DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS E INFORMÁTICA

TEMA: PROPUESTA METODOLÓGICA PARA LA AUTENTICACIÓN DE LOS DATOS ALMACENADOS EN UN EPASSPORT, BASADA EN INFRAESTRUCTURA DE CLAVE PÚBLICA SEGÚN LAS RECOMENDACIONES DE LA ORGANIZACIÓN DE AVIACIÓN CIVIL INTERNACIONAL Y APLICADA AL CASO ECUATORIANO

AUTOR: PACHECO JÁCOME, JOSÉ LUIS

DIRECTOR: ING. GALÁRRAGA HURTADO, JUAN FERNANDO

SANGOLQUÍ

2019

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

CERTIFICACIÓN

Certifico que el trabajo de titulación, "***PROPUESTA METODOLÓGICA PARA LA AUTENTICACIÓN DE LOS DATOS ALMACENADOS EN UN EPASSPORT, BASADA EN INFRAESTRUCTURA DE CLAVE PÚBLICA SEGÚN LAS RECOMENDACIONES DE LA ORGANIZACIÓN DE AVIACIÓN CIVIL INTERNACIONAL Y APLICADA AL CASO ECUATORIANO***" fue realizado por el señor ***Pacheco Jácome, José Luis*** el mismo que ha sido revisado en su totalidad, analizado por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Sangolquí, 11 de enero de 2019

Firma:

.....

Ing. Juan Fernando Galárraga Hurtado

C. C. 1711464816

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

AUTORÍA DE RESPONSABILIDAD

Yo, ***Pacheco Jácome, José Luis***, declaro que el contenido, ideas y criterios del trabajo de titulación: ***Propuesta metodológica para la autenticación de los datos almacenados en un ePassport, basada en infraestructura de clave pública según las recomendaciones de la Organización de Aviación Civil Internacional y aplicada al caso Ecuatoriano*** es de mi autoría y responsabilidad, cumpliendo con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Consecuentemente el contenido de la investigación mencionada es veraz.

Sangolquí, 1 de febrero de 2019

Firma

.....

José Luis Pacheco Jácome

C.C.: 171603145-3

ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS

INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

AUTORIZACIÓN

Yo, ***Pacheco Jácome, José Luis*** *autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación:* ***Propuesta metodológica para la autenticación de los datos almacenados en un ePassport, basada en infraestructura de clave pública según las recomendaciones de la Organización de Aviación Civil Internacional y aplicada al caso Ecuatoriano*** *en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.*

Sangolquí, 1 de febrero de 2019

Firma

.....

José Luis Pacheco Jácome

C.C.: 171603145-3

DEDICATORIA

Dedico este trabajo a mis padres Guillermo y Marcia, que me han apoyado incondicionalmente a lo largo de toda la vida, así como a mis hermanos, a mi esposa Karina y mis hijos Nicolás y Sofía quienes constituyen mi motivación en los momentos más difíciles y mi inspiración en toda tarea emprendida.

AGRADECIMIENTO

Agradezco desde lo más profundo de mi corazón a todas aquellas personas que de una u otra manera me han bridado su apoyo, su cariño, su amistad, y su compañía en a lo largo todos estos años y cuyos nombres son demasiados para incluirlos en su totalidad. Estas personas son fundamentalmente mi familia, mis amigos, compañeros, y profesores, para quienes la palabra ‘gracias’ no alcanzar para expresar mis sentimientos de gratitud.

ÍNDICE DE CONTENIDOS

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

RESUMEN

Este trabajo busca ayudar en la comprensión de los conceptos que están detrás del pasaporte electrónico, regulado por la Organización de Aviación Civil Internacional y su Doc 9303. Con este fin se describen sus características lógicas y físicas, y se hace una revisión de los estándares ISO/IEC 14443 (que define su comunicación por radio frecuencia) e ISO/IEC 7816-4 (que define los comandos Smart Cards para intercambio) así como de la especificación PC/SC que facilita la interoperación de Smart Cards y dispositivos lectores correspondientes a diferentes fabricantes en el entorno PC. También se explican los principales mecanismos de seguridad aplicables a los ePassports y que sirven para proteger su Autenticidad (por medio de la Autenticación Pasiva que hace uso de una Infraestructura de Clave Pública), Confidencialidad (por medio del Control de Acceso Básico) y Originalidad (por medio de la Autenticación Activa). Esto sirve de preámbulo para plantear una metodología que permita hacer una validación electrónica del pasaporte, basada en tres resultados posibles. Por último se ha desarrollado, en el lenguaje de programación Java, una herramienta como instrumento para la aplicación de la metodología, que ha servido para realizar la validación de un conjunto de pasaportes, cada uno con características de entrada pre-establecidas, y que cubren las principales alternativas que pueden presentarse. El resultado de las pruebas ha demostrado que la metodología propuesta es útil para validar pasaportes electrónicos.

PALABRAS CLAVE:

- **PASAPORTE ELECTRÓNICO**
- **RFID**
- **SISTEMA DE INSPECCIÓN**
- **PKI**

ABSTRACT

This work tries to help in the comprehension of the concepts behind the electronic passport, regulated by International Civil Aviation Organization and his Doc 9303. With this aim, its logical and physical characteristics are described, and a revision of the ISO/IEC 14443 (which defines its communication through RF) and ISO/IEC 7816-4 (which defines the commands used in Smart Cards for exchange) standards is made, as well as the PC/SC specification that facilitates Smart Card and reader's interoperation corresponding to different manufactures, in the PC environment. It also explains the main security mechanisms applicable to ePassports and that serve to protect their Authenticity (through Passive Authentication that makes use of a Public Key Infrastructure), Confidentiality (through Basic Access Control) and Originality (through Active Authentication). All this serves as a preamble to propose a methodology that allows to perform an electronic validation of the passport, based on three possible results. Finally, in the Java programing language, a tool has been developed as an instrument for the methodology's application, which has served to validate a set of passports, each with pre-established input characteristics, and which cover the main alternatives that can be presented. The result of the test has shown that the proposed methodology is useful for validating electronic passports.

KEY WORDS

- **EPASSPORT**
- **RFID**
- **INSPECTION SYSTEM**
- **PKI**

1 INTRODUCCIÓN

1.1. Antecedences

En el ámbito laboral he tenido la oportunidad de involucrarme en los procesos de fabricación de pasaportes electrónicos ecuatorianos, y en virtud de que el estado ecuatoriano tiene previsto iniciar próximamente con la emisión de dichos pasaportes, se vuelve oportuno el proponer una metodología para la autenticación del contenido de un pasaporte electrónico, y a su vez desarrollar una herramienta de software que permita leer y verificar la autenticidad de dicho contenido, datos que fueron almacenados en el documento durante el proceso de emisión, y están protegidos con una firma digital respaldada por una infraestructura de clave pública.

1.2. Planteamiento del Problema

El hecho de incorporar un circuito integrado sin contacto (CI) al pasaporte, brinda ciertas ventajas para el estado emisor como:

- Mejora las seguridades del documento por efecto de la redundancia, ya que incluye en el CI algunos de los datos impresos en el documento (foto del titular y Zona de Lectura Mecánica). A su vez estos datos han sido firmados digitalmente, con lo que la seguridad inicial se fortalece con la aportada por la criptografía de clave asimétrica.
- Facilita el traslado internacional de los ciudadanos, ya que por ejemplo uno de los requisitos para que los ciudadanos de un país ingresen a la Unión Europea sin necesidad de visa es que dicho país emita pasaportes electrónicos.
- Rapidez en el despacho de viajeros en los aeropuertos, ya que posibilita la automatización de dicho procesos.

- Ayuda a la verificación de la identidad del portado del documento, ya que incluye datos biométricos obligatorios como el rostro y opcionales como huellas digitales e iris, datos que pueden servir como entrada a sistemas de reconocimiento biométricos (ICAO Doc 9303-9, 2015).

Estas ventajas justifican la inversión que el estado emisor tiene que hacer antes de emitir pasaportes electrónicos, inversión que principalmente debe destinarse a tres campos:

1. La fabricación del nuevo documento, ya que el costo de fabricar un pasaporte electrónico es mayor que el de un pasaporte mecánico.
2. La adquisición de un sistema de emisión de pasaportes, que incluye la puesta en marcha de una infraestructura de clave pública (PKI por sus siglas en inglés)
3. La adquisición de un sistema de inspección de pasaportes.

En cuanto al sistema de inspección, esto implica por un lado la compra del hardware necesario para la lectura mecánica y electrónica del documento y por otro lado del software encargado de validar la autenticidad de los datos impresos en el documento y el contenido de su CI sin contacto, para lo cual es necesario realizar una serie de pasos en el orden correcto, o dicho de otra manera regirse a una metodología. Por lo expuesto es necesario que cualquier oficina pública, que lo requiera, pueda contar la metodología requerida y una herramienta de bajo costo para aplicarla (basta con adquirir un lector RFID de Smart cards), con la cual poder acceder al contenido de un ePassport y leer los datos biométricos del portador del documento, con el fin de validar la autenticidad de estos datos por un lado o, si existe la necesidad, integrarlo a otro sistema o repositorio de datos para verificar la identidad del portador del documento como el caso de los sistemas de identificación biométricos. La herramienta, instrumento de aplicación de la metodología, a su vez pueden servir para validar la correcta emisión de ePassports, probando no

solo compatibilidad con normas internacionales, sino también el uso de diferentes algoritmos criptográficos para firma y hash, tarea que se realiza durante el proceso de emisión y es producto del inter-funcionamiento entre un sistema de emisión de pasaportes y una PKI.

1.3. Objetivos

1.3.1. Objetivo General.

Proponer una metodología que permita realizar la verificación de un Pasaporte electrónico, validando la autenticidad de los datos contenidos en el CI sin contacto.

1.3.2 Objetivos Específicos.

i. Analizar los estándares y especificaciones asociados a la tecnología RFID y su interacción con la PC.

ii. Investigar los algoritmos criptográficos de firma: RSA, DSA y DSA de curva elíptica (ECDSA), así como los distintos algoritmos de condensación, empleados por una PKI adecuada para emisión de ePassports.

iii. Implementar una herramienta para validar que la emisión de ePassports se realice en cumplimiento a la normativa internacional establecida.

1.4. Justificación

La justificación del proyecto radica principalmente en la utilidad (y beneficios derivados) que pueda ofrecer su producto final, y esta utilidad es básicamente la de guiar (gracias al procedimiento metodológico propuesto) el acceso y la verificación automática de la autenticidad de un pasaporte electrónico, proceso que descrito en forma breve consiste en: acceder al CI sin contacto, leer los datos biométricos y demográficos del portador del documento, verificar la

integridad y autenticidad de esos datos (firmados digitalmente en base a una PKI) y al mismo tiempo identificar al portador del documento.

Quienes se verían principalmente beneficiados con su uso serían los controles migratorios fronterizos y aeroportuarios, dichos beneficios estarían enmarcados en los entornos de seguridad y eficiencia:

- Seguridad porque tendrían la capacidad de realizar, de manera adecuada, la verificación del contenido del CI sin contacto, lo que sumado a los controles manuales de las características físicas del libretín brinda mayor certeza en la validación de la autenticidad del documento.
- Eficiencia porque se brindan facilidades para que el proceso de identificación del portador del documento sea automático, lo que implica que el despacho de viajeros sea más rápido.

Otros usos potenciales derivados serían:

- La herramienta, producto de la metodología que se planteará, le puede ser útil a la oficina emisora de documentos de viaje, para verificar que un pasaporte electrónico puede ser accedido y verificado en una forma estandarizada lo que significa que fue fabricado y su CI sin contacto fue posteriormente personalizado (durante el proceso de emisión) cumpliendo con la especificación ICAO Doc 9303. Esta tarea de verificación es requerida en un proceso de emisión de pasaportes electrónicos previa a la entrega al ciudadano. Adicionalmente la herramienta también se podría usar para probar diferentes algoritmos criptográficos (tanto de firma como de condensación) que se pueden usar, por la PKI, durante la personalización del documento.

- La herramienta, podría también ser usada por cualquier entidad estatal que requiera acceder a los datos que identifican al titular del documento, como por ejemplo la policía o la justicia, quienes además necesitarán integrarlo a sus sistemas informáticos, gracias a lo cual, con un proceso automático podrían, no solo identificar al portador del documento, sino también comprobar si este consta en alguna lista negra (como lista de difusión roja de Interpol o lista de prohibición de salida del país).

Los argumentos mencionados sustentan los usos que la metodología y su instrumento de aplicación (la herramienta) resultantes pueden tener, por lo que el llevar a cabo un proyecto para su desarrollo sería una labor plenamente justificada.

1.5. Alcance

El presente proyecto se enfocará en proponer una metodología para la autenticación del contenido de una pasaporte biométrico y en forma conjunta se va a desarrollar una herramienta de software que permita validar la metodología planteada, es decir que dicha herramienta permitirá acceder al contenido de un pasaporte biométrico y validar su autenticidad. El desarrollo de la propuesta metodológica y/o de la herramienta de software estará delimitado por las siguientes consideraciones:

1.5.1. Mecanismos de seguridad

Los mecanismos o métodos de seguridad que incluirán en el proceso son los considerados como OBLIGATORIOS de ejecutar para un sistema de inspección en el Doc 9303 de ICAO (Autenticación Pasiva y Control de Acceso Básico), más un procedimiento que permita verificar la autenticidad del chip y que este no se trata de una copia (ej. Autenticación Activa). Estos mecanismos son:

- Autenticación Pasiva, que sirve para comprobar la integridad y autenticidad de los datos contenidos en el CI sin contacto y requiere la verificación de la firma digital de los datos contenidos en el CI sin contacto, así como también de la cadena de confianza asociada a dicha firma. Los certificados requeridos para realizar esta verificación deberán estar accesibles localmente en el equipo en el que corra la aplicación, es decir que se trata de un proceso offline.
- Control de acceso base (BAC por sus siglas en inglés) que controla el acceso a los datos menos sensibles contenidos en el CI sin contacto, e impide la escucha no autorizada de la comunicación entre el pasaporte electrónico y el equipo lector.
- Autenticación Activa prueba que los datos se están leyendo del chip RF original y no de una copia o clon de este.
- En virtud de que el Doc 9303 en su parte 12 menciona que los estados expedidores pueden usar para firma alguno de los algoritmos: RSA, DSA o ECDSA; y para condensación o hash alguno de los algoritmos SHA-224, SHA-256, SHA-384 o SHA-512, la herramienta a desarrollar deberán soportar todos estos algoritmos criptográficos.

1.5.2. Interacción de la herramienta con otros sistemas

Durante la definición de la metodología no se contempla la interacción con otros sistemas y la herramienta que se plantea desarrollar tampoco interactuará con otras aplicaciones o estará conectada a alguna base de datos de registro de ciudadanos, esto dado que la aplicación pretende validar la autenticidad electrónica del documento, mas no validar la identidad del ciudadano portador del documento. Por este motivo resulta innecesario el manejo de usuarios o roles, o registrar los resultados de la lectura de un pasaporte de manera persistente.

1.5.3. Compatibilidad de la herramienta con dispositivos

La herramienta a desarrollar tendrá la capacidad de interactuar con cualquier dispositivo lector de tarjetas inteligentes que se sujete a estándares, lo que significa que debe cumplir con las normas ISO/IEC 14443 Tipo A o Tipo B e ISO/IEC 7816-4 y con la especificación PC/SC. En lo que concierne a dispositivos lectores de pasaportes con capacidad de lectura óptica, la interacción con tales dispositivos no se basa en estándares sino que está atada al SDK o API proporcionado por cada fabricante, motivo por el cual para el presente proyecto no se trabajará con equipos lectores ópticos de pasaportes. En virtud de esto los datos requeridos para el acceso al chip RF serán ingresados manualmente.

2 ESTADO DEL ARTE

La Organización de Aviación Civil Internacional (ICAO), organismo especializado de la ONU, es encargada de redactar especificaciones normativas para pasaportes y otros documentos de viaje a través del documento 9303 cuya primera versión data de 1980. En ,1998 la ICAO inició sus esfuerzos por incorporar la tecnología a los MRTD con el fin de mejorar la identificación biométrica, pero los acontecimientos acaecidos del 11 de septiembre de 2001 aceleraron la conclusión de dicha tarea, y los resultados se publicaron en el 2006 (Doc 9303 parte 1, volumen 2) y 2008 (Doc 9303 parte 3, volumen 2), en tales documentos se especifican ya los temas referentes al uso de la biometría y la tecnología contactless, la estructura lógica de datos - LDS y la infraestructura de clave pública - PKI (ICAO, 2015).

Es importante señalar que detalles concernientes a la implementación específica del chip incluido en el Pasaporte Electrónico Ecuatoriano como por ejemplo: fabricante, modelo, sistema operativo y versión del mismo, etc., no pueden ser revelados debido a cuestiones de seguridad, sin embargo para el desarrollo del presente trabajo basta con conocer que se trata de un chip conforme con la especificación de ICAO.

2.1. El ePassport

Un ePassport o e-passport, también conocido como pasaporte electrónico o biométrico, es un pasaporte que tiene embebido un circuito integrado (IC) sin contacto y una antena, y para diferenciarlo de un pasaporte mecánico tradicional lleva impreso el símbolo de micro plaqueta contenida en el anverso del libretín.

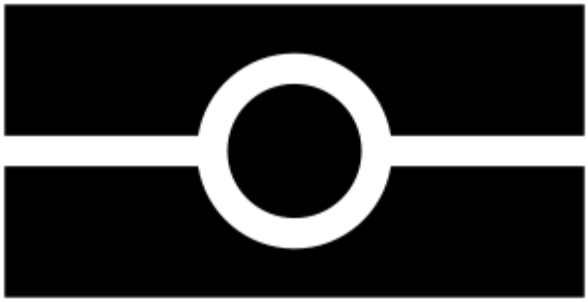

Figura 1: Símbolo de chip contenido, que identifica a un pasaporte electrónico
Fuente: *(ICAO, 2015)*

Los Países no están obligados a emitir pasaportes electrónicos, sin embargo los beneficios derivados de hacerlo son:

- El chip sin contacto brinda al pasaporte una capa adicional de seguridad pues contiene una copia redundante de los datos del portador impresos en la página de datos del documento. Estos datos a su vez, están protegidos por una PKI (diseñada exclusivamente para la emisión de documentos de viaje) a través de una firma digital, con la finalidad de garantizar su integridad y autenticidad. Por otro lado la lectura de dichos datos está restringida por un método de control de acceso, lo que blinda al documento de ataques como skimming y eavesdropped. Además de poder autenticar al documento (verificando su firma digital), gracias al empleo de la biometría, los Estados receptores también están en la capacidad de verificar la identidad de su titular.
- La validación de un ePassport puede automatizarse, lo que por un lado es más confiable y por otro agilita el despacho de pasajeros en los controles fronterizos (ICAO, 2015).
- Un ePassport además tiene la posibilidad de evolucionar para adaptarse a exigencias futuras, con mejoras que podrían ser incorporadas al chip, como por ejemplo: protocolo de autenticación, método de control de acceso o algoritmo de firma digital más seguros.

