NIST Special Publication 800-73-3

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

Ramaswamy Chandramouli
David Cooper
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

February 2010

U.S. Department of Commerce
Gary Locke, Secretary

National Institute of Standards and Technology
Dr. Patrick D. Gallagher, 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 800-73-3, Part 2, 36 pages (February 2010)

Acknowledgements

The authors (Ramaswamy Chandramouli, David Cooper, James Dray, Hildegard Ferraiolo, William MacGregor of NIST, Ketan Mehta of Booz Allen Hamilton and Scott Guthery of HID Global) 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 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 800-73-3 (SP 800-73-3) 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 in 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 800-73-3 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 800-73-3 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 800-73-3 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 800-73-3 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 800-73-3 Part 2 – *End-Point PIV Card Application Card Command 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.

Readers should also be aware of SP 800-73-3 Part 1, Section I, for the Revision History of SP800-73, Section II, which details Configuration Management Recommendations, and Section III, which specifies NPIVP Conformance Testing Procedures.

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 Concepts and Constructs*, 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 800-73-3 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 programming 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, OIDs, and values of algorithm identifiers, key references, 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. In essence, when changing from one application to another, the global security status indicators shall remain unchanged.

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 security status indicators associated with the PIV Card Application PIN, the PIN Unblocking Key (PUK), and the PIV Card Application Administrator are application security status indicators for the PIV Card Application, whereas the security status indicator associated with the Global PIN is a global 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 (PIV Card Application PINs and PUKs) 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 PIV Card Application PIN shall be limited to values 0x30 – 0x39, the ASCII values for the decimal digits '0' – '9'. The bytes comprising the PUK shall be limited to the values 0x00 – 0xFE (i.e., 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'

If the Global PIN is used by the PIV Card Application then the above encoding, length, and padding requirements for the PIV Card Application PIN shall apply to the Global PIN.

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.[3]

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 an L_c value of '03'.

[3] It is assumed that the GET DATA command will use the GET RESPONSE command to accomplish the reading of larger PIV Data Objects. The GET RESPONSE command is illustrated in A.3.1 (Command 3).

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 able to 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 able to 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'[4] 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.

[4] 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 references '80' and '81' specific to the PIV Card Application (i.e., local key reference) and 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. The ability to change reference data associated with key references '81' and '00' using the PIV Card Application CHANGE REFERENCE DATA command is optional.

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 data 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 PIN to its initial value and changes the PIN's reference data. The command enables recovery of the PIV Card Application PIN in the case that the cardholder has forgotten the PIV Card Application PIN.

The only key reference allowed in the P1 parameter of the RESET RETRY COUNTER command is the PIV Card Application PIN.

If the current value of the PUK's retry counter is zero, then the PIN's retry counter shall not be reset, and the PIV Card Application shall return the status word '69 83'.

If the card command succeeds, then the PIN's retry counter shall be set to its reset retry value. Optionally, the PUK's retry counter may be set to its initial reset retry value. The security status of the PIN's key reference shall not be changed.

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

The initial retry counter associated with the PUK; i.e., the number of failures of the RESET RETRY COUNTER command before the PUK's retry counter reaches zero, is issuer dependent.

If 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 PIN and shall return the status word '6A 80'.

Command Syntax

CLA	'00'
INS	'2C'
P1	'00'
P2	Key reference '80'. 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 Section 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.[5]

The GENERAL AUTHENTICATE command shall be used with the PIV authentication Keys ('9A', '9B', '9E') 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 with the PIV Digital Signature Key ('9C') to realize the signing functionality on the PIV client application programming interface. Data to be signed is expected to be hashed off card. Appendix A, Section A.3 illustrates the use of the GENERAL AUTHENTICATE command for signature generation.

The GENERAL AUTHENTICATE command shall be used with the PIV Key Management Key ('9D') and the retired PIV Key Management Keys ('82' – '95') to realize key establishment schemes specified in SP 800-78 (ECDH and RSA). Appendix A.4 illustrates the use of the GENERAL AUTHENTICATE command for key establishment schemes aided by the PIV Card Application.

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 [6]
P2	Key reference: • See Table 3, Part 1 for key references of retired private Key Management Keys • See Table 6-1, SP 800-78 for all other key references
L_c	Length of data field
Data Field	See Table 6
L_e	Absent or length of expected response

[5] For cryptographic operations with larger keys, e.g., RSA 2048, it is assumed that the GENERAL AUTHENTICATE command will use the GET RESPONSE command to return the complete result of the cryptographic operation. The GET RESPONSE command is illustrated in A.3.1 (Command 3).

