System
PIN	Personal Identification Number
PIV	Personal Identity Verification
PKCS	Public Key Cryptography Standard
PKI	Public Key Infrastructure
RFU	Reserved for Future Use
RID	Registered application provider IDentifier
RSA	Rivest, Shamir, Adleman
SP	Special Publication
TIG	Technical Implementation Guidance
VM	Virtual Machine

A.3 Notation

The sixteen hexadecimal digits shall be denoted using the alphanumeric characters 0, 1, 2…, A, B, C, D, E, and F. A byte consists of two hexadecimal digits, for example, '2D'. A sequence of bytes may be enclosed in single quotation marks, for example 'A0 00 00 01 16' rather than given as a sequence of individual bytes, 'A0' '00' '00' '01' '16'.

A byte can also be represented by bits b8 to b1, where b8 is the most significant bit (MSB) and b1 is the least significant bit (LSB) of the byte. In textual or graphic representations, the leftmost bit is the MSB. Thus, for example, the most significant bit, b8, of '80' is 1 and the least significant bit, b1, is 0.

All bytes specified as RFU shall be set to '00' and all bits specified as reserved for future use shall be set to 0.

All lengths shall be measured in number of bytes unless otherwise noted.

Data objects in templates are described as being mandatory (M), optional (O) or conditional (C). 'Mandatory' means the data object shall appear in the template. 'Optional' means the data object may appear in the template. In the case of 'conditional' data objects, the conditions under which they are required are provided in a footnote to the table.

In other tables the M/O column identifies properties of the PIV Card Application that shall be present (M) or may be present (O).

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this standard are to be interpreted as described in IETF RFC 2119, Key Words for Use in RFCs to Indicate Requirement Levels [7].

Appendix B—References

[1] Federal Information Processing Standard 201-1, Change Notice 1, *Personal Identity Verification (PIV) of Federal Employees and Contractors*, March 2006. (See http://csrc.nist.gov)

[2] Government Smart Card Interoperability Specification, Version 2.1, NIST Interagency Report 6887 – 2003 Edition, July 16, 2003.

[3] ISO/IEC 7816 (Parts 4, 5, 6, 8, and 9), *Information technology — Identification cards — Integrated circuit(s) cards with contacts.*

[4] PACS v2.2, *Technical Implementation Guidance: Smart Card Enabled Physical Access Control Systems*, Version 2.2, The Government Smart Card Interagency Advisory Board's Physical Security Interagency Interoperability Working Group, July 27, 2004. http://www.smart.gov/information/TIG_SCEPACS_v2.2.pdf

[5] NIST Special Publication 800-87, *Codes for the Identification of Federal and Federally-Assisted Organizations*, April 2008. (See http://csrc.nist.gov)

[6] *PKI for Machine Readable Travel Documents Offering ICC Read-Only Access Version - 1.1* Date - October 01, 2004. Published by authority of the Secretary General, International Civil Aviation Organization

[7] IETF RFC 2119, "Key Words for Use in RFCs to Indicate Requirement Levels," March, 1997.

[8] *DoD CAC Middleware Requirements Release 3.0*, Version 1.0, Access Card Office, March 21, 2006. http://www.smart.gov/iab/documents/DoDcacMiddlewareRequirements.pdf.

[9] *DoD Implementation Guide for CAC Next Generation (NG)*, Version 2.6, DMDC Card Technologies & Identity Solutions Division (CTIS), November, 2006. http://www.smart.gov/iab/documents/CACngImplementationGuide.pdf.