El Doc 9303 especifica tres características biométricas (una obligatoria y dos opcionales) a usar en los ePassports, las mismas que deben ser almacenadas en forma de imágenes digitales en el CI sin contacto (ICAO, 2015):

a) Imagen facial, de uso obligatorio y conforme a ISO/IEC 19794-5, para su aplicación en sistemas de reconocimiento de rostro.
b) Biometría de la huella digital, de uso opcional y conforme a ISO/IEC 19794-4, para su aplicación en sistemas de reconocimiento de huella digital.
c) Biometría del iris, de uso opcional y conforme a ISO/IEC 19794-6, para su aplicación en sistemas de reconocimiento de iris.

Las características del CI sin contacto que incorpora el ePassport, se describen en la siguiente sección.

2.2. Características físicas e interfaz de comunicación

El ePassport hace uso de la tecnología RFID para la comunicación, así no requiere entrar en contacto con el dispositivo lector facilitando su lectura. El CI sin contacto y su antena pueden estar colocado de varias maneras en un ePassport: en la cubierta anterior, en la cubierta posterior, en la página de datos o en otra de las páginas del libretín. El tamaño de la antena se ajusta a ISO/IEC 14443-1 (características físicas) Clase 1 (tamaño de antena ID-1) 85,60 x 53,98 mm, operando a una frecuencia de 13.56 MHz (ICAO, 2015). Pueden existir 2 tipos de interfaz de señal: tipo A y tipo B, cuyas principales diferencias están en el método de modulación, codificación de los bits y los procedimientos del protocolo de inicialización. Ambos tipos de interfaz utilizan el mismo protocolo de transmisión. Estas características permiten al dispositivo lector comunicarse con el CI del pasaporte a una distancia de hasta 10 cm. En (ICAO, 2015) se

recomienda una velocidad no menor a 424 kb/s. La manera en que funciona el CI se especifica en la norma ISO/IEC 14443 (partes 2, 3 y 4) de la que se hará una revisión breve en el capítulo 3.

2.3. Características Lógicas del CI sin contacto

La estructura de ficheros del CI sin contacto debe ser la definida en ISO/IEC 7816-4, en esta misma norma se especifican las ordenes o comandos para interactuar con las aplicaciones contenidas en el CI si contacto.

- Capacidad: La capacidad de almacenamiento del CI sin contacto debe ser de al menos 32kBs, requeridos para almacenar los datos obligatorios (ICAO, 2015).
- Aplicaciones: El chip obligatoriamente debe contener la aplicación conocida como eMRTD, cuyo identificador de aplicación AID según se especifica en (ISO/IEC 7816, 2013) debe ser: 0xA0000002471001. El chip puede albergar otras aplicaciones a discreción del estado expedidor (ICAO, 2015).

2.3.1. Estructura Lógica de Datos (LDS)

ICAO ha definido una Estructura Lógica de Datos única para inter-funcionamiento mundial, La LDS se basa en una estructura jerárquica de ficheros transparentes de longitud variable y acceso aleatorio, en la que existen ficheros especializados (DF) y ficheros elementales (EF) (ISO/IEC 7816, 2013). Opcionalmente un fichero especializado conocido como fichero maestro (MF) puede estar en la raíz de la jerarquía. Un DF puede albergar aplicaciones o contener EF's. Un DF que alberga una aplicación se llama *application DF*. En la **Figura 2** se muestra un esquema de la estructura de ficheros.

2.3.2. Organización de los datos (agrupamiento de los elementos de datos)

Dentro de la LDS los datos están organizados en agrupamientos lógicos de datos conexos llamados grupos de datos (DG), cada uno de los cuales tiene asignado un número de referencia, DG1 a DG16 respectivamente. Cada DG está almacenado en un EF Transparente. Los grupos de datos 1 a 16 están protegidos contra escritura, y solo el Estado expedidor debe tener acceso de escritura sobre ellos. El acceso a cada EF se hace directamente por medio de un identificador breve asignado (ICAO, 2015).

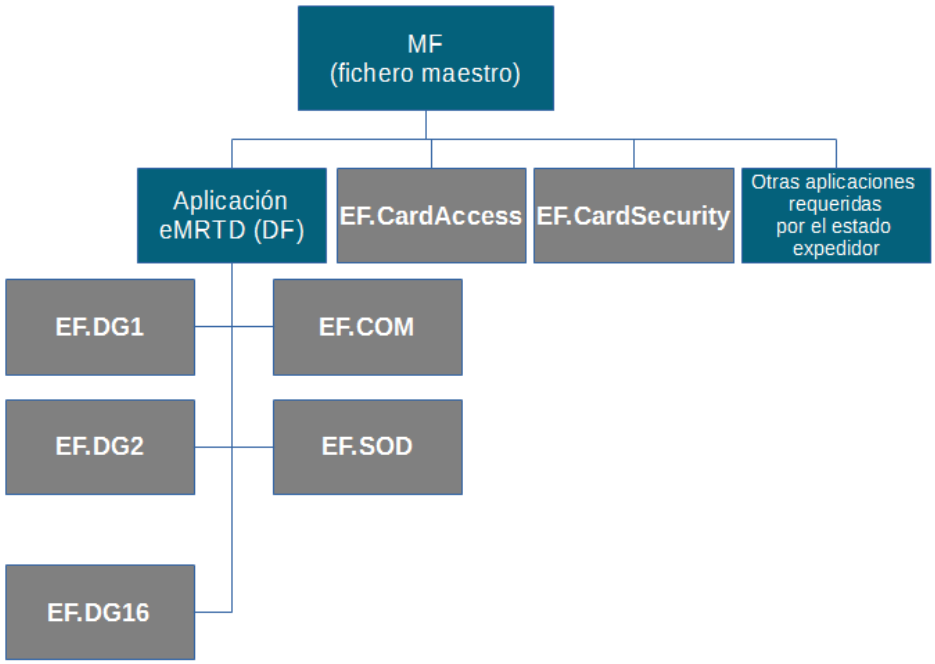

Figura 2: Estructura Lógica de Datos
Fuente: (ICAO, 2015)

2.3.3. Elementos de datos obligatorios y opcionales

El Doc 9303 especifica grupos de datos que se deben incluir obligatoriamente en la LDS y grupos de datos que se pueden incluir opcionalmente. Los datos cuya presencia se exige en el

ePassport son una copia del MRZ en el grupo de datos 1, el rostro del titular en el grupo de datos 2 y el Objeto de Seguridad del Documento EF.SOD (ICAO, 2015). La ***Tabla 1*** muestra los datos OBLIGATORIOS, OPCIONALES y CONDICIONALES, el contenido asignado a cada EF y su formato.

Tabla 1

Datos obligatorios y opcionales contenidos en la LDS

Fichero Elemental	Datos que contiene	Obligatoriedad
EF.COM	Versión de LDS (1.7 o 1.8), versión de Unicode y lista de DGs presentes.	Obligatorio
EF.DG1	Contenido de la Zona de Lectura Mecánica: tipo de documento, estado expedidor, nombre del titular, número de documento + dígito verificador, nacionalidad, fecha de nacimiento + dígito verificador, sexo, fecha de caducidad + dígito verificador, datos opcionales + dígito verificador, dígito verificador compuesto.	Obligatorio
EF.DG2	Rostro codificado: imagen facial del titular codificada con el estándar ISO/IEC 19794-5 que a su vez es compatible con la estructura de datos CBEFF.	Obligatorio
EF.DG3	Dedo(s) codificado(s): una o varias huellas digitales o palmas codificadas en el estándar ISO/IEC 19794-4 que es compatible con la estructura CBEFF.	Opcional
EF.DG4	Ojos(s) codificado(s): uno o más iris correspondiente con el formato CBEFF.	Opcional
EF.DG5	Retrato exhibido: uno o más retratos exhibidos formateados según ISO/IEC 10918-1 (formato JFIF) o ISO/IEC 15444 (JPEG 2000).	Opcional
EF.DG6	Reservado, uso futuro	Opcional
EF.DG7	Firma o marca habitual exhibida: una o más firmas exhibidas formateadas según ISO/IEC 10918-1 (formato JFIF) o ISO/IEC 15444 (JPEG 2000).	Opcional
EF.DG8	Elementos datos (no definido aún): uno o más elementos de datos con formato definido por el estado expedidor.	Opcional
EF.DG9	Elemento estructura (no definido aún)	Opcional
EF.DG10	Elemento substancia (no definido aún): uno o más elementos de datos con formato definido por el estado expedidor.	Opcional

CONTINÚA

EF.DG11	**Detalles personales adicionales del titular (cada dato puede o no estar presente): nombre completo del titular (en caracteres nacionales), otros nombres, número personal, fecha de nacimiento completa, lugar de nacimiento, dirección permanente, teléfono, profesión, título, resumen personal, imagen del documento de ciudadanía formateada según ISO/IEC 10918-1 (formato JFIF) o ISO/IEC 15444 (JPEG 2000), otros números de documentos de viaje válidos, información de custodia.**	**Opcional**
EF.DG12	Detalles del documento adicionales (cada dato puede o no estar presente): autoridad expedidora, fecha de expedición, nombres de otras personas incluidas en el MRTD, aprobaciones/observaciones, requisitos impositivos de salida, imagen anterior del documento y/o imagen posterior del documento formateadas según ISO/IEC 10918-1 (formato JFIF) o ISO/IEC 15444 (JPEG 2000), fecha y hora de personalización del documento, número de serie del sistema de personalización	Opcional
EF.DG13	Detalles opcionales a discreción del estado expedidor	Opcional
EF.DG14	Opciones de seguridad: SecurityInfos en estructura ASN.1 que permiten implantar opciones de seguridad para características biométricas adicionales (autenticación de CI).	Obligatorio si el eMRTD apoya la correspondencia de autenticación de ci o pace
EF.DG15	Información de clave pública de autenticación activa	Condicional si se implanta la autenticación de ci de autenticación activa
EF.DG16	Personas que han de notificarse: información de contacto con personas para notificación de emergencia, para cada elemento se registra nombre de la persona, teléfono, dirección y fecha de registro.	Opcional
EF.SOD	Objeto de seguridad de documento, contiene los hash de todos los DGs presentes y está firmado digitalmente por el Estado emisor, debe codificarse como SignedData, según RFC 3369, con el formato de codificación distinguida (DER).	Obligatorio
EF.CardAccess	Parámetros usados para PACE: PACEInfo y PACEDomainParameterInfo, formateado como SecurityInfos codificado en DER.	Condicional si el MRTD apoya PACE
EF.CardSecurity	Contiene información de clave pública de autenticación de Chip requerida para PACE-CAM, así como también las SecurityInfos contenidas en EF.CardAccess, formateado como SignedData codificado en DER.	Condicional si el MRTD apoya pace con correspondencia de autenticación de ci

Fuente: (ICAO, 2015)

2.3.4. Formato de los datos

Cada DG contiene objetos de datos dentro de una plantilla (como se define en ISO/IEC 7816-4 y 7816-6), cada uno de las cuales se registra usando el esquema de codificación BER TLV (Basic-Encoding-Rules/Type-Lenght-Value) de ASN.1 (Abstract Syntax Notation One) especificado en ISO/IEC 8825-1, es decir que cada objeto de datos contiene un rótulo de identificación (en hexadecimal por ejemplo 0x5A), una longitud y un valor únicos (ICAO, 2015). Los rótulos que deben emplearse para cada objeto de datos están especificados en el propio Doc 9303. Para permitir que los estados receptores puedan saber cuáles son los datos que están presentes en aquellos grupos de datos que contienen datos subordinados, se incluye una llamada Correspondencia de presencia de datos (DEPM), que consiste en una lista de rótulos que identifica los datos específicos registrados en un grupo de datos (ICAO, 2015). Para los grupos de datos 2, 3 y 4 y los elementos opcionales de seguridad biométrica adicional definidos en el Doc 9303-12, se debe usar el Marco común de formatos de intercambio biométrico CBEFF (ICAO, 2015).

2.4. El uso actual del ePassport

2.4.1. Alrededor del mundo

Aun cuando los esfuerzos de ICAO para incorporar la tecnología a los pasaportes iniciaron en 1998, no fue hasta 2004 que un primer país (Bélgica) llego a emitir un pasaporte electrónico conforme con ICAO. Y a partir de entonces el uso del pasaporte electrónico se ha difundido en el mundo en los últimos años, y en la actualidad existen más de 100 estados y no-estados (por ejemplo Naciones Unidas) que emiten pasaporte electrónicos y más 490 millones de

pasaportes biométricos en circulación. En la ***Figura 3*** se muestran los estados que actualmente expiden un pasaporte biométrico.

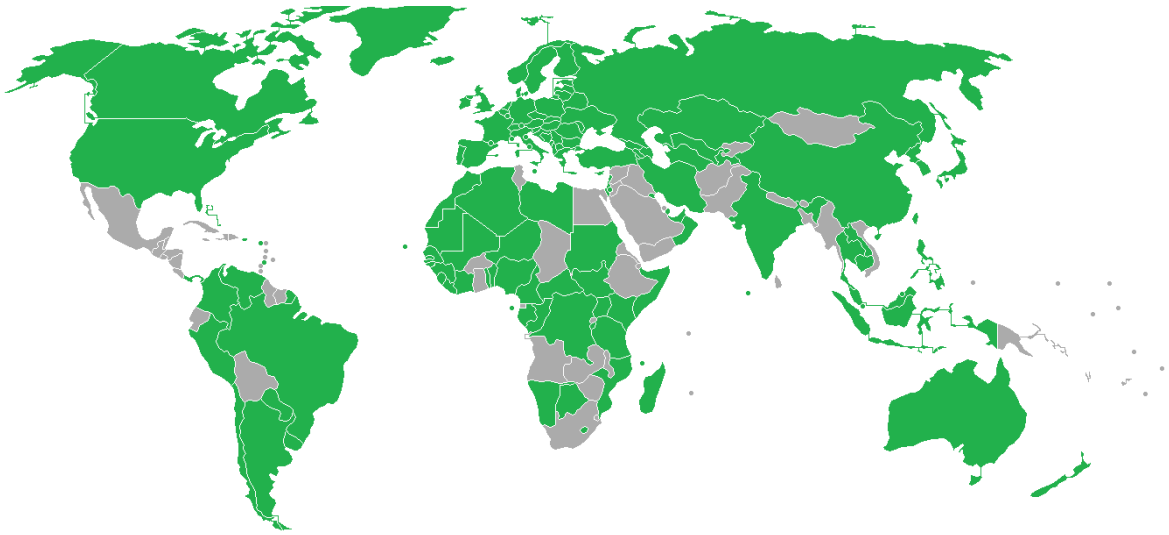

***Figura 3*:** Estados que emiten pasaportes electrónicos (en color verde)
Fuente: (Alinor, 2010)

2.4.2. Ámbito local y regional

Como se puede apreciar en la Ilustración 3, quedan pocos países en América del Sur que aún no emiten pasaporte electrónico, estos son Guyana, Surinam, Ecuador y Bolivia. En el caso ecuador, el gobierno estima que el primer pasaporte electrónico se emitirá en el primer semestre del 2019, esta decisión se ve motivada porque la emisión del pasaporte electrónico es un requisito, por parte de la Unión Europea, para la exención del visado Schengen para periodos de corta duración (Martinez, 2018). Algo similar ocurrió con nuestros países vecinos Colombia y Perú en donde la implementación el ePassport fue uno de los compromisos que ahora permite a sus ciudadanos viajar a Europa sin necesidad de visa.

2.5. Protección y Autenticidad de los datos almacenados electrónicamente

Las partes 11 (Mecanismos de seguridad para los MRTD) y 12 (Infraestructura de clave pública para los MRTD) del Doc 9303 se enfocan en los aspectos de seguridad del pasaporte electrónico. Los mecanismos de seguridad del chip RF tienen, por un lado, el objetivo de proteger los datos del portador contra acceso no autorizado, y por otro lado, garantizar que el documento fue emitido por una institución gubernamental y los datos en él contenidos no son falsificados.

2.5.1. Mecanismos de Seguridad para ePassports

El Doc 9303 proporciona mecanismos de seguridad electrónica destinados a proteger la autenticidad (integridad incluida), originalidad y confidencialidad de los datos almacenados en el chip RF integrado al pasaporte. De estos mecanismos, únicamente aquel destinado a proteger la autenticidad de los datos (Autenticación Pasiva) es obligatorio de implementar para los estados emisores (BSI, 2015). En la **Tabla 2** se resumen los diferentes mecanismos de seguridad, de los cuales se tratará con más detalle en el capítulo 4.

Tabla 2

Mecanismos de seguridad aplicables a un ePassport

Mecanismo	Protege	Técnica Criptográfica
Autenticación Pasiva (PA)	Autenticidad	Firma Digital
Autenticación Activa (AA)	Originalidad	Desafío-Respuesta
Autenticación de Chip (CA)	Originalidad y Confidencialidad	Autenticación y Canal Seguro
Control de Acceso al Chip • **Control de Acceso Básico (BAC)** • **Establecimiento de Conexión Autenticada por Contraseña (PACE)**	Confidencialidad	Autenticación y Canal Seguro

Fuente: (BSI, 2015)

2.5.1.1 Autenticidad de los datos

El mecanismo básico impuesto por la ICAO para verificar la autenticidad de un ePassport es la Autentificación Pasiva (PA) y a breves términos consiste en la verificación de la firma digital de los datos contenidos en el CI sin contacto (ICAO, 2015), para lo cual el Doc 9303 en su parte 12 define una Infraestructura de Clave Pública (PKI), sobre la cuál trataremos más adelante en este mismo capítulo.