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.
Exponentiation	'85'	C	A parameter used in ECDH key agreement 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, authentication-related data, signed data, shared secret, or transported key
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 Tables 7 and 8
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'

The public key data object in tag '86' is encoded as follows:

Tag	Length	Value
86	L	04 || X || Y (see Section 2.3.3 of [8])

Note: The octet '04' indicates that the X and Y coordinates of point P are encoded without the use of point compression. The length L is 65 bytes for points on Curve P-256 and 97 bytes for points on Curve P-384.

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[6]. The PIV Card Application decrypts the response using this reference data and the algorithm associated with the key reference (for example 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 9B 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.
	'90 00'	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 validating the certificate. Assuming the certificate is valid, the client application requests the PIV Card Application to sign a challenge using the private key associated with this certificate (i.e., key reference '9A') and the appropriate algorithm (e.g., algorithm identifier '06'), which can be determined from the certificate as described in Part 1, Appendix C.1. The response is

[6] The Card Management Key is the PIV Card Application Administration Key used for managing the PIV Card Application.

verified using the public key in the certificate. If the signature verifies, 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 when the X.509 Certificate for PIV Authentication includes a 1024-bit RSA public key.

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

Command	Response	Comment
'00 87 06 9A 88 7C 81 85 82 00 81 81 80 00 01 FF … 58 EA C3 00' ("…" represents 122 bytes of challenge data)		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. The challenge data, which in this example is encoded in accordance with the PKCS #1 v.1.5 signature padding scheme, is '00 01 FF ... 58 EA C3'. L_e is '00' to indicate that the expected length of the response data field is 256 bytes.
	'7C 81 83 82 81 80 88 0E BA … 41 E6 EE 90 00' ("…" represents 122 bytes of the signed challenge)	PIV Card Application returns the result of signing the challenge using the indicated key reference data and algorithm ('88 0E BA … 41 E6 EE').

A.3 Signature Generation with the Digital Signature Key

The GENERAL AUTHENTICATE command can be used to generate signatures. The pre-signature hash and padding (if applicable) is computed off card. The PIV Card Application receives the hashed value of the original message, applies the private signature key (key reference '9C'), and returns the resulting signature to the client application.

Listed below are the card commands sent to the PIV Card Application to generate a signature. It is assumed that the cardholder PIN has been successfully verified prior to sending the GENERAL AUTHENTICATE command.

A.3.1 RSA

This example illustrates signature generation using RSA 2048 (i.e., algorithm identifier '07'). Command chaining is used in the first command since the padded hash value sent to the card for signature generation is bigger than the length of the data field.

Command 1: (GENERAL AUTHENTICATE – first chain):

CLA	'10' indicating command chaining
INS	'87'
P1	'07'
P2	'9C'
L_c	Length of data field
Data Field	'7C' – L1 { '82' '00' '81' L2 {first part of the PKCS #1 v1.5 or PSS padded message hash value }}
L_e	Absent (no response expected)

Response 1:

Data Field	Absent
SW1-SW2	'90 00' (Status word)

Command 2: (GENERAL AUTHENTICATE – last chain):

CLA	'00' indicates last command of the chain.
INS	'87'
P1	'07'
P2	'9C'
L_c	Length of data field
Data Field	{second and last part of the PKCS #1 v1.5 or PSS padded message hash value}
L_e	Length of expected response

Response 2:

Data Field	'7C' – L1 {'82' L2 {first part of signature} }
SW1-SW2	'61 xx' where xx indicates the number of bytes remaining to send by the PIV Card Application

Command 3: (GET RESPONSE APDU):

CLA	'00'
INS	'C0'
P1	'00'
P2	'00'
L_e	xx Length of remaining response as indicated by previous SW1-SW2

Response 3:

Data Field	{second and last part of signature}
SW1-SW2	'90 00' (Status word)

A.3.2 ECDSA

The following example illustrates signature generation with ECDSA using ECC: Curve P-256 (i.e., algorithm identifier '11'). Command chaining is not used in this example, as the hash value fits into the data field of the command. Padding does not apply to ECDSA.