2.5.1.2 Control de Acceso al Chip

Los controles de acceso al chip proporcionan privacidad al portador del documento ya que previenen el acceso no autorizado a los datos menos sensibles almacenados en el CI sin contacto (MRZ e imagen facial). El Control de Acceso Básico (BAC) está basado en criptografía simétrica lo que limita la fuerza de las claves de sesión derivadas a la fuerza de la entrada (clave semilla obtenida del MRZ), por otro lado el protocolo de Establecimiento de Conexión Autenticada por Contraseña (PACE) está basado en criptografía asimétrica y provee claves de sesión cuya fuerza es independiente de la entropía de la entrada (BSI, 2015).

ICAO también permite, a los estados emisores, la opción de incluir en el CI sin contacto datos más sensibles (como huellas digitales e imágenes de iris), pero debe restringirse el acceso a estos datos sólo a terminales autorizados, para lo cual se proporciona dos alternativas: restringirlos con un Control de Acceso Extendido (EAC) o cifrarlos; sin embargo ICAO no proporciona una implementación en este sentido, dejando al estado emisor decidir la implementación a usar, basados en especificaciones internas o en convenios bilaterales de estados que comparten esta información (ICAO, 2015). La implementación más extendida de EAC es la usada por los países miembros de la Unión Europea y especificada en el reporte técnico TR-

03110 de la Oficina Federal Alemana para la Seguridad de la Información BSI (Surós, 2013). EAC a su vez se compone de otros protocolos como son:

- Autenticación del Chip (CA): Establece un canal de comunicación seguro y permite la detección de chips RF clonados.
- Autenticación del Terminal (TA): consiste en la autenticación del dispositivo lector para que este pueda acceder a los datos más sensibles almacenados en el chip RF.

2.5.1.3 Prevención de la sustitución del chip

Para evitar la sustitución del chip, ICAO permite el uso de tres mecanismos para autenticar el chip (ICAO, 2015):

1. Autenticación Activa, que se explica en la sección 4.1, la información requerida para la ejecución de este mecanismo, si es apoyado por el chip, se encuentra en DG15.
2. Autenticación del chip, la información requerida para la ejecución de este mecanismo, si es apoyado por el chip, se encuentra en los correspondientes SecurityInfos dentro de DG14.
3. PACE con Mapping de autenticación de chip (PACE-CAM), la información requerida para la ejecución de este mecanismo, si es apoyado por el chip, se encuentra en una estructura PACEInfo dentro de CardAccess.

2.5.2. PKI para la emisión de ePassports

La ICAO a través del Doc 9303 define una PKI diseñada exclusivamente para la emisión de documentos, en la que existe una única Autoridad Certificadora (CA) por país, conocida como Autoridad Certificadora de Firma de País o CSCA, que es quién emite los certificados a las entidades finales encargadas de firmar los documentos conocidas como Firmante de Documentos

– DS (ICAO, 2015). En un país puede haber más de un firmante de documentos DS. Los certificados que puede emitir una CSCA son:

- Certificados de firmante de documento C_{DS}, necesarios para firmar la LDS de los pasaportes electrónicos.
- Certificados para entidades firmantes de lista maestra (opcional). Una lista maestra es una lista, firmada digitalmente, de certificados CSCA (nacionales y extranjeros) en los que el emisor de la lista confía.
- Certificado de firmante de lista de desviaciones (opcional).
- Certificados de enlace CSCA (opcional), que pueden ser usados para ayudar al estado receptor a establecer confianza en una nueva clave/certificado CSCA después de una renovación de claves.
- La CSCA además puede emitir Listas de Revocación de Certificados (CRL), que sirven para informar los certificados emitidos que han sido revocados.

Para que los estados receptores puedan validar los documentos emitidos por un estado emisor, el estado emisor emite un certificado auto-firmado que contiene su clave pública de CSCA, este certificado CSCA es distribuido a los estados receptores, quienes establecen su confianza en dicho certificado (verificando su autenticidad a través de medios fuera de banda como teléfono o email) y en base al el validan los certificados emitidos por el estado emisor, entre ellos, los certificados de firmante de documento C_{DS} que sirven para firmar la LDS de los ePassports. Como se puede apreciar la verificación de la autenticidad de los datos almacenados en el ePassport se basa en la confianza que recae sobre el certificado CSCA de país, por lo que es de vital importancia que el par de claves (pública-privada) de CSCA se generen y almacene en

una infraestructura altamente protegida y fuera de línea (ICAO, 2015). La ***Figura 4*** muestra los componentes de una PKI de país.

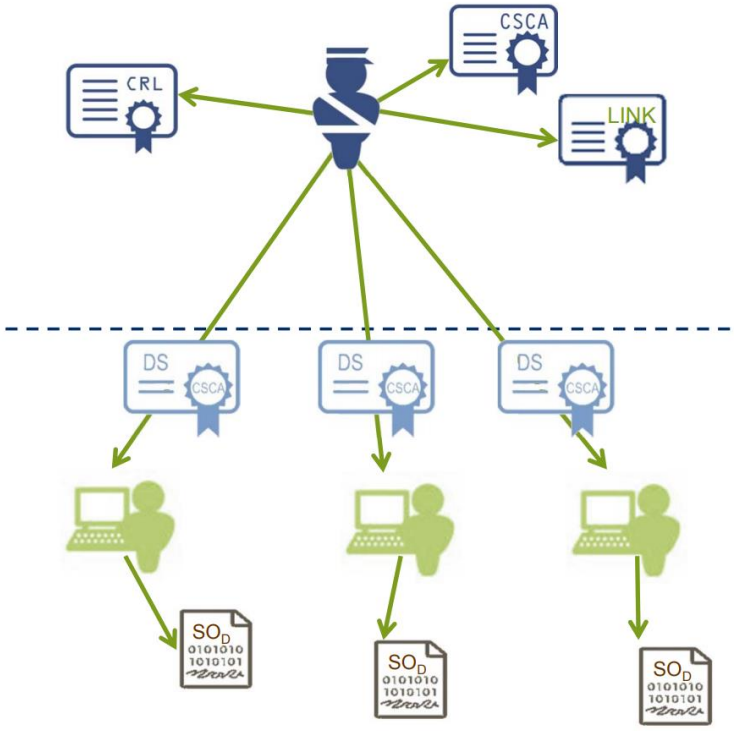

Figura 4: Componentes de una PKI de país, Firmante de País, y Firmante de Documentos
Fuente: (Joynes, 2012)

La distribución de los certificados entre estados se puede realizar de varias maneras (ICAO, 2015):

- Por intercambio bilateral los estados pueden intercambiar certificados CSCA, CRL's y listas maestras.
- Por medio de un directorio de clave pública (PKD) proporcionado por ICAO. Al ser un repositorio centralizado los estados tienen la facilidad de que pueden cargar su propia información y descargar la de otros estados en un único repositorio centralizado, evitando

la complejidad de mantener múltiples acuerdos de intercambio bilateral (un acuerdo con cada estado expedidor).

- Por medio de Listas Maestras, que al contener varios certificados CSCA en una sola lista, también ayudan a reducir la complejidad de mantener múltiples acuerdos de intercambio bilateral.

Además como se manifiesta en (ICAO, 2015) una PKI para firma de documentos de viaje electrónicos debe apoyar los siguientes algoritmos criptográficos de firma digital:

- RSA descrito en RFC 4055, se recomienda usar el mecanismo de firma RSASSA-PSS.
- Algoritmo de firma digital (DSA), especificado en FIPS 186-4.
- DSA de curva elíptica (ECDSA), descrito en X9.62 o en ISO/IEC 15946.

Y los siguientes algoritmos para condensación: SHA-224, SHA-256, SHA-384 y SHA-512 referidos a FIPS 180-2.

3 TECNOLOGÍA RFID EMPLEADA EN UN ePASSPORT

El término RFID (Radio Frequency Identification) hace referencia a un sistema de comunicación que hace uso de ondas de radio y abarca una amplia variedad de tecnologías, cada una de ellas trabajando en diferentes bandas de frecuencia (entre los 120 kHz y 10 GHz) y reguladas por diferentes estándares. Una de las tecnologías RFID es la conocida como Contactless Smart Cards (tarjetas inteligentes sin contacto) y es precisamente la empleada en el ePassport. Esta tecnología se basa fundamenta en 2 estándares: ISO/IEC 14443 (para las capas inferiores) e ISO/IEC 7816 (para las capas superiores). La Tabla 3 muestra una analogía con las capas del modelo de referencia OSI y el estándar que opera en cada una de ellas.

Tabla 3

Tecnología Contactless Smart Cards como un esquema de capas

Capas OSI	Contactless Smart Cards
Capas superiores	Comandos APDU para intercambio y sistema de archivos (ISO 7816-4)
Transporte	Protocolo de transmisión (ISO 14443-4)
Enlace	Inicialización y anticolisión (ISO 14443-3)
Física	Potencia RF e interface de señal (ISO 14443-2)

El CI sin contacto contenido en un ePassport usa una antena para comunicarse con el dispositivo lector por medio de la tecnología de inducción a velocidades de transmisión de datos de 106 a 848 kbits/s. Al acercar un ePassport al dispositivo lector, el campo magnético generado por este último induce una corriente eléctrica que permite al microprocesador del ePassport operar, de manera que puede iniciar el procedimiento de comunicación con el dispositivo lector. Para mejorar la comprensión de esta tecnología, en primer lugar se hará una revisión breve de la norma ISO/IEC 14443 que determina su modo de operación, en capas inferiores a las del ámbito

de aplicación. Posteriormente se va a pasar a revisar, con un poco más de detalle, la norma ISO/IEC 7816-4 que determina la manera en que se intercambian los datos en capas superiores, algo sumamente importante para el poder llevar a cabo el desarrollo de una herramienta de validación de ePassports.

3.1. Breve revisión de la norma ISO/IEC 14443 (contactless Smart Cards)

Esta norma está compuesta por 4 partes, de las cuales la parte 1 únicamente trata características físicas de una tarjeta contactless Smart Cards como por ejemplo su tamaño, por lo que no amerita su revisión en el presente trabajo.

En la parte dos del estándar se definen dos modos de operación llamados Tipo A y Tipo B, ambos operando a una frecuencia 13.56 MHz (ISO/IEC 14443, 2008). La **Tabla 4** muestra las principales diferencias de ambos modos de operación dependiendo del sentido de la comunicación, en donde PCD (proximity coupling device) se refiere al dispositivo lector y PICC (proximity integrated circuit card) se refiere al CI sin contacto.

En la parte tres del estándar se explica cómo puede el lector identificar los chips RF en el campo magnético y establecer comunicación con un chip específico. Esta parte de la norma específica el formato de byte, tamaño de frames y sincronización usados durante las dos fases iniciales de comunicación (inicialización y anticolisión). En la fase de inicialización el chip queda activado al estado READY por el uso de comandos intercambiados entre el lector y chip. Una colisión se da cuando múltiples CIs están en el rango de lectura del mismo lector y responden simultáneamente, cuando esto sucede el lector es incapaz de interpretar la señal recibida. Para resolver este problema, el estándar define el protocolo anticolisión que debe efectuarse para cada uno de los modos de operación: protocolo de detección Búsqueda binaria

dinámica (Dynamic Binary Search o DBS) para el Tipo A y ALOHA enmarcado ranurado dinámico (Dynamic Framed Slotted ALOHA o DFSA) para el Tipo B (ISO/IEC 14443, 2008). A continuación se explica brevemente ambos protocolos.

Tabla 4

Diferencias entre Tipo-A y Tipo-B, según el sentido de la comunicación

PCD a PICC		
Característica	**Type-A**	**Type-B**
Modulación	Modulación por amplitud 100% ASK (Amplitude Shift Keying)	Modulación por amplitud 10% ASK (Amplitude Shift Keying)
Codificación de bits	Codificación Miller modificada	NRZ
Alimentación de energía	Recibe ráfagas de energía del lector mientras están en comunicación	Recibe energía continuamente del lector mientras está en contacto con este.
Velocidad de datos	106 kbits/s	106 kbits/s hasta 847 kbits/s
PICC a PCD		
Característica	**Type-A**	**Type-B**
Modulación	Load OOK	Load BPSK
Codificación de bits	Codificación Manchester	NRZ-L
Subcarrier	f/16	f/16
Velocidad de datos	106 kbits/s	106 kbits/s

3.1.1. Protocolo DBS

Este protocolo funciona como un algoritmo de búsqueda de árbol binario. El lector empieza enviando un paquete Query que anuncia el inicio de las ranuras de tiempo. Si solo hay una tarjeta en el área esta será identificada exitosamente, caso contrario ocurre una colisión (cuando más de una tarjeta responde en la misma ranura de tiempo). Cuando ocurre la colisión las

tarjetas involucradas se dividen en dos grupos, 0 y 1. Las tarjetas que pertenecen al primer grupo (grupo 0) vuelven a responder al lector en la siguiente ranura de tiempo, si más de una tarjeta responde se produce una nueva colisión y por lo tanto se tiene que realizar una nueva división. Las divisiones continúan recursivamente hasta que el conjunto es reducido a una solo tarjeta. El proceso termina cuando todas las tarjetas han sido reconocidas por el lector. En este protocolo el peso computacional recae en el lector, y no es eficiente cuando el número de tarjetas a reconocer es grande (Álvarez, 2013).

3.1.2. Protocolo DFSA

El tiempo es dividido en unidades llamadas ranuras (o slots), estas ranuras están confinadas a una súper-estructura llamada frame (o ciclo). Cada frame tiene un número diferente de ranuras. El lector anuncia el inicio de una ranura por medio de un paquete Query, los CIs aleatoriamente seleccionan una ranura dentro del frame para enviar su información al lector. Cuando un frame finaliza, un ciclo de identificación concluye y el lector tiene que decidir si para el siguiente frame incrementa, decrementa o mantiene el número de ranuras de tiempo. Dado que el número de CIs en rango por ciclo es comúnmente desconocido, el lector debe estimar de antemano el número de CIs que van a competir en ese ciclo, esto puede hacerse a través de la información estadística recopilada en ciclos anteriores. Luego el lector ajusta el tamaño del frame para lograr el máximo rendimiento (Álvarez, 2013).

La parte cuatro de la norma define el protocolo de transmisión (half-duplex) usado en ambos modos de operación, en particular define la secuencia del protocolo de activación y desactivación. La secuencia de activación se concluye a través del proceso “Protocolo y Selección de Parámetros” sobre el cual se pueden intercambiar datos. Una vez finalizado el

intercambio de datos, el CI se puede desactivar a través del proceso de de-selección (ISO/IEC 14443, 2008). En esta parte de la norma se establece que la unidad de comunicación a usar es el APDU, usado en la tecnología Contact Smart Card (tarjetas inteligentes de contacto) y que está definido en ISO/IEC 7816-4: Organización, seguridad y comandos para intercambio, de lo cual se hablará más adelante en este capítulo.

ISO/IEC 14443 es un estándar base, solo proporciona especificaciones básicas y requerimientos pero no contiene escenarios de prueba. Para este propósito fue introducido el estándar ISO/IEC 10373-6, proporcionando test para ambos CIs (o tarjetas) y lectores, sobre todas las capas cubiertas por la norma.

3.2. El estándar ISO/IEC 7816-4 (organización, seguridad y comandos para intercambio)

Una vez que se ha establecido un canal físico y sobre este un protocolo de comunicación entre el lector y la tarjeta, el protocolo del nivel de aplicación está habilitado a operar. Tal protocolo está definido en ISO/IEC 7816-4 y es utilizado en las tecnologías Smart Cards de contacto y sin contacto.

3.2.1. APDU

(ISO/IEC 7816, 2013) Define una estructura de mensaje de protocolo que consiste de APDUs (Application Protocol Data Units) que son intercambiados entre el lector y la tarjeta por el protocolo de la capa inferior. Existen 2 estructuras, una usada para enviar comandos a la tarjeta y la otra usada por la tarjeta para enviar sus respuestas y llamadas command APDU y response APDU respectivamente. Un command APDU está conformado por una cabecera y un cuerpo, que

a su vez están conformados por varios campos (ISO/IEC 7816, 2013). Los campos que componen el APDU se muestran en la Tabla 5.

Tabla 5

Campos que componen un APDU

	Campo	**Longitud (bytes)**	**Descripción**
Cabecera	CLA	1	Clase de instrucción, indica el tipo de comando, ejemplo: inter-industria o propietario
	INS	1	Código de instrucción, indica el comando específico
	P1	1	Parámetro 1 del comando
	P2	1	Parámetro 2 del comando
Cuerpo	Lc	0, 1 o 3	Número de bytes del campo Datos
	Datos	Lc	Bloque de datos
	Le	0, 1 o 3	Tamaño de la respuesta esperada

Fuente: (ISO/IEC 7816, 2013)

Los campos CLA, INS, P1 y P2 son obligatorias, mientras que Lc, el campo de Datos y Le pueden o no estar presentes, dando pie a que el cuerpo del APDU pueda tomar 4 formas distintas (ISO/IEC 7816, 2013):

1. APDU formado sólo por la cabecera: No se transfieren datos a la tarjeta y tampoco se reciben datos de esta como resultado del comando, en cuyo caso Lc, el campo Datos así como también Le son nulos.
2. APDU formado por cabecera + Le: No se transfieren datos a la tarjeta pero si se reciben datos de esta.
3. APDU formada por cabecera + Lc + Datos: Si se transfieren datos a la tarjeta pero no se reciben datos de respuesta, es decir Le es nulo.

4. APDU formada por cabecera + Lc + Datos + Le: Si se transfieren datos a la tarjeta y también se reciben datos de respuesta de esta.

La **Tabla 6** muestra el significado de cada bit dentro del byte CLA

Tabla 6

Bits que componen el byte CLA

b8	b7	b6	b5	b4	b3	b2	b1	Significado
0	-	-	-	-	-	-	-	Comando inter-industria
1	-	-	-	-	-	-	-	Comando propietario
1	1	0	-	-	-	-	-	Manejo biométrico de datos de usuario
0	0	0	0	-	-	-	-	El único o último comando de una cadena
0	0	0	1	-	-	-	-	No es el último comando de una cadena
0	0	0	-	0	0	-	-	Sin mensajes seguros (SM)
0	0	0	-	0	1	-	-	No usado
0	0	0	-	1	0	-	-	No soportado
0	0	0	-	1	1	-	-	Mensaje seguro, cabecera autenticada
0	0	0	-	-	-	0	0	Número del canal lógico (0)

Fuente: (ISO/IEC 7816, 2013)

Los response APDU están formados por un cuerpo (opcional) y una cola (obligatoria). El cuerpo puede ser nulo, dependiendo por un lado del comando específico al que responde y por otro de si el comando fue ejecutado con éxito por la tarjeta, cuando incluye datos su longitud está determinada por el campo Le del correspondiente command APDU (ISO/IEC 7816, 2013). La cola (o trailer) está formada dos bytes de estado SW1 y SW2, tal como se muestra en la **Tabla 7**.

Tabla 7

Campos que componen un responde APDU

	Campo	Longitud (bytes)	Descripción
Cuerpo	Datos	Variable	Datos de respuesta de la tarjeta
Trailer	SW1	1	Byte 1 de estado de procesamiento del comando
	SW2	1	Byte 2 de estado de procesamiento del comando

Los bytes de estado retornan un código de estado, en el que un byte es usado para especificar una categoría de error y el otro es usado para especificar un estado específico del comando o una indicación de error. Este esquema se ilustra en la ***Figura 5***. Por su parte la

Tabla 8 muestra una lista de los posibles valores que pueden tomar los bytes de estado.

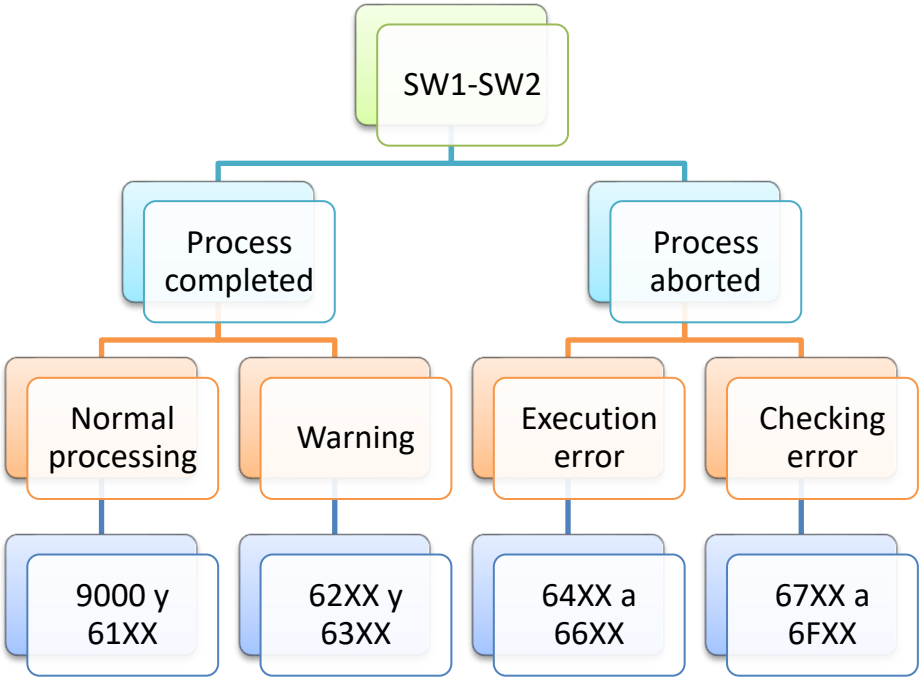

Figura 5: Bytes de estado del APDU Response

Tabla 8

Valores de los bytes de estado y su significado

	SW1	SW2	Significado	Observaciones
Normal	90	00	Sin más calificación	
	61	XX	SW2 codifica la cantidad de bytes de datos aún disponibles	

CONTINÚA

Warning	62	00	Sin información	Estado de la memoria no-volátil no ha cambiado
		02 a 80	Disparado por la tarjeta	
		81	Parte de los datos retornados podrían estar corruptos	
		82	Fin de archivo o registro alcanzado antes de leer Ne bytes	
		83	Archivo seleccionado desactivado	
		84	Información de control de archivo no formateada correctamente	
		85	Archivo seleccionado en estado de terminación	
		86	No hay datos de entrada disponibles de un sensor en la tarjeta	
Warning	63	00	Sin información	Estado de la memoria no-volátil ha sido cambiado
		81	Archivo llenado por la última escritura	
		CX	Contador de 0 a 15 codificado por X	
Error	64	00	Sin información	Estado de la memoria no-volátil no ha cambiado
		01	Respuesta inmediata requerida por la tarjeta	
		02 a 80	Disparado por la tarjeta	
Error	65	00	Sin información	Estado de la memoria no-volátil ha sido cambiado
		01	Respuesta inmediata requerida por la tarjeta	
Error	67	00	Lc incorrecto: sin cambio del estado interno	
Error	68	00	Sin información	Funciones en CLA no soportadas
		81	Canal lógico no soportado	
		82	Mensajería segura no soportada	
		83	Esperado ultimo comando de cadena	
		84	Encadenamiento de comandos no soportado	

CONTINÚA

Error	**69**	**00**	**Sin información**	**Comando no permitido**
		81	Comando incompatible con estructura de archivo	
		82	Estado de seguridad no satisfecho	
		83	Método de autenticación bloqueado	
		84	Dato de referencia no utilizable	
		85	Condiciones de uso no satisfechas	
		86	Comando no permitido (sin EF actual)	
		87	Mensajería segura esperada, objetos de datos perdidos	
		88	Incorrectos objetos de datos de mensajería segura	
Error	6A	00	Sin información	Parámetros incorrectos P1-P2
		80	Parámetros incorrectos en el campo de datos del comando	
		81	Función no soportada	
		82	Archivo o aplicación no encontrada	
		83	Registro no encontrado	
		84	No hay suficiente espacio de memoria en el archivo	
		85	Nc inconsistente con la estructura TLV	
		86	Parámetros incorrectos P1-P2	
		87	Nc inconsistente con los parámetros P1-P2	
		88	Datos referenciado o de referencia no encontrados	
		89	El archivo ya existe	
		8A	Nombre DF ya existe	

Fuente: (ISO/IEC 7816, 2013)

3.2.2. Comandos ISO 7816-4

En (ISO/IEC 7816, 2013) se define un conjunto de comandos inter-industria, orientados a 2 áreas de funcionalidad para el software de aplicación: funciones de seguridad y acceso a archivos en el sistema de archivos de la tarjeta.

Cada componente del sistema de archivos de la tarjeta tiene asociada una lista de propiedades de acceso, que restringen el acceso a los componentes del sistema de archivos mediante un procedimiento de autenticación, el cuál puede ser simple (y conseguirse en un solo comando) o complejo (y requerir la correcta consecución de una serie de comandos) (Guthery & Jurgensen, 2002). A continuación se repasan los comandos más importantes.

3.2.2.1. Comandos para Manipulación en el Sistema de Archivos

Para manipular el sistema de archivos de la tarjeta inteligente se define un protocolo de nivel de aplicación en forma de una colección de funciones para seleccionar, leer y escribir archivos, a continuación se listan los principales.

3.2.2.1.1. SELECT

Este comando selecciona un archivo por su identificador de archivo (FID), su nombre (solo DFs) o un path. Solo identificadores largos de archivo son soportados (ISO/IEC 7816, 2013). La **Tabla 9** muestra los valores que pueden tomar los campos del APDU para el comando SELECT.

Tabla 9

Comando SELECT

Comando	
CLA	Definido en la Tabla 6
INS	'A4'
P1	Codificado en la Tabla 10: P1 para comando SELECT
P2	Codificado en la Tabla 11: P2 para comando SELECT
Lc	Depende del campo Data
Data	Ausente o identificador de archivo o path o nombre del DF (de acuerdo con P1)
Le	Depende de la respuesta esperada (con o sin datos)

CONTINÚA

Respuesta	
Data	Ausente o información de control de archivo (de acuerdo a P2)
SW1-SW2	Alguno de los mostrados en la Tabla 8

Fuente: (ISO/IEC 7816, 2013)

La **Tabla 10** muestra el valor de cada bit que forma el campo P1, dependiendo de la orden deseada, para el comando SELECT y la **Tabla 11** muestra el valor de cada bit que forma el campo P2, dependiendo de la orden deseada, para el comando SELECT.

Tabla 10

Conformación del byte P1 para comando SELECT

b8	**b7**	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**Significado**	**Campo Data**
0	0	0	0	0	0	x	x	Selección por FID	
0	0	0	0	0	0	0	0	-Seleccionar MF, DF o EF	Identificador de archivo
0	0	0	0	0	0	0	1	-Seleccionar DF hijo	Identificados de DF
0	0	0	0	0	0	1	0	-Seleccionar EF	Identificador de EF
0	0	0	0	0	0	1	1	-Seleccionar DF padre	Ausente
0	0	0	0	0	1	x	x	Selección por nombre de DF	
0	0	0	0	0	1	0	0	-Seleccionar por nombre de DF	
0	0	0	0	1	0	x	x	Selección por path	
0	0	0	0	1	0	0	0	Seleccionar por MF	Path sin el MF
0	0	0	0	1	0	0	1	Seleccionar DF actual	Path sin id de DF actual

Fuente: (ISO/IEC 7816, 2013)

Tabla 11

Conformación del byte P1 para comando SELECT

b8	**b7**	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**Significado**
0	0	0	0	-	-	x	x	Ocurrencia del archivo
0	0	0	0	-	-	0	0	-Primera o única ocurrencia
0	0	0	0	-	-	0	1	-Última ocurrencia – no soportado
0	0	0	0	-	-	1	0	-Próxima ocurrencia–solo para selección por nombre y sin SM
0	0	0	0	-	-	1	1	-Previa ocurrencia

CONTINÚA

0	0	0	0	x	x	-	-	Información de control de archivo
0	0	0	0	0	0	-	-	-Retorna plantilla FCI
0	0	0	0	0	1	-	-	-Retorna plantilla FCP
0	0	0	0	1	0	-	-	-Retorna plantilla FMD – no soportado
0	0	0	0	1	1	-	-	-Sin datos de respuesta si Le está ausente, propietario si Le está presente

Fuente: (ISO/IEC 7816, 2013)

3.2.2.1.2. CREATE FILE

Este comando crea un archivo dentro del DF actual, el archivo creado pasa a ser el archivo actual. El comando solo se puede ejecutar si el estado de seguridad satisface los atributos de seguridad del DF actual (ISO/IEC 7816, 2013). La **Tabla 12** muestra la conformación del comando CREATE FILE.

Tabla 12

Comando CREATE FILE

Comando	
CLA	Definido en la Tabla 6
INS	‘E0’
P1-P2	‘0000’: FID y parámetros de archivo codificados en el campo Data P1 != ‘00’: byte descriptor de archivo P2 identificador de EF corto en los bits 8 a 4; bits 3 a 1 propietario
Lc	Depende del campo Data
Data	Plantilla FCP y posibles plantillas adicionales o ausente
Le	Depende de la respuesta esperada (con o sin datos)
Respuesta	
Data	Ausente
SW1-SW2	Alguno de los mostrados en la Tabla 8

Fuente: (ISO/IEC 7816, 2013)

3.2.2.1.3. READ BINARY

El comando lee datos solo de archivos elementales (EF). El campo de datos de respuesta tiene una parte o el total del contenido de un EF (ISO/IEC 7816, 2013). La **Tabla 13** muestra la conformación del comando READ BINARY.

- Si INS = ‘B0’ el actual EF puede ser leído o un EF puede ser referenciado por un identificador de archivo corto. O bien P1/P2 contienen un offset de 15 bits en el archivo actual o P1 contiene el SFI y P2 contiene un offset de 8 bits.
- Si INS = ‘B1’ un EF puede ser referenciado por un identificador de archivo, ya sea corto o largo. P1 y P2 contienen el identificador de archivo.

Tabla 13

Comando READ BINARY

Comando	
CLA	Definido en la Tabla 6
INS	‘B0’ o ‘B1’
P1-P2	Si INS=’B0’ o bien P1/P2 contienen un offset de 15 bits en el archivo actual o P1 contiene el SFI y P2 contiene un offset de 8 bits. Si INS = ‘B1’, P1 y P2 contienen el identificador de archivo corto o largo.
Lc	Depende del campo Data
Data	Ausente (INS = ‘B0’), u offset del objeto de datos (INS = ‘B1’)
Le	Depende de la respuesta esperada, es obligatorio
Respuesta	
Data	Datos leídos (INS = ‘B0’), o datos encapsulados (INS = ‘B1’)
SW1-SW2	Alguno de los mostrados en la Tabla 8

Fuente: (ISO/IEC 7816, 2013)

3.2.2.1.4. UPDATE BINARY

El comando solo actualiza datos en un archivo elemental (EF). Esto no cambia el tamaño de archivo. El campo de datos del comando contiene el nuevo contenido de un EF (ISO/IEC

7816, 2013). La **Tabla 14** muestra los valores que pueden tomar los campos que conforman el comando UPDATE BINARY.

- Si INS = ‘D6’ el actual EF o un EF referenciado por un identificador de archivo corto será escrito. O bien P1-P2 contienen un offset de 15 bits en el archivo actual o P1 contiene el identificador de archivo corto (SFI) y P2 contiene un offset de 8 bits.
- Si INS = ‘D7’ un EF puede ser referenciado por un identificador de archive, ya sea corto o largo. P1 y P2 contienen el identificador de archivo – ‘0000’ referencia al archivo actual.

Tabla 14

Comando UPDATE BINARY

Comando	
CLA	Definido en la Tabla 6
INS	‘D6’ o ‘D7’
P1-P2	Si INS = ‘D6’ o bien P1-P2 contienen un offset de 15 bits en el archivo actual o P1 contiene el SFI y P2 contiene un offset de 8 bits. Si INS = ‘D7’ P1 y P2 contienen el identificador de archivo corto o largo.
Lc	Depende del campo Data
Data	Cadena de bytes a ser escritos (INS = ‘D6’), u offset y cadena de bytes (INS = ‘D7)
Le	Ausente
Respuesta	
Data	Ausente
SW1-SW2	Alguno de los mostrados en la Tabla 8

Fuente: (ISO/IEC 7816, 2013)

3.2.2.2. Comandos para características de seguridad.

3.2.2.2.1. GET CHALLENGE

El comando retorna un reto (Ne bytes de datos random) para su uso en un procedimiento relacionado a la seguridad, por ejemplo en un comando EXTERNAL AUTHENTICATE. Para

Ne=8, el reto es almacenado en la tarjeta inteligente y valido al menos para el próximo comando (ISO/IEC 7816, 2013). La conformación del comando GET CHALLENGE se muestra en la **Tabla 15**.

Tabla 15

Comando GET CHALLENGE

Comando	
CLA	Definido en la Tabla 6
INS	'84'
P1-P2	'0000' (cualquier otro valor está reservado para uso futuro)
Lc	Ausente
Data	Ausente
Le	Presente para Ne > 0
Respuesta	
Data	Reto (datos random)
SW1-SW2	Alguno de los mostrados en la Tabla 8

Fuente: (ISO/IEC 7816, 2013)

3.2.2.2.2. EXTERNAL AUTHENTICATE

El comando condicionalmente actualiza el estado de seguridad usando el resultado (si o no) del cómputo de la tarjeta, basado en: un reto previamente enviado por la tarjeta, una clave almacenada en la tarjeta y datos de autenticación trasmitidos por el lector. Cualquier autenticación satisfactoria requiere el uso del último reto obtenido desde la tarjeta (ISO/IEC 7816, 2013). La **Tabla 16** muestra la conformación del comando EXTERNAL AUTHENTICATE.

Tabla 16

Comando EXTERNAL AUTHENTICATE

Comando	
CLA	Definido en la Tabla 6
INS	'82'
P1-P2	'0000' (cualquier otro valor está reservado para uso futuro)
Lc	Depende del campo data
Data	Datos relacionados con la autenticación, o ausente si se quiere saber el número de intentos restantes (SW1-SW2 = '63CX') o si la verificación es requerida o no (SW1-SW2 = '9000')
Le	Ausente
Respuesta	
Data	Ausente
SW1-SW2	Alguno de los mostrados en la Tabla 8

Fuente: (ISO/IEC 7816, 2013)

3.2.2.2.3. INTERNAL AUTHENTICATE

El comando inicia el cómputo de los datos de autenticación por la tarjeta usando el reto enviado por el lector y un secreto relevante almacenado en la tarjeta (ISO/IEC 7816, 2013).

- Si el secreto relevante está ligado al MF, entonces el comando podría ser usado para autenticar la tarjeta como un todo.
- Si el secreto relevante está ligado a otro DF, entonces el comando podría ser usado para autenticar ese DF.

Cualquier autenticación exitosa podría estar sujeta a la conclusión de comandos anteriores (por ejemplo VERIFY, SELECT) o selecciones (por ejemplo el secreto relevante). Los valores de los campos que conforman el comando INTERNAL AUTHENTICATE se muestran en la

Tabla 17.

Tabla 17

Comando INTERNAL AUTHENTICATE

Comando	
CLA	Definido en la Tabla 6
INS	'88'
P1	Algoritmo criptográfico o biométrico a usar o '00'
P2	De acuerdo con la tabla 18 o '00'
Lc	Depende del campo data
Data	Datos relacionados con la autenticación
Le	Depende de la respuesta esperada
Respuesta	
Data	Datos relacionados con la autenticación
SW1-SW2	Alguno de los mostrados en la Tabla 8

Fuente: (ISO/IEC 7816, 2013)

La **Tabla 18** muestra la forma en que se debe codificar el calificador de datos de referencia en P2.

Tabla 18

Configuración del byte P2 para INTERNAL AUTHENTICATE

b8	**b7**	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**Significado**
0	0	0	0	0	0	0	0	Sin información
0	-	-	-	-	-	-	-	Dato de referencia global (ej.: clave de MF específico)
1	-	-	-	-	-	-	-	Dato de referencia específico (ej.: clave de DF específico)
-	x	x	-	-	-	-	-	'00'
-	-	-	X	x	x	x	x	Calificador, ej.: número del dato de referencia o del secreto

Fuente: (ISO/IEC 7816, 2013)

3.2.2.2.4. MANAGE SECURITY ENVIRONMENT (MSE)

El comando prepara el mensaje seguro (SM) y otros comandos de seguridad (ej.: EXTERNAL, INTERNAL y GENERAL AUTHENTICATE y PERFORM SECURITY OPERATION) (ISO/IEC 7816, 2013). Véase la **Tabla 19**.