Command – GENERAL AUTHENTICATE

CLA	'00'
INS	'87'
P1	'11'
P2	'9C'
L_c	Length of data field
Data Field	'7C' – L1 { '82' '00' '81' L2 {hash value of message}}
L_e	Length of expected response

Response:

Data Field	'7C' – L1 {'82' L2 (r,s)} where • (r,s) is DER encoded with the following ASN.1 structure: Ecdsa-Sig-Value ::= SEQUENCE { r INTEGER, s INTEGER } • L1 is the length of tag '82' TLV structure • L2 is the length of the DER encoded Ecdsa-Sig-Value structure
SW1-SW2	'90 00' (Status word)

A.4 Key Establishment Schemes with the PIV Key Management Key

FIPS 201 specifies an optional public key pair and associated X.509 Certificate for Key Management. The Key Management Key (KMK) is further defined in SP 800-78, which defines two distinct key establishment schemes for the KMK:

1) RSA key transport and
2) Elliptic Curve Diffie-Hellman (ECDH) key agreement.

The use of the KMK for RSA key transport and ECDH key agreement is discussed in Sections A.4.1 and A.4.2, respectively.

A.4.1 RSA Key Transport

In general, RSA transport keys are used to establish symmetric keys, where a sender encrypts a symmetric key with the receiver's public key and sends the encrypted key to the receiver. The receiver decrypts the encrypted key with the corresponding private key. The decrypted symmetric key subsequently is used by both parties to protect further communication between them. Many types of security protocols employ

the RSA key transport technique. S/MIME for secure email and TLS for secure web communications are two of the many protocols employing RSA transport keys to distribute symmetric keys between entities.

A.4.1.1. RSA Key Transport with the PIV KMK

As specified in SP 800-78, the on-card private KMK can be an RSA transport key that complies with PKCS #1[9]. In the scenario described above, a sender encrypts a symmetric key with the KMK's public RSA transport key. The role of the on-card KMK private RSA transport key is to decrypt the sender's symmetric key on behalf of the cardholder and provide it to the client application cryptographic module.

A.4.1.1.1 The GENERAL AUTHENTICATE Command

Listed below are the card commands sent to the PIV Card to decrypt the symmetric key. It is assumed that the cardholder's PIN has been successfully verified prior to sending the GENERAL AUTHENTICATE command to the card.

Command 1 – GENERAL AUTHENTICATE (first chain)

CLA	'10' indicates command chaining
INS	'87'
P1	'07'
P2	'9D'
L_c	Length of data field
Data Field	'7C' – L1 {'82' '00' '81' L2 {first part of c}} where • c is the cipher text representative as defined in Section 5.1.2 of PKCS #1 v2.1
L_e	Absent (no response expected)

Response 1:

Data Field	Absent
SW1-SW2	'90 00' (Status word)

Command 2 – GENERAL AUTHENTICATE (last chain)

CLA	'00' indicates last command of the chain
INS	'87'
P1	'07'
P2	'9D'
L_c	Length of data field
Data Field	{second and last part of ciphertext representative c}}
L_e	Length of expected response

Response 2:

Data Field	'7C' – L1 {'82' L2 {first part of message representative m}} where m is as defined in PKCS #1 v2.1 [9] Section 5.1.2

SW1-SW2	'61 xx' where x indicates the number of bytes remaining to send

Command 3: (GET RESPONSE APDU):

CLA	'00'
INS	'C0'
P1	'00'
P2	'00'
L_e	xx Length of remaining response as indicated by previous SW1-SW2

Response 3:

Data Field	{second and last part of message representative m}
SW1-SW2	'90 00' (Status word)

A.4.2 Elliptic Curve Cryptography Diffie-Hellman

An ECDH key agreement scheme does not send an encrypted symmetric key to the participating entities. Instead, the two entities involved in the key agreement scheme compute a shared secret by combining their ECC private key(s) with the other party's public key(s). The resulting shared secret (Z) serves as an input to a Key Derivation Function (KDF), which each entity independently invokes to derive a common secret key. The secret key may be used as a session key or may be used to encrypt a session key.

A. 4.2.1 ECDH with the PIV KMK

The PIV Card supports ECDH key agreement by performing the Elliptic Curve Cryptography Cofactor Diffie-Hellman (ECC CDH) Primitive (see Section 5.7.1.2 of SP 800-56A [7]) using its ECC KMK private key and an ECC public key that is provided as input to the GENERAL AUTHENTICATE command. All other procedures required to complete the key agreement are performed by the cardholder's client application and its associated cryptographic module.

A.4.2.1.1 The GENERAL AUTHENTICATE Command

The sequence of commands to perform the ECC CDH Primitive from Section 5.7.1.2 of SP 800-56A with the private ECC KMK is illustrated below for ECC: Curve P-256:

Command – GENERAL AUTHENTICATE

CLA	'00'
INS	'87'
P1	'11'
P2	'9D'
L_c	Length of data field
Data Field	'7C' – L1 {'82' '00' '85' L2 { '04' || X || Y}} , where • '04 || X || Y' is the other party's public key, a point on Curve P-256, encoded without the use of point compression as described in Section 2.3.3 of [8].

	• The length of each coordinate (X and Y) is 32 bytes and • The value of L2 is 65 bytes
L_e	Length of expected response

Response:

Data Field	'7C' – L1 {'82' L2 {shared secret Z}} where • Z is the X coordinate of point P as defined in SP 800-56A, Section 5.7.1.2 • L2 is 32 bytes
SW1-SW2	'90 00' (Status word)

A.4.2.2 PIV KMK Specific ECDH Key Agreement Schemes

SP 800-56A describes five different ECDH key agreement schemes that a client application cryptographic module may implement. These schemes differ in 1) the number of keys (1 or 2) and 2) the type of keys (ephemeral or static) used by each party. Since the PIV Card only computes the ECC CDH Primitive using its static private key, the client application cryptographic module only employs the PIV Card in implementing an ECDH key agreement scheme when the scheme involves the use of the cardholder's static key pair. The ECDH key agreement schemes that involve the use of at least one party's static key pair, and thus may involve the use of the PIV Card are:

+ C(2, 2) – Each party has a static key pair and generates an ephemeral key pair (see Section 6.1.1 of SP 800-56A)

  In this scheme, the information sent between the client application and the PIV Card is the same when acting as the initiator or the responder; the other party's static public key is sent to the PIV Card; and a static shared secret is returned by the PIV Card in plaintext. Note that an ephemeral key pair is generated by the client application, and the private key of that key pair is combined with the other party's ephemeral public key to produce an ephemeral shared secret.

+ C(1, 2) – The initiator has a static key pair and generates an ephemeral key pair, while the responder has a static key pair (see Section 6.2.1 of SP 800-56A)

  When the cardholder is acting as the initiator, the other party's static public key is sent to the PIV Card; and a static shared secret is returned in plaintext by the PIV Card. Note that in this case, an ephemeral key pair is generated by the client application cryptographic module, and the corresponding ephemeral private key is combined with the other party's static public key to produce a second shared secret.

  When the cardholder is acting as the responder, two public keys are sent by the client application to the PIV Card (the other party's static and ephemeral public keys), and two shared secrets are returned in plaintext (the static shared secret and the ephemeral shared secret). Note that two GENERAL AUTHENTICATE commands are required to provide the two shared secrets to the client application's cryptographic module.

+ C(1,1) – The initiator generates only an ephemeral key pair, while the responder has only a static key pair (see Section 6.2.2 of SP 800-56A).

In this scheme, the PIV Card is only employed by the client application if the cardholder is acting as the responder. In this case, the other party's ephemeral public key is sent to the PIV Card, and the shared secret is returned by the PIV Card in plaintext.

+ C(0,2) – Both the initiator and responder use only static key pairs (see Section 6.3 of SP 800-56A)

  In the C(0,2) scheme, the information sent between the client application's cryptographic module and the PIV Card is the same when acting as the initiator or the responder; the other party's static public key is sent to the PIV Card, and the static shared secret is returned in plaintext. Note that for this scheme, the client application cryptographic module also generates a nonce when acting as the initiator of the scheme.

The C(2,0) scheme does not involve the use of static keys and so the PIV Card would not be involved in the implementation of this scheme.

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., 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 of 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
APT	Application Property Template
ASCII	American Standard Code for Information Interchange
ASN.1	Abstract Syntax Notation One
BER	Basic Encoding Rules
CLA	Class (first) byte of a card command
DER	Distinguished Encoding Rules
DES	Data Encryption Standard
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
ECDH	Elliptic Curve Diffie-Hellman
EC CDH	Elliptic Curve Cryptography Cofactor Diffie-Hellman
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 Organization for Standardization

ITL	Information Technology Laboratory
KDF	Key Derivation Function
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
PKCS	Public-Key Cryptography Standards
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
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, …, 9, 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.

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] NIST Special Publication 800-78-2, Cryptographic *Algorithms and Key Sizes for Personal Identity Verification*, February 2010. (See http://csrc.nist.gov)

[7] NIST Special Publication 800-56A, *Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography (Revised)*, March 2007. (See http://csrc.nist.gov)

[8] Standards for Efficient Cryptography Group (SECG), “SEC 1: Elliptic Curve Cryptography”, Version 1.0, September 2000.

[9] Jakob Jonsson and Burt Kaliski, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1", RFC 3447, February 2003. (See http://tools.ietf.org/html/rfc3447)