Tabla 19

Conformación del comando MANAGE SECURITY ENVIRONMENT

Comando	
CLA	Definido en la Tabla 6
INS	'22'
P1	De acuerdo con la Tabla 3.18
P2	De acuerdo con la Tabla 3.19
Lc	Depende del campo data
Data	Datos específicos del comando
Le	Ausente
Respuesta	
Data	Ausente1
SW1-SW2	Alguno de los mostrados en la Tabla 8

Fuente: (ISO/IEC 7816, 2013)

La **Tabla 20** muestra la forma en que se codifica el parámetro P1 para el comando *MANAGE SECURITY ENVIRONMENT* y la **Tabla 21** muestra los valores que puede tomar P2 para el comando MSE.

Tabla 20

Conformación del byte P1 para MSE

b8	**b7**	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**Significado**
-	-	-	1	-	-	-	-	Mensaje seguro en el campo datos del comando
-	-	1	-	-	-	-	-	Mensaje seguro en el campo datos de respuesta
-	1	-	-	-	-	-	-	Calcular, descifrar, autent. interna y acuerdo de clave
1	-	-	-	-	-	-	-	Verificar, cifrar, autent. externa y acuerdo de clave
-	-	-	-	0	0	0	1	SET
1	1	1	1	0	0	1	0	STORE (no soportado)
1	1	1	1	0	0	1	1	RESTORE
1	1	1	1	0	1	0	0	ERASE (no soportado)

Fuente: (ISO/IEC 7816, 2013)

Tabla 21

Posibles valores del byte P2 para MSE

Valor	**Significado**
'XX'	SEID en el caso de STORE, RESTORE y ERASE ('00' en caso de SET)

CONTIÚA

	Etiqueta de la plantilla de referencia de control presente en el campo data
‘A4’	Plantilla de referencia de control para autenticación (AT)
‘A6’	Plantilla de referencia de control para acuerdo de claves (KAT)
‘AA’	Plantilla de referencia de control para código hash (HT)
‘B4’	Plantilla de referencia de control para suma de chequeo criptográfica (CCT)
‘B6’	Plantilla de referencia de control para firma digital (DST)
‘B8’	Plantilla de referencia de control para confidencialidad (CT)
Cualquier otro valor es reservado para uso futuro	

Fuente: (ISO/IEC 7816, 2013)

3.2.2.2.5. MUTUAL AUTHENTICATE

Esta función usa las mismas funcionalidades que los comandos EXTERNAL e INTERNAL AUTHENTICATE. Está basada en un previo comando GET CHALLENGE y una clave, posiblemente secreta, almacenada en la tarjeta. La tarjeta y el lector comparten datos relacionados con la autenticación, incluyendo dos retos: uno generado por la tarjeta, y otro generado por el lector. La operación puede ser realizada solo si el estado de seguridad satisface los atributos de seguridad para esta operación (ISO/IEC 7816, 2013). En la **Tabla 22** se explica la conformación del comando MUTUAL AUTHENTICATE.

Tabla 22

Conformación del comando MUTUAL AUTHENTICATE

Comando	
CLA	Definido en la Tabla 6
INS	‘82’
P1	Algoritmo criptográfico o biométrico a usar o ‘00’
P2	De acuerdo con la tabla 3.16 o ‘00’
Lc	Depende del campo data
Data	Datos relacionados con la autenticación
Le	Depende de la respuesta esperada
Respuesta	
Data	Datos relacionados con la autenticación
SW1-SW2	Alguno de los mostrados en la Tabla 8

Fuente: (ISO/IEC 7816, 2013)

3.2.2.2.6. PERFORM SECURITY OPERATION

Inicia las operaciones de seguridad consecuentes, de acuerdo a los datos especificados en P1-P2. Estas operaciones pueden ser:

- Calculo / Verificación de una suma de comprobación criptográfica
- Calculo de una firma digital
- Calculo de un código hash
- Verificación de un certificado
- Cifrar y descifrar

La clave de referenciada así como el algoritmo referenciado se deben conocer implícitamente o deben ser especificados en una plantilla de referencia de control en un comando MANAGE SECURITY ENVIRONMENT. Cualquier comando PERFORM SECURITY OPERATION solo puede ser realizado si el estado de seguridad satisface los atributos de seguridad para la operación (ISO/IEC 7816, 2013). La **Tabla 23** explica cómo se debe conformar el comando PERFORM SECURITY OPERATION.

Tabla 23

Conformación del comando PERFORM SECURITY OPERATION

Comando	
CLA	Definido en la Tabla 6
INS	'2A'
P1	Etiqueta (datos en la respuesta) o '00' (sin datos en la respuesta)
P2	Etiqueta (datos en el comando) o '00' (sin datos en el comando)
Lc	Depende del campo data
Data	Ausente o valor del objeto de datos especificado en P2
Le	Depende de la respuesta esperada
Respuesta	

CONTINÚA

Data	Ausente o valor del objeto de datos especificado en P1
SW1-SW2	Alguno de los mostrados en la Tabla 8

Fuente: (ISO/IEC 7816, 2013)

3.3. La interacción chip RFID – PC

Una tarjeta inteligente es un medio seguro por naturaleza, capaz de proporcionar un almacén seguro para información sensible como: claves privadas, números de cuenta o contraseñas. Además provee un procesamiento aislado que puede utilizar dicha información sin que sea expuesta al entorno del PC, importante para operaciones como: generación de firmas digitales usando claves privadas, autenticación de red basada en secretos o créditos para compras pre-pago. Pero actualmente no ha recibido el uso esperado en el entorno PC por la falta de interoperabilidad en varios niveles (PC/SC Workgroup, 2013):

- En primer lugar, la industria carece de estándares para interconectar PC con periféricos lectores de tarjetas inteligentes.
- En segundo lugar, no existe una interfaz de programación de aplicaciones (API) de alto nivel ampliamente aceptada para la funcionalidad común de las tarjetas inteligentes, algo que de existir permitiría a las aplicaciones reducir su dependencia sobre una implementación de tarjeta específica.
- En tercer lugar no se definen los mecanismos para permitir que múltiples aplicaciones efectivamente compartan los recursos de una única tarjeta, críticos para la usabilidad de tarjetas multi-aplicación y tarjetas criptográficas genéricas que serán usadas en un ambiente de PC multiproceso.

La especificación PC/SC pretende ser el framework establecido con la finalidad de resolver estos problemas.

3.3.1. Especificación PC/SC

PC/SC es una especificación que facilita la integración Smart Card (SC) en el entorno del Computador Personal (PC), su fin es por una lado asegurar la interoperabilidad de Smart Cards (o Integrated Circuit Cards - ICCs), smart card readers (Interface Devices - IFDs) y computadores y por otro facilitar el desarrollo de aplicaciones Smart Card para PC y otras plataformas de cómputo gracias a la definición de una API (interfaz de programación de aplicaciones) (PC/SC Workgroup, 2013). Es desarrollada por el grupo de trabajo PC/SC, del que actualmente forman parte compañías como Gemalto, HID Global o HP entre otras. La ***Figura 6*** muestra la interacción de una PC con una Smart Card (ICC), a través de un dispositivo lector (IFD).

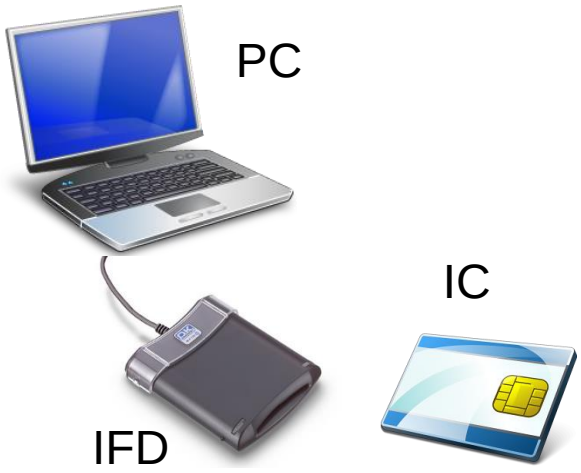

Figura 6: Uso de Smart Cards en el entorno PC

Gracias a la especificación PC/SC las aplicaciones de PC no necesitan tener conocimiento de detalles correspondientes a los lectores de tarjetas inteligentes cuando se comunican con una smart card (Alonso, 2013). La especificación está dividida en 10 partes (PC/SC Workgroup, 2013):

- Parte 1: proporciona una revisión de la arquitectura del sistema y componentes definidos por el grupo de trabajo.
- Parte 2: Detalles de compatibilidad ICC-IFD, características y requerimientos de interoperabilidad.
- Parte 3: Describe la interfaz, y la funcionalidad requerida para dispositivos IFD compatibles. Existe también un suplemento para proporcionar información sobre números RID.
- Parte 4: Discute consideraciones de diseño para dispositivos IFD. En particular, proporciona una implementación recomendada para IFDs integrados a teclados PS/2.
- Parte 5: Describe las interfaces y funcionalidad soportada por el ICC Resource Manager, un componente de nivel de sistema requerido.
- Parte 6: Describe el modelo ICC Service Provider, identifica las interfaces requeridas, e indica como puede ser ampliado para cumplir los requisitos específicos del dominio de aplicación.
- Parte 7: Describe consideraciones de diseño para desarrolladores de aplicaciones, y cómo hacer uso de otros componentes.

- Parte 8: Describe funcionalidad recomendada para los ICCs destinados a soportar requisitos criptografía y de almacenamiento de propósito general. Esto está orientado hacia el soporte de estándares de Internet y PC para seguridad y privacidad.
- Parte 9: Describe la administración de los IFDs con algunas capacidades extendidas como entrada para PIN de seguridad o funcionalidad de interfaz de usuario.
- Parte 10: Describe la administración de los IFDs con capacidad de estrada para PIN de seguridad.
- Apéndice: Lista de referencias ISO.

3.3.1.1. Componentes de la Arquitectura PS/SC

La arquitectura definida en la especificación está conformada por varios componentes mostrados en la ***Figura 7***. La aplicación ofrece al usuario el interfaz de comunicación, y accede al lector por medio del gestor de recursos (ICC Resource Manager), que se encarga de controlar qué lectores tiene disponible el PC y si éstos tienen o no una tarjeta conectada. Para poder hacer uso de los lectores es necesario que el controlador correspondiente esté instalado, así como el del dispositivo de entrada/salida, pues un lector puede conectarse de diversas formas, ya sea de manera inalámbrica, por USB o por puerto serie (Alonso, 2013).

El acceso a los dispositivos ICC se hace mediante aplicaciones basadas en PC, a través de un dispositivo periférico IFD. Puede haber múltiples IFD por sistema, y son compatibles una variedad de canales de E/S. Asociado a cada IFD en el sistema existe un IFD Handler. Las aplicaciones escritas para aprovechar esta arquitectura generalmente utilizarán tanto un Resource Manager como un ICC Service Provider específico (PC/SC Workgroup, 2013). A continuación se mencionan los elementos que conforman la arquitectura PC/SC:

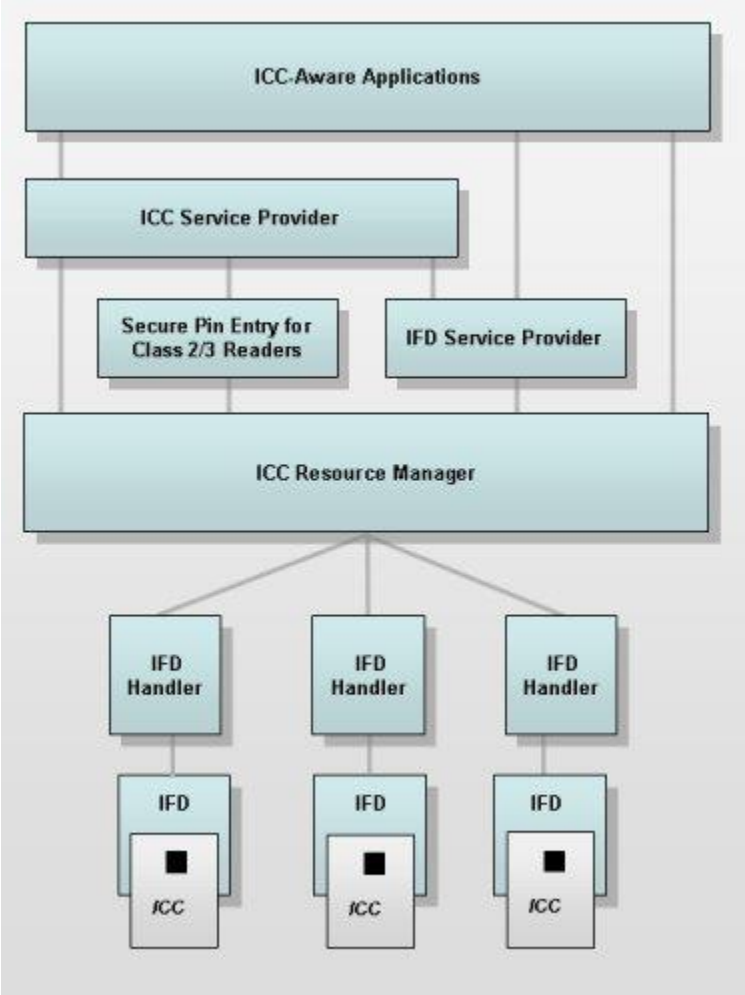

Figura 7: Arquitectura definida para PS/SC
Fuente: (PC/SC Workgroup, 2013)

- Integrated Circuit Card (ICC): Smart card ya sea con contacto o sin contacto.
- Interface Device (IFD): Lector de tarjetas inteligentes, a través del cual el ICC se comunica con la PC.
- ICC Aware Application: Es una aplicación dentro del ambiente operativo del PC, que quiere hacer uso de la funcionalidad proporcionada por uno o más ICCs.

3.3.1.1.1. Interface Device Handler (IFD Handler)

Es el controlador del IFD instalado en el PC, un software de bajo nivel que permite al PC usar la funcionalidad del IFD. Se distinguen dos variedades (PC/SC Workgroup, 2013):

- Slot Logical Devices**:** Es el término, en lado del PC, de los protocolos de comunicación ICC (ISO 7816-3, ISO 7816-10 O ISO/IEC 14443/15693). En el nivel del IFD Handler la API oculta todas las distinciones entre ICCs basados en el manejo del protocolo ISO, ya sea síncrono o asíncrono.
- Functional Logical Devices**:** Su principal rol es permitir al IFD Service Provider bloquear independientes tipos de funcionalidad del IFD a través del Resource Manager.

3.3.1.1.2. ICC Resource Manager

Componente clave de la arquitectura, responsable de gestionar los otros recursos relevantes ICC dentro del sistema y apoyar el acceso controlado a los IFD y, a través de ellos, los ICC individuales. Resuelve tres problemas básicos en la gestión del acceso a múltiples IFDs e ICCs (PC/SC Workgroup, 2013):

a) Es responsable de la identificación y seguimiento de los recursos: IFDs instalados, tipos de ICC conocidos, eventos de inserción y remoción.
b) Es responsable de controlar la asignación de IFDs y recursos (y por lo tanto acceso a ICCs) en múltiples aplicaciones.
c) Gracias a que admite primitivas de transacción, permite que varios comandos se ejecuten sin interrupción para completar una transacción dentro del ICC.

3.3.1.1.3. Service Providers

Los Service Providers son responsables de encapsular la funcionalidad expuesta por un ICC o IFD específico, y hacerla accesible a través de interfaces de programación de alto nivel. Estas interfaces pueden ser mejoradas y ampliadas para satisfacer las necesidades de los dominios de aplicación específicos. Existen dos tipos:

a) ICC Service Provider (ICCSP): es una interface a la funcionalidad del ICC. Existen diferentes tipos de ICCSP: los ICC Operating System Service Providers (ICCOSSP) son la interface a la funcionalidad de un sistema operativo de ICC específico, mientras los Service Providers que sirven de interface a una aplicación en la tarjeta (on-card) se denominan Application Domain Service Provider (ADSP). El Cryptographic Service Provider (CSP) encapsula acceso a la funcionalidad criptográfica de un ICC a través de interfaces de programación de alto nivel para servicios de propósito general como: generación y administración de claves, firmas digitales, digesto de mensajes o importación/exportación de claves (PC/SC Workgroup, 2013).
b) IFD Service Provider (IFDSP): encapsula acceso, e interfaces a la funcionalidad de un IFD en la misma forma en que un ICC Service Provider sirve de interface a la funcionalidad de un ICC. La implementación de la interfaz del IFDSP interactúa con la implementación del controlador IFD (IFD Handler) en un modo que es transparente para el Resource Manager (PC/SC Workgroup, 2013).

La arquitectura general de PC/SC puede además ser presentada como un protocolo de comunicación punto a punto, como se ilustra en la ***Figura 8***.

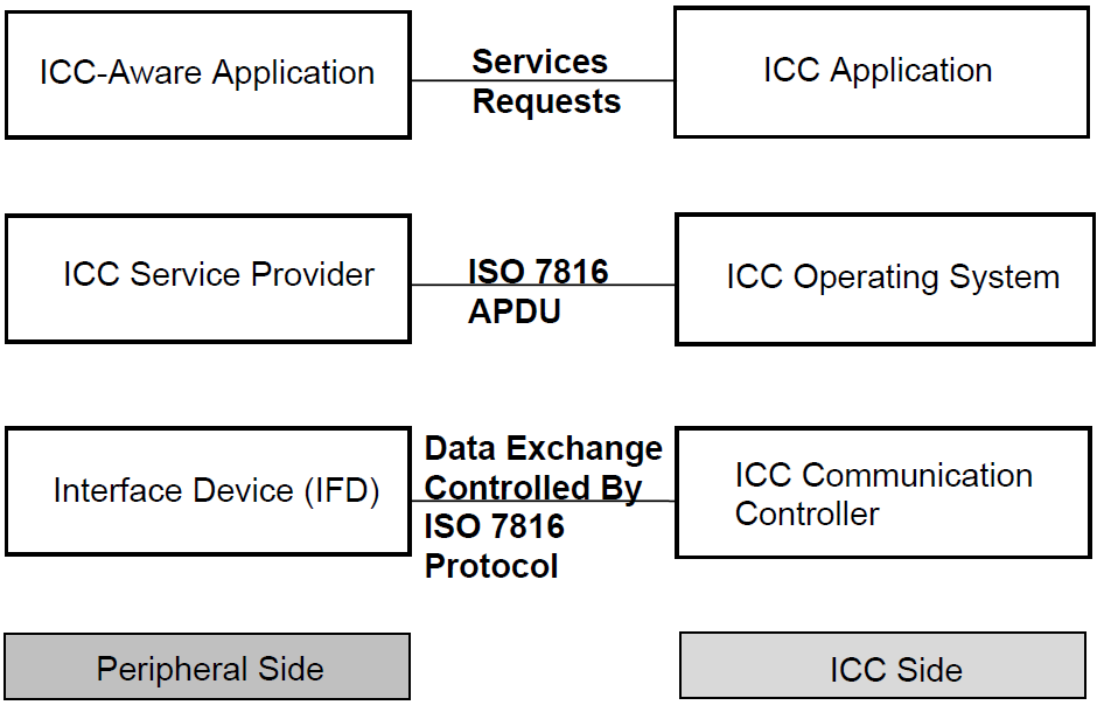

Figura 8: Arquitectura PC/SC como un protocolo punto a punto.
Fuente: (PC/SC Workgroup, 2013)

Micosoft Windows implementa la especificación PC/SC por medio del API WinSCard mientras que para Linux existe el proyecto PC/SC -Lite pero actualmente no soporta toda la funcionalidad (Schalk & Bienert, 2013). En cuanto a lenguajes de programación se refiere, Java posee una completa API llamada Smart Card I/O que soporta la comunicación con lectores y smart cards PC/SC, por medio de las clases definidas en el paquete javax.smartcardio. Para C/C++ en Windows se puede usar directamente el API WinSCard, ya que esta está escrita enteramente en lenguaje C. Al día de hoy el framework .NET no incluye soporte para smart card, por lo que el desarrollo en lenguajes como C# o Visual Basic requiere una interface (o wrapper) entre el lenguaje de programación y el API WinSCard de Windows (Schalk & Bienert, 2013).

4 OPERACIONES CRIPTOGRÁFICAS PARA VALIDAR ePASSPORTS

Este capítulo está dedicado a la descripción de las operaciones criptográficas que intervienen durante en el proceso de acceso al documento para validar su contenido. El propósito de este capítulo es, por un lado facilitar el entendimiento de tales algoritmos para su implementación en la herramienta propuesta, y por otro realizar una comparación de los distintos algoritmos de firma digital que sirva de pauta al decantarse por determinada opción (plasmada en el tipo de claves de los certificados de Firmante de País CSCA y Firmante de Documentos DS).

4.1. Operaciones Criptográficas Comunes

En esta sección de describen aquellas operaciones criptográficas de uso común por los mecanismos de seguridad para pasaportes electrónicos ya mencionados en la sección 2.5.1. Estas operaciones de uso común son principalmente la Función de Derivación de Claves (KFD) y los Mensajes Seguros (SM).

4.1.1. Función de Derivación de Claves (KFD)

Se trata de un algoritmo simétrico para obtención de claves, ya sea 3DES (Triple Data Encryption Standard) o AES (Advanced Encryption Standard), a partir de un secreto compartido (Kseed), y un entero big endian (c) de 32 bits. Según (ICAO, 2015) KDF(K, c) se define así:

keydata = SHA-1(Kseed || c)

Para obtener claves 3DES de 128 bits:

- keydataA = bytes 1 a 8 de keydata
- keydataB = bytes 9 a 16 de keydata
- Se deben ajustar los bits de paridad de keydataA y keydataB.

Para obtener claves AES de 128 bits:

- Se deben usar los primeros 16 bytes.

4.1.2. Mensajes Seguros (SM)

Mensajes seguros es como se llama al mecanismo que permite asegurar el canal de comunicación entre el dispositivo lector y el pasaporte, logrando de esta manera confidencialidad y autenticación de los datos. Una vez establecido SM, cada APDU intercambiado entre el pasaporte y el lector tiene sus datos cifrados (usando una clave de sesión de encriptación KS_{ENC}) además se le ha añadido un código de autenticación de mensaje MAC (usando una clave de sesión de autenticación de mensaje KS_{MAC}). Según (ISO/IEC 7816, 2013) un mensaje seguro puede basarse en 3DES o AES, y está formado por Objetos de Datos codificados con BER TLV, con las etiquetas correspondientes y se construye partiendo de un APDU no protegido. En (ICAO, 2015) se describe el proceso de construcción de un command APDU de mensaje seguro, así como también la verificación del response APDU:

a) En la cabecera, se establece el valor del byte CLA en ‘0C’, y se rellena agregando los 4 bytes: ‘80000000’:
    - CmdHeader = ‘0C’ ‘INS’ ‘P1’ ‘P2’ ‘80’ ‘00’ ‘00’ ‘00’
b) Se rellena el campo data con ‘8000…’ para que su longitud en bytes sea múltiplo de 8:
    - Data = Data || ‘80’ ‘00’ …
c) Se cifra Data con la clave KS_{ENC} (puede usarse 3DES o AES):
    - $EncData = E(KS_{ENC}, Data)$
d) Se construye el DO de etiqueta ‘87’:
    - DO87 = ‘87’ ‘L’ ‘01’ || EncData

e) Se deben concatenar CmdHeader y DO87:
   - M = CmdHeader || DO87

f) Se calcula MAC de M

   i. Se incrementa SSC en 1
      - SSC = SSC +1

   ii. Se concatena SSC y M y se añade relleno, para que la longitud sea múltiplo de 8
      - N = SSC || MAC ‘80’ ‘00’ …

   iii. Se calcula MAC de N con KS_{MAC}:
      - CC = MAC(KS_{MAC}, N)

g) Se construye el DO ‘8E’ (L = 08):
   - DO8E = ‘8E’ ‘08’ || CC

h) Se construye el APDU protegido calculando el nuevo valor de Lc y usando Le = ‘00’:
   - ProtectedAPDU = CmdHeader || Lc || DO87 || DO8E || Le

i) Se recibe el response APDU del ePassport
   - RAPDU = ‘99’ ‘02’ || SW1-SW2 || ‘8E’ ‘08’ || CC || SW1-SW2

j) Se verifica el CC de RAPDU calculando el MAC de DO99

   i. Se incrementa SSC en 1:
      - SSC = SSC + 1

   ii. Se concatena SSC y DO99 y se añade relleno
      - K = SSC || ‘99’ ‘02’ ‘SW1’ ‘SW2’ ‘80’ ‘00’ ‘00’ ‘00’

   iii. Se calcula el MAC de K con KS_{MAC}:
      - CC’ = MAC(KS_{MAC}, K)

   iv. Se compara el CC’ calculado con el CC de RAPDU

- CC = CC’
- Si son iguales, el RAPDU ha sido autenticado satisfactoriamente

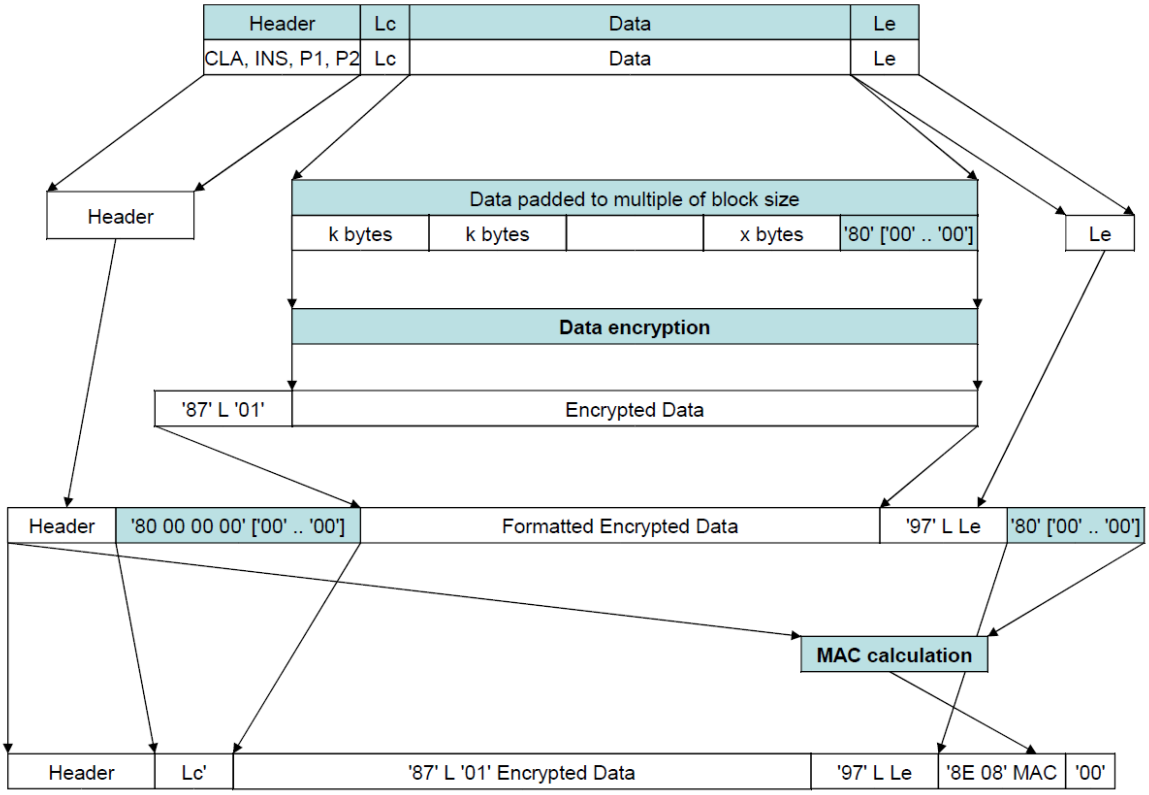

Figura 9: Proceso de construcción de un command APDU seguro
Fuente: (BSI, 2015)

El proceso descrito en líneas anteriores únicamente contempla el caso de un comando APDU en el que el campo Data no es nulo y el byte Le (longitud de la respuesta esperada) si lo es. Si el byte Le no es nulo, se lo debe envolver dentro de un DO con etiqueta ´97´. El DO97 más un relleno debe usarse en el cálculo del MAC, tal como se observa en la ***Figura 9***.

4.2. Descripción de los mecanismos de seguridad

Como se pudo apreciar en la sección 2.5.1 existen 2 opciones (BAC y PACE) para proteger el acceso al chip y 3 opciones para prevenir la sustitución del mismo. El presente trabajo

busca definir una metodología para validar pasaportes electrónicos, y por lo tanto no es necesario considerar todas las opciones disponibles, sino que basta con elegir una opción para proteger cada aspecto de seguridad del pasaporte electrónico. De tal manera que los mecanismos que se cubrirá en este trabajo son:

- Para validar los datos en el chip: Autentificación Pasiva, pues es el único mecanismo disponible y por supuesto de cumplimiento obligatorio según ICAO.
- Para el control de acceso al chip: BAC, puesto que su ejecución es obligatoria en un proceso de inspección según ICAO, y además para la elaboración de este trabajo no se cuenta con acceso a documentos que implementen PACE y por tal motivo no se lo implementará en la herramienta que se desarrollará. De todas maneras en vista de que PACE es una recomendación de ICAO, en este capítulo se realizará una breve descripción de su funcionamiento.
- Para prevenir la sustitución del chip: Autenticación Activa, porque se puede implementar de forma independiente (a través del Grupo de Datos 15) a otros mecanismos de seguridad, a diferencia de lo que ocurre con la Autenticación del chip que forma parte de EAC y de PACE-CAM que opera con PACE. Y recordemos que ni EAC ni PACE se considerarán en este trabajo.

4.2.1. Control de Acceso Básico (BAC)

El método de control de acceso BAC, especificado en (ICAO, 2015), se puede dividir en tres etapas secuenciales: obtención de las claves BAC, autenticación mutua y cálculo de las claves de sesión.

4.2.1.1. Obtención de las claves de acceso básico al documento (Kenc y Kmac)

- MRZ_information = número de documento || fecha de nacimiento || fecha de caducidad
- H = SHA-1(MRZ_information)
- K_{SEED} = 16 bytes más significativos de H
- K'_{ENC} = 16 bytes más significativos de SHA-1(K_{SEED} || 00000001)
- K'_{MAC} = 16 bytes más significativos de SHA-1(K_{SEED} || 00000002)
- K_{ENC} = ajuste de bits de paridad (K'_{ENC})
- K_{MAC} = ajuste de bits de paridad (K'_{MAC})

4.2.1.2. Autenticación mutua

Se usa el protocolo de desafío-respuesta de tres pasadas (mecanismo para establecimiento de claves 6 de ISO/IEC 11770-2) usando 3DES para calcular los cifrados E_{IFD} y E_{IC} (según ISO/IEC 11568-2) en modo CBC con IV cero (8 bytes a cero). Se usa el algoritmo MAC 3 para calcular los códigos de autenticación de mensaje M_{IFD} y M_{IC} (según ISO/IEC 9797-1) que deben ser de 8 bytes, con DES para cifrar, IV cero (8 bytes a cero) y el método 2 de ISO/IEC 9797-1 para padding o relleno (ICAO, 2015). La ***Figura 10*** explica el proceso de autenticación mutua usado en BAC.

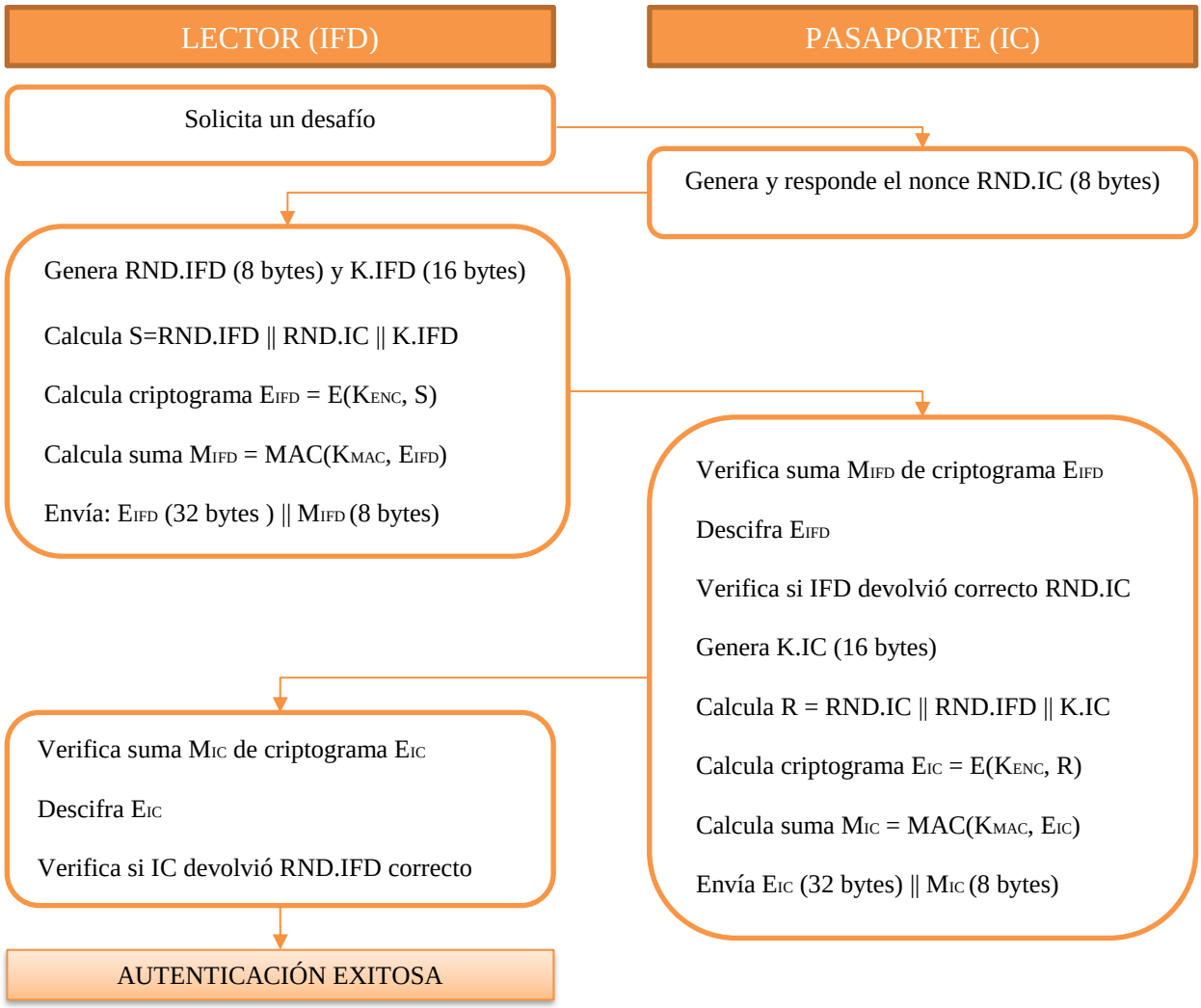

***Figura 10*:** Protocolo BAC (Autenticación Mutua)

4.2.1.3. Cálculo de las claves de sesión KS_{ENC} y KS_{MAC}

Una Vez que se ha realizado una autenticación mutua satisfactoria, las claves de sesión para Mensajes Seguros deben calcularse de la siguiente forma:

- Clave semilla Ks = K.IC xor K.IFD
- KS_{ENC} = KFD(Ks, c1) y KS_{MAC} = KFD(Ks, c2)
- Con c1 = 0000001 y c2 = 0000002

Se debe calcular el Contador de secuencia de envío (SSC):

- SSC = 4 bytes menos significativos de RND.IC || 4 bytes menos significativos de RND.IFD

  En adelante todas las comunicaciones deben estar protegidas con Mensajes Seguros.

4.2.2. Establecimiento de conexión autenticada por contraseña (PACE)

PACE es un protocolo de acuerdo de claves Diffie-Hellman en el que tanto el CI del documento como el sistema de inspección se autentifican mutuamente basándose en una contraseña compartida (Π), que puede ser obtenida del MRZ al igual que en BAC, o puede ser el Card Access Number (CAN) si este está impreso en la página de datos del documento, los parámetros para su utilización (cifrador simétrico, algoritmo de acuerdo de claves, parámetros de dominio, y mappings) se encuentran en el archivo EF.CardAccess, que está en el fichero maestro del chip RF (ICAO, 2015). En PACE se parte de una contraseña débil (MRZ_information o CAN) para generar claves de sesión fuertes para Mensaje Seguro SM (BSI, 2015). Uno de los componentes claves de PACE es su llamada función de mapeo (mapping), que se utiliza para mapear un número aleatorio a los parámetros utilizados para la criptografía asimétrica. Actualmente existen tres alternativas de mapping (Bender, 2009):

a) Mapping Genérico, basado en operaciones de grupo genéricas. Esto se puede ser genéricamente adaptado a todos los sistemas de criptografía asimétrica y es fácil de implementar en Smart cards.
b) Mapping Integrado por la cual el número aleatorio está directamente integrado en los parámetros utilizados para la criptografía asimétrica. Si bien esto es fácil de implementar para la criptografía estándar, requiere algoritmos más sofisticados para criptografía de curva elíptica (como la nueva función Hash2Point desarrollada por Thomas Icart).

c) Mapping de Autenticación de Chip que amplía el Mapping Genérico e integra Autenticación de Chip en el protocolo PACE.

Según se especifica en (ICAO, 2015) el chip RF y el Terminal deben realizar los siguientes pasos usando una cadena de comandos GENERAL AUTHENTICATE:

1. El chip aleatoriamente elige un número random, lo encripta usando la clave derivada de la contraseña compartida y envía el número aleatorio cifrado al terminal, donde es recuperado.
2. Ambos el chip y el terminal usan una función mapping para mapear el número aleatorio a parámetros `para criptografía asimétrica.
3. El chip y el terminal ejecutan el protocolo Diffie-Hellman basado en los parámetros generados durante el paso 2.
4. El chip y el terminal derivan claves de sesión, que son confirmadas intercambiando y chequeando los tokens de autenticación.

4.2.3. Autenticación Pasiva (PA)

La autenticación pasiva es en si la autenticación de los datos contenidos en el chip RF del pasaporte, y se llama “pasiva” porque no requiere que el CI sin contacto tenga capacidad de procesamiento. La PA consiste en verificar la firma digital del objeto de seguridad del documento o SOD. Para esto se requiere tener acceso a la clave pública del firmante de documento DS, que también puede ser obtenida leyendo el certificado de DS (C_{DS}) del ePassport. Al verificar la firma digital del SOD se verifica su autenticidad, y puesto que el SOD contiene un hash de cada grupo de datos contenido en la LDS, también puede verificar la autenticidad del contenido de la LDS. La Autenticación Pasiva especificada en (ICAO, 2015) consiste de los siguientes pasos:

1) Se debe leer del CI sin contacto el objeto de seguridad SOD que también contiene el certificado de firmante de documento DS_C.
2) Se debe validar la cadena de confianza del DS_C usado para firmar el objeto de seguridad SOD. Esta validación incluye la verificación del estado de revocación del DS_C. Más adelante en la sección 4.2.3.1 se da más detalles sobre el proceso de la validación de la ruta de certificación, mientras que en la sección 4.2.3.2 hay información adicional sobre el proceso de chequeo del estado de revocación y la validación de una CRL.
3) Usando la clave pública del firmante de documento, validada en el paso previo, se debe verificar la firma del objeto de seguridad SOD (ver ***Figura 11***).

Si las verificaciones de 2 y 3 resultan correctas, entonces el contenido de SOD es auténtico.

4) Se lee del CI sin contacto los grupos de datos pertinentes.
5) Para cada grupo de datos leído, se debe verificar su autenticidad e integridad, calculando el hash de su contenido y comparándolo con el correspondiente valor contenido en el objeto SOD.

Si los valores hash del paso 5 son idénticas, entonces el contenido del grupo de datos no ha sufrido cambios. En la ***Figura 11*** se representa el proceso de validación de la cadena de confianza en la Autentificación Pasiva.

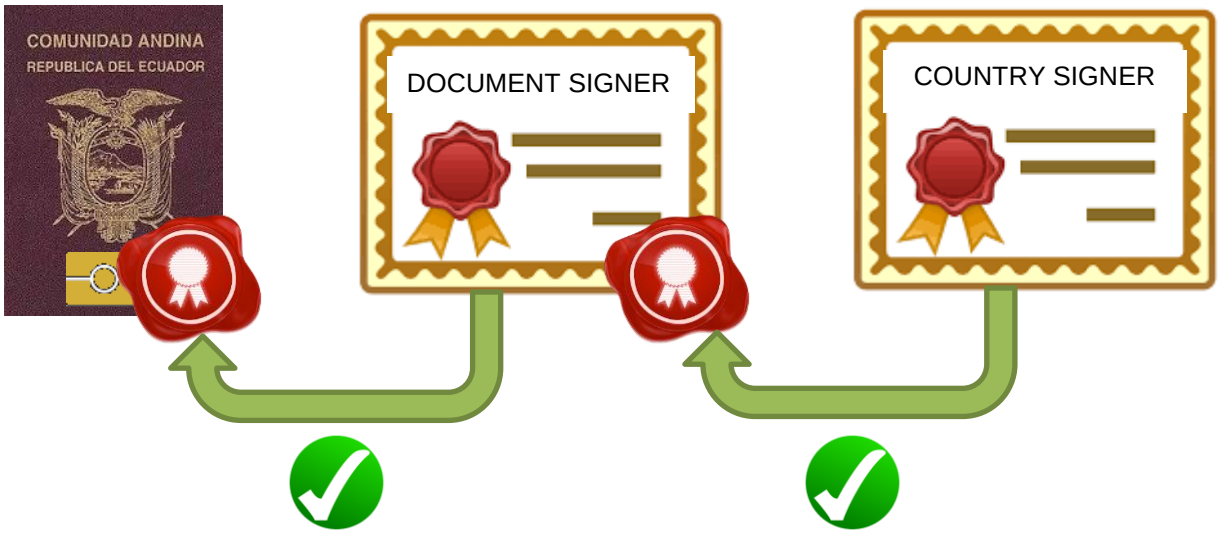

***Figura 11*:** Proceso de validación en la Autenticación Pasiva

El Doc9303 (ICAO, 2015) también menciona las siguientes verificaciones adicionales consideradas buenas prácticas:

- Chequear la presencia de DocumentTypeExtension en el certificado DS_C.
    - Si existe, debería ser consistente con el Tipo de Documento del Grupo de Datos 1.
    - Si no, chequear que KeyUsage del certificado DS este establecido en digitalSignature y que el certificado DS_C no contiene la extensión ExtendedKeyUsage.
- Chequear la consistencia de los códigos de país de:
    - Campo Subject y si está presente SubjectAltName del certificado DS_C;
    - Campo Subject y si está presente SubjectAltName del certificado CSCA;
    - Del Grupo de Datos 1; y
    - En el MRZ impreso en el documento.

Adicionalmente se debería comparar el contenido del Grupo de Datos 1 con el MRZ.

- Verificar que la fecha de emisión del documento esté incluida en el Periodo de Uso de la Clave Privada en el certificado DS_C.

4.2.3.1. Validación de la ruta de certificación

Para la validación de la ruta de certificación en una PKI para eMRTD se puede usar el algoritmo definido en RFC 5280, pero únicamente se necesita un subconjunto de sus etapas (ICAO, 2015). A continuación se mencionan las etapas pertinentes a los eMRTD, que requieren como entrada la siguiente información sobre el punto de confianza:

- Nombre del expedidor de confianza (campo subject del certificado CSCA).
- Algoritmo de clave pública de confianza.
- Clave pública de confianza.
- Parámetros de clave pública de confianza, si el algoritmo de clave pública los requiere.

El procesamiento de cada certificado para validar su ruta de certificación consiste en:

- Verificar la información básica del certificado:
    - La firma del certificado, usando el Algoritmo de clave pública, la calve pública, y los parámetros de clave pública.
    - Que el periodo de validez incluya la hora actual.
    - Que en el momento actual el certificado no esté revocado.
    - Que el nombre del expedidor del certificado se corresponde con el Nombre del expedidor de confianza.
- Reconocer y procesar cualquier otra extensión crítica presente en el certificado.

Ante un fallo en la verificación de la información básica del certificado o si existe una extensión crítica no reconocida que no puede ser procesada, el procedimiento de validación de la ruta falla.

4.2.3.2. Verificación de revocación y validación de las CRL

Parte de la validación del certificado DS_C, es también validar la CRL usada para verificar la revocación de dicho certificado y por supuesto procesar la CRL para verificar el estado de revocación del DS_C. Validar la CRL consiste en verificar su firma, puesto que debería estar firmada por el firmante de país CSCA. La validación de las CRL también puede hacerse usando una versión simplificada del algoritmo definido en RFC 5280, la información que se requiere del certificado cuyo estado de revocación se quiere verificar (en este caso el certificado DS_C) es: el número de serie y el nombre del expedidor. Únicamente las siguientes etapas se requieren para validar una CRL y chequear una revocación (ICAO, 2015):

a) Obtener la CRL actual de la CSCA que expidió el certificado.
b) Verificar que el expedidor de la CRL y el certificado es la misma CSCA, basta con comprobar que el nombre del país, contenido en el campo expedidor, sea el mismo en ambos.
c) Validar la ruta de certificación para el expedidor de la CRL. En este caso el CSCA expedidor es el punto de confianza de la ruta de certificación.
d) Verificar la firma de la CRL.
e) Buscar le certificado en la CRL por nombre del expedidor y número de serie.

Si las etapas a), b), c) o d) fallan, el estado resultante será UNDETERMINED, si el certificado se encontró dentro de la CRL el estado será UNSPECIFIED, y si la validación de la CRL fue exitosa y el certificado no encontró dentro de la CRL, el estado será UNREVOKED.

4.2.4 Autenticación Activa (AA)

Es implementada por medio de un par de claves, individual para cada documento, la pública almacenada en DG15 y la privada en la memoria segura y puede ser usada solo internamente por el chip. El chip debe probar el conocimiento de su clave privada en un protocolo desafío-respuesta, en el que el sistema de inspección verifica la firma digital de un reto elegido aleatoriamente por el terminal. La Autenticación Activa se complementa con la comparación del MRZ visual con el MRZ en DG1, de esta forma se asegura que los datos son leídos del chip genuino, correspondiente al libretín genuino. Como se mencionó antes la AA requiere que el chip RF, tenga capacidad de procesamiento. En la AA se deben ejecutar las etapas siguientes (ICAO, 2015):

- El MRZ visual leído de la página de datos se compara con el MRZ del Grupo de Datos 1. El que se correspondan garantiza la autenticidad del MRZ visual, puesto que la autenticidad e integridad del DG1 ya fue verificada con PA. La autenticidad de la clave pública contenida en DG15 también ha sido ya probada en la PA.
- Para completar la AA se ejecuta un protocolo desafío-respuesta, usando el comando INTERNAL AUTHENTICATE, y solo después de que se ha establecido Mensaje Seguro. A continuación se detalla dicho protocolo:
  - El lector genera un random de 8 bytes RND.IFD y lo envía al chip con el comando APDU INTERNAL AUTHENTICATE.

- Los cálculos realizados por el chip requieren el uso de un algoritmo para firma, ya sea RSA o ECDSA. La explicación subsecuente considera el uso de RSA, con un tamaño de módulo k de 1024 bits (128 bytes) y como algoritmo de hash SHA-1.
  - Determina M2 a partir del APDU recibido, M2 = RND.IFD
  - Crea el indicador de fin:

    T = 'BC' (para SHA-1)

    t (longitud en octetos de T) = 1
  - Determina las longitudes c (capacidad de la firma) y L_{M1} (longitud del hash H):

    c = k – Lh – (8 x t) – 4 = 1024 – 160 – (8 x 1) – 4 = 852 bits

    L_{M1} = c – 4 = 848 bits
  - Genera un nonce M1 de longitud LM1 (106 bytes)
  - Crea M = M1 | M2
  - Calcula H = SHA-1(M)
  - Construir F = '6A' | M1 | H | T

    '6A' (partial recovery) significa que no se va a devolver la parte conocida RND.IFD
  - Cifrar F con la clave privada de AA, S = Enc(F)
  - Se envía S en el Response APDU
- El sistema de inspección debe verificar la firma de la siguiente forma:
  - Descifra S con la clave pública y obtiene F = Dec(S)
  - Determina el algoritmo de hash por el indicados de fin T = 'BC' → SHA-1
  - Extrae, de F, la condensación D (20 bytes anteriores a T)
  - Extrae, de F, M1 (106 bytes a partir del segundo byte de F)

- ‘6A’ indica recuperación parcial, pero la firma tiene la misma longitud del módulo k, entonces se debe concatenar M1 con M2 (M2 = RND.IFD)

  M* = M1 | M2

- Calcular D* = SHA-1(M*)
- Comparar D y D*. Si son iguales la verificación fue exitosa.

La *Figura 12* muestra los distintos elementos involucrados y procura explicar cómo gracias a la Autenticación Activa el MRZ visual (impreso en la página de datos del libretín) se corresponde a EF.DG1, cuya integridad/autenticidad así coma la de EF.DG15 fueron validadas cuando se verificó EF.SOD durante la Autenticación Pasiva. La clave pública en EF.DG15 a su vez se corresponde con la calve Privada en la memoria segura, es decir DG15 y chip se corresponden. Por lo tanto, existe una correspondencia entre libretín, chip RF y el contenido del chip (grupos de datos y SOD).

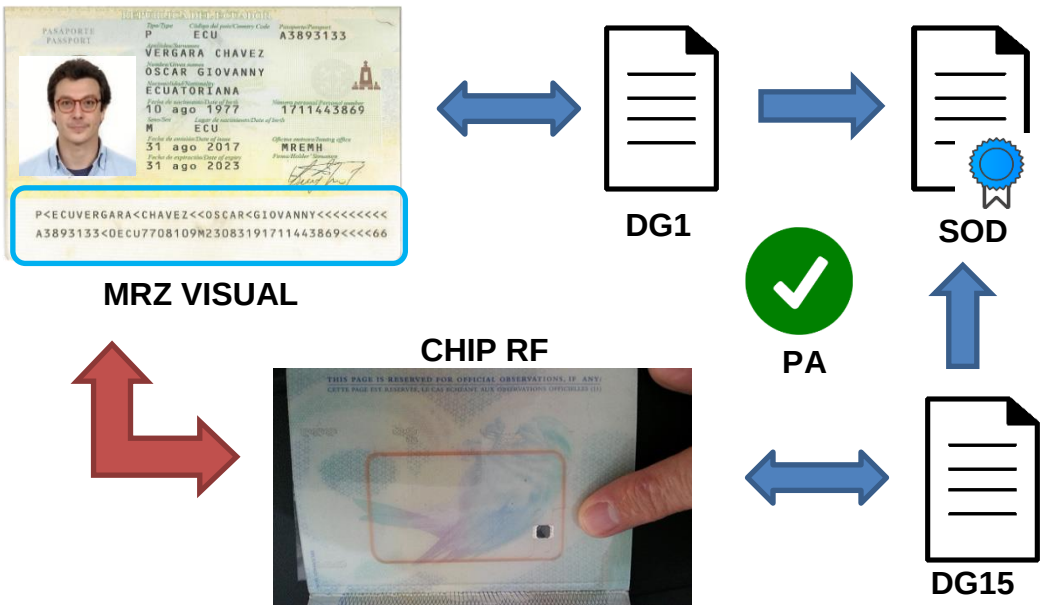

Figura 12: Elementos involucrados en la Autenticación Activa

4.3. Algoritmos usados en la firma digital de eMRTDs

El proceso de cálculo de firma digital se compone de dos etapas: primero se calcula una condensación (hash) y luego se la cifra, por tal motivo una firma digital requiere el uso combinado de dos algoritmos uno de hash y otro de cifrado o firma. En este apartado se hace mención de las opciones disponibles, para ambos tipos de algoritmos, en una PKI adecuada para emisión de documentos electrónicos. También se hace una comparación, de dichos algoritmos, basada en los aspectos relevantes para la emisión y verificación de pasaportes electrónicos.

4.3.1. Algoritmos de Firma Digital

Como se mencionó en la sección 2.5.2 cuando se habló de la PKI para pasaportes electrónicos, se permiten tres algoritmos para firma: RSA, DSA y ECDSA. La premisa fundamental en la que se basa cada uno de estos algoritmos se menciona a continuación, los algoritmos en sí, así como sus fundamentos matemáticos no se describen aquí, ya que no son objeto del presente trabajo.

a) RSA: Basa su seguridad en la dificultad de descomponer un número grande en el producto de dos primos (P*Q = N), “esto significa que dados suficientes recursos computacionales y tiempo, un adversario no debería ser capaz de romper RSA (obtener la clave privada) por factorización, (…) actualmente no han sido propuestos otros métodos para romper RSA eficientemente” (Jansma, 2004).
b) DSA: El Algoritmo de Firma Digital – DSA fue específicamente diseñado para firma digital, y es una variante del esquema de firma ELGamal. Su seguridad recae en la intratabilidad del problema del logaritmo discreto.

c) ECDSA: Es la versión de ECC (criptografía de curva elíptica) del algoritmo DSA, su seguridad se basa en la dificultad de calcular logaritmos discretos en el grupo de puntos de una curva elíptica definida sobre un campo finito (BSI, 2018).

4.3.1.1. Comparación de los algoritmos de firma digital

En el contexto de la firma digital de documentos de viaje electrónicos hay algunas consideraciones que se deben tomar en cuenta a la hora de identificar las características más apropiadas de un algoritmo de firma. Estas consideraciones son por ejemplo el tamaño de almacenamiento del CI sin contacto (particularmente importante durante el proceso de personalización del documento), el tiempo requerido para firmar el objeto SOD durante la personalización del chip RF, o el tiempo requerido para verificar la firma en la Autenticación Pasiva, durante el proceso de inspección. A continuación se hace una comparación de los algoritmos RSA, DSA y ECDSA en base a estas características, para lo cual es importante aclarar que cuando se trata de comparar algoritmos entre sí, la comparación se la debe hacer con tamaños de clave dados (para cada algoritmo) que proporcionen una fortaleza o nivel de seguridad semejante, al respecto (Barker, 2016) explica que la fuerza de la seguridad de un algoritmo (para un tamaño de clave dado) se puede describir en términos de la cantidad de esfuerzo que toma probar todas las claves posibles de un algoritmo simétrico de fuerza de seguridad comparable, es decir si un algoritmo A con una clave de Y-bits, tiene una fuerza estimada comparable a la de un algoritmo simétrico B con una clave de X-bits, entonces se dice que el algoritmo A puede proveer X bits de seguridad. En base a lo dicho, la **Tabla 24** muestra la fuerza en bits (columna 1) de los algoritmos de firma RSA, DSA y ECDSA, así como el respectivo algoritmo de clave simétrica al que se pueden comparar, (columna 2) dependiendo del tamaño del módulo o clave. En la

columna 3 k también es el tamaño del módulo n. En la columna 4 L es el tamaño de la clave pública y N el de la clave privada. En la columna 5 f también es el tamaño del orden del punto base G. 2TDEA se refiere a Triple Data Encryption Algorithm con 2 claves idénticas y una diferente, y 3TDEA se refiere a DEA con las 3 claves diferentes. Las combinaciones de algoritmo/tamaño de clave de la primera fía (aquellas que se han estimado con una fortaleza de seguridad máxima de menos de 112 bits) ya no están aprobadas para ser usadas en aplicaciones de protección criptográfica por el Gobierno Federal de los Estado Unidos.

Tabla 24

Fortaleza de la seguridad de los algoritmos de firma

Fuerza de la seguridad en bits	Algoritmos de clave simétrica	RSA (criptografía de factorización de enteros)	DSA (criptografía de campo-finito)	ECDSA (criptografía de curva elíptica)
≤ 80	2TDEA	k = 1024	L = 1024 N = 160	f = 160-223
112	3TDEA	k = 2048	L = 2048 N = 224	f = 224-255
128	AES-128	k = 3072	L = 3072 N = 256	f = 256-383
192	AES-192	k = 7680	L = 7680 N = 384	f = 384-511
256	AES-256	k = 15360	L = 15360 N = 512	f = 512+

Fuente: (Barker, 2016)

El espacio ocupado por la firma es importante tomando en cuenta la capacidad de almacenamiento limitada de los chips RF (en el caso Ecuatoriano aproximadamente 80 Kbytes), en este sentido el trabajo de (Endrodi, 2002) muestra que al comparar RSA con algoritmos de

curva elíptica, estos últimos producen un tamaño de firma menor, para tamaños de clave que proporcionan idéntico nivel de seguridad. Por ejemplo para claves de 1024 bits RSA y de 160 bits ECC, el tamaño de la firma ECC es tres veces menor que el de la firma RSA.

En cuanto al tiempo tomado en el proceso de firma, según el trabajo de (Jansma, 2004), para tamaños de clave que ofrecen equivalente nivel de seguridad, a medida que aumenta el tamaño de clave, ECC tiende a superar en desempeño a RSA. Es decir que el algoritmo de mejor desempeño en la generación de firma depende totalmente del tamaño de clave elegido. (Jansma, 2004) También miden el desempeño en la verificación de la firma, resultando que en todos los casos RSA es superior a ECDSA. Al respecto en la **Tabla 25** se puede ver una comparación del desempeño de RSA versus el de ECDSA para dos diferentes tamaños de clave. Como (Kessler, 2016) tambien afirma: "Algunos investigadores han encontrado que ECDSA es más rápido que RSA para el proceso de firma y descifrado, sin embargo ECDSA es un poco más lento para la verificación de la firma y el cifrado".

Tabla 25

Comparación del desempeño entre RSA y ECDSA para firma y verificación.

	RSA		**ECDSA**	
Tamaño de clave	1024 bits	2240 bits	163 bits	233 bits
Firma	0.01 seg.	0.15 seg.	0.15 seg.	0.34 seg.
Verificación	0.01 seg.	0.01 seg.	0.23 seg.	0.51 seg.

Fuente: (Jansma, 2004)

Si por otro lado se compara RSA con DSA, según manifiesta (Sivaraman, 2017), DSA es más rápido para el proceso de firma y descifrado, mientras que RSA es más rápido para el proceso de verificación de la firma y cifrado. Una ventaja importante que tiene ECDSA es que con un tamaño de clave mucho más pequeño puede ofrecer el mismo nivel de seguridad que los

otros algoritmos, tal como lo afirman (Jurišic, 1997), con una tamaño de clave de 160 bits, ECDSA ofrece el mismo nivel de seguridad criptográfica que DSA o RSA con una clave de 1024 bits. Según ambos autores eso “resulta en parámetros de sistema más pequeños, certificados de clave pública más pequeños,… implementación más rápida, menor requerimientos de poder, y procesadores de hardware más pequeños”.

Al topar un tema fundamenta como lo es la seguridad, entendida como la dificultad para resolver el problema matemático en el cual se basa el algoritmo de firma, resulta ser que ECDSA tiene ventaja sobre RSA y DSA.

> La razón principal del atractivo de ECDSA es el hecho de que no existe un algoritmo sub exponencial conocido para resolver el problema del logaritmo discreto de curva elíptica sobre una curva elíptica adecuadamente elegida. Por lo tanto se requiere tiempo full exponencial para resolverlo, mientras que el mejor algoritmo conocido para resolver la factorización de enteros para RSA y el problema de problema del logaritmo discreto en DSA toman ambos tiempos sub exponenciales. (Khalique, 2010).

La **Tabla 26** contiene un resumen comparativo de las cuatro características tratadas para los tres algoritmos de firma, de ella se puede desprender que ECDSA reúne la mayoría de las características deseables (color verde) para firma-verificación de pasaportes electrónicos, sin embargo no es la mejor opción en cuanto a rapidez en el despacho de viajeros en los controles fronterizos, ya que la verificación de la firma toma más tiempo que RSA.

Tabla 26

Comparación entre los algoritmos de firma RSA, DSA y ECDSA

Característica	RSA	DSA	ECDSA
Rapidez en la generación de la firma	Más lento que DSA y ECDSA	Más rápido que RSA	Más rápido que RSA (con claves grandes)
Rapidez en la verificación de la firma	Más rápido que DSA y ECDSA	Más lento que RSA	Más lento que RSA
Espacio requerido para la firma	Mayor que ECDSA	Mayor que RSA	Menor que RSA
Seguridad (dificultad para resolverlo)	Sub exponencial	Sub exponencial	Full exponencial

4.3.2. Algoritmos para sumas de verificación hash

Una función hash toma una entrada de longitud arbitraria (aunque limitada) y produce un valor de longitud fija, y su diseño busca ser resistente a la pre-imagen (que no sea posible encontrar un mensaje que produzca un valor hash dado) y resistente a la colisión (que no sea posible encontrar dos mensajes que produzcan el mismo valor hash) (Barker, 2016). Como se mencionó en la sección 2.5.2, las opciones de funciones hash que tiene una PKI para emisión de documentos son: SHA-224, SHA-256, SHA-384 y SHA-512.

4.3.2.1. Consideraciones para el algoritmo de hash

La fortaleza de la seguridad debe constituirse una característica fundamental al elegir un algoritmo de hash para firma, al respecto (Barker, 2016) nos proporciona un ordenamiento de los algoritmos para hash, en base a la fortaleza máxima estimada de seguridad medida en bits. Dicha categorización puede verse en la **Tabla 27**, donde se puede apreciar que la función de hash SHA-512 resulta ser la de mayor fortaleza. De manera similar a lo ocurrido con los algoritmos de

firma, las funciones hash con una fortaleza menor a 112 bits ya no están aprobadas para ser usadas en aplicaciones de protección criptográfica por el Gobierno Federal de los Estado Unidos.

Tabla 27

Fortaleza máxima estimada de la seguridad de las funciones hash

Fuerza de la seguridad en bits	Algoritmos de clave simétrica
≤ 80	SHA-1
112	SHA-224
128	SHA-256
192	SHA-384
256	SHA-512

Fuente: (Barker, 2016)

En cuanto la velocidad en el cálculo, la diferencia resulta ser despreciable, así lo demuestra el experimento realizado por (Latinov, 2018), quién compara la velocidad de generación del hash entre SHA-256 y SHA-512, para cadenas de caracteres de diferente longitud, y por ejemplo para una cadena de longitud de 72 caracteres obtiene una diferencia promedio de 50 milisegundos. Si la velocidad de cálculo resulta ser despreciable, la única característica a tomar en cuenta es la seguridad brindada por el algoritmo, en cuyo caso SHA-512 tiene ventaja sobre las demás opciones.

5 DESARROLLO DEL PRODUCTO

La ICAO mediante su Doc 9303 proporciona una serie de requisitos que se deben cumplir durante la validación de la autenticidad de los datos almacenados en el chip RF del documento. La verificación de la autenticidad de los datos se hace a través de la Autenticación Pasiva (PA) y la originalidad del chip se verifica con la Autenticación Activa (AA). Por otro lado, la confidencialidad de los datos del titular contenidos en el chip, está protegida con un control de acceso. Acorde al alcance del presente trabajo, el método de control de acceso que se empleará es aquel con el carácter de obligatorio, para sistemas de inspección, por parte de la ICAO, es decir el BAC (Control de Acceso Básico). ICAO recomienda que si un ePassport soporta tanto BAC como PACE, de preferencia se use PACE por ser más seguro que BAC, sin embargo actualmente no se tiene acceso a pasaportes biométricos ecuatorianos con soporte para PACE.

En este capítulo se plantea y describe una metodología para la verificación de la autenticidad de los datos contenidos en el chip RF. Inicialmente se presenta el diagrama de flujo de la metodología y posteriormente se realiza una descripción de los pasos que componen el proceso.

5.1 Pre-requisitos

El proceso de validación de ePassports con la metodología planteada necesita el cumplimiento de tres pre-requisitos básicos:

1. Tener acceso a los certificados de CCSCA, de DS y a las CRLs. Las formas en las que un estado receptor puede tener accedo a esta información, para alimentar sus sistemas de inspección, ya fueron tratadas en la sección 2.5.2 y son tres: por intercambio bilateral, descargada desde el directorio de clave pública (PKD) de ICAO o por medio de listas

maestras. Una forma de proporcionar el certificado de DS, a los países receptores, es incluirlo en el documento dentro de EF.SOD. Adicionalmente se debe establecer confianza en el CSCA, es decir que la credibilidad del proceso de validación, con la metodología propuesta, recae en la confianza que se tenga de que el certificado de CSCA es el auténtico emitido por el estado cuyos pasaportes se van a validar.

2. Tenar acceso a un dispositivo lector de chips RF, compatible con PC/SC, ISO/IEC 14443 e ISO/IEC 7816-4. En el Capítulo 3 se pudo observar que las diferentes normas o especificaciones utilizan diferente nomenclatura para referirse a el: Proximity Coupling Device (PCD) en ISO/IEC 14443, Interface Device (IFD) en PC/SC.
3. Tener acceso a pasaportes electrónicos personalizados, es decir con datos del titular grabados en el chip e impresos en la página de datos del documentos. En condiciones normales un proceso de inspección solo puede iniciar si el portador del documento entrega su pasaporte voluntariamente a un oficial fronterizo. Con esta acción se asume que el portador el documento autoriza al sistema de inspección a acceder a sus datos personales almacenados en el chip RF.

5.2 Diagrama de flujo de la metodología

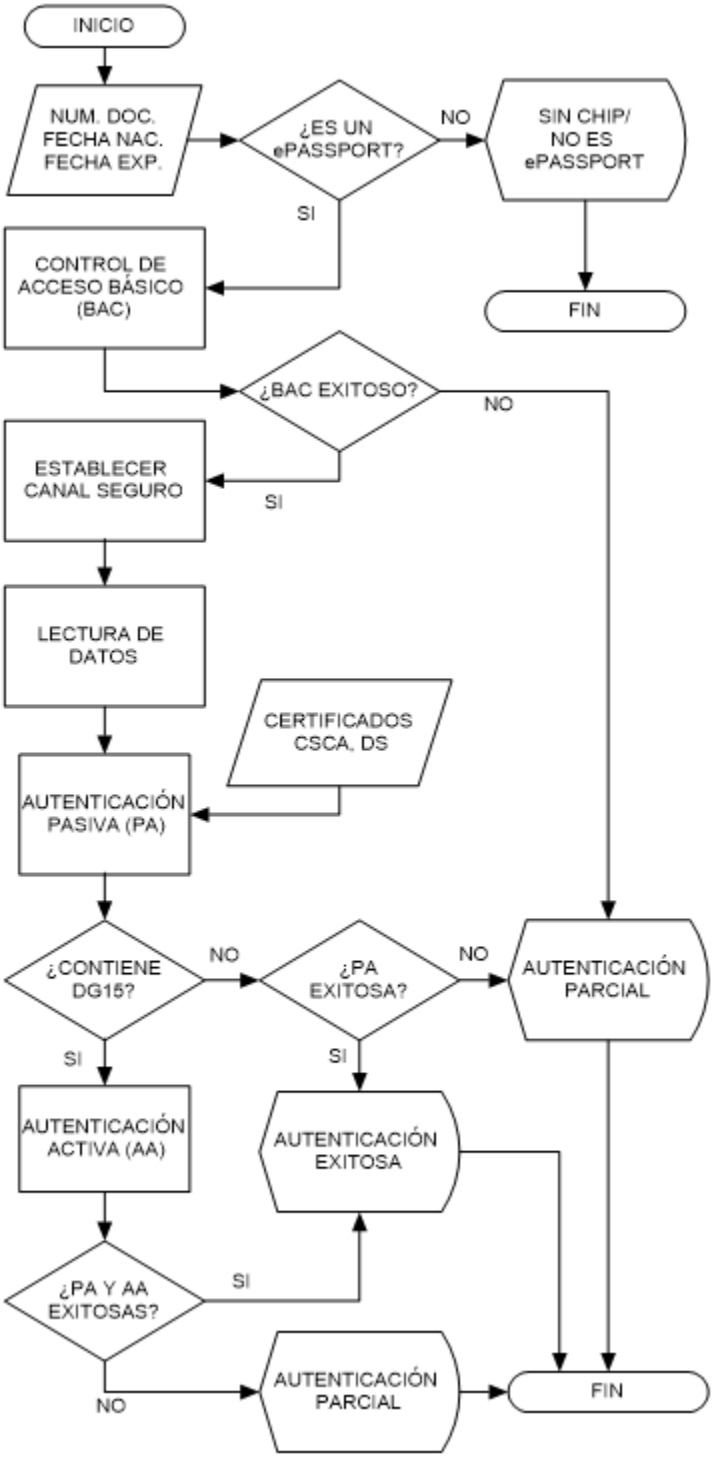

Figura 13: Diagrama de flujo de la metodología

5.3 Descripción de la metodología propuesta

La metodología planteada resume los resultados finales que se pueden dar, ante la validación de un ePassport, en tres posibles respuestas:

1. Autenticación Exitosa.
2. Sin chip/No es ePassport.
3. Autenticación Parcial.

El que una aplicación de sistema de inspección pueda ofrecer al menos estas tres respuestas, es considerada buena práctica por ICAO (ICAO, 2018). El significado de cada una de las respuestas se indica más adelante en la sección 5.3.7 cuando se hable de la interpretación de los resultados. Para que el proceso inicie se requiere como entrada la información contenida en el MRZ, más específicamente: el número de documento, la fecha de nacimiento y la fecha de expiración. Esta información solo se puede obtener de un libretín abierto, lo que significa que el portador entrego el documento al oficial fronterizo, hecho que se interpreta como el consentimiento para acceder al contenido del chip.

A continuación se realiza una descripción de los pasos involucrados en la metodología. En los puntos en los que se requiere interactuar con el chip RF se indica el command APDU que debe usarse, así como el response APDU esperado, dichos comandos ya fueron tratados en la sección 3.2.2.

5.3.1 Comprobación de chip contenido

Cuando un portador presenta su documento para el proceso de validación, lo primero que se debería hacer es verificar que efectivamente se trata de un pasaporte electrónico. Si el

dispositivo lector no detecta un chip RF, se asume que no existe uno o que este está dañado y el proceso termina mostrando el mensaje SIN CHIP/NO ES PASAPORTE ELECTRÓNICO.

5.3.2 Control de Acceso Básico (BAC)

Antes de llevar a cabo el protocolo BAC, el primer comando APDU que debe ejecutarse es el comando SELECT. Con este comando se selecciona la aplicación eMRTD, cuyo identificador de aplicación AID es: 0xA0000002471001 (sección 2.3). Solo cuando se ha seleccionada la aplicación eMRTD se puede acceder a los ficheros contenidos en ella (EF.COM, EF.SOD, EF.DG1….EF.DG16). La respuesta esperada a este comando debe ser '90 00', cualquier otra respuesta significa que el documento no está listo para que se pueda llevar a cabo un proceso de inspección. La Tabla 28 muestra el comando SELECT y su respuesta.

Tabla 28

Command y Response APDU para seleccionar la aplicación ICAO

Command						
CLA	**INS**	**P1**	**P2**	**Lc**	**Data**	**Le**
00	A4	04	0C	07	A0000002471001	-
Response						
Datos					**SW1-SW2**	
-					90 00	

Fuente: (ICAO, 2015)

Se ejecutan los tres procedimientos que conforman el control de acceso BAC descritos en la sección 4.2.1:

- El lector deriva las claves de acceso basadas en los datos del MRZ
- Chip y lector se autentifican mutuamente, usando los comandos GET CHALLENGE y EXTERNAL AUTHENTICATE
    - Comando GET CHALLENGE y su respuesta, ver **Tabla 29**.

Tabla 29

Command y Response APDU para GET CHALLENGE, para pedir un reto al chip.

Command				
CLA	**INS**	**P1**	**P2**	**Le**
00	84	00	00	08
Response				
Datos				**SW1-SW2**
RND.IC				90 00

Fuente: (ICAO, 2015)

- Comando EXTERNAL AUTHENTICATE y su respuesta, Tabla 30.

Tabla 30

Command y Response APDU para EXTERNAL AUTHENTICATE.

Command						
CLA	INS	P1	P2	Lc	Data	Le
00	82	00	00	28	cmd_data	28
Response						
Datos						**SW1-SW2**
resp_data						90 00

Fuente: (ICAO, 2015)

- El lector calcula las claves de sesión

Una respuesta diferente a ’90 00’ tanto para el comando GET CHALLENGE como para el comando EXTERNAL AUTHENTICATE significa un fallo en el protocolo BAC. La respuesta ’90 00’ indica que BAC se ejecutó exitosamente y que en adelante cada mensaje que se envíe al chip RF debe estar construido con mensaje seguro (SM).

5.3.3 Establecimiento del canal seguro

Todos los mensajes que se intercambien con el ePassport, a partir de la ejecución exitosa de BAC, deben estar protegidos por Mensajes Seguros haciendo uso de las claves de sesión calculadas previamente. La construcción de Mensajes Seguros se explica en la sección 4.1.2. A través del canal seguro establecido se debe leer del chip RF los grupos de datos cuyo contenido es considerado menos sensible: DG1, DG2, DG7, etc., así como el objeto de seguridad del documento SOD.

5.3.4 Lectura de la información

Los grupos de datos presentes, en el chip RF, se pueden conocer al leer el EF.COM (Common Data Elements), de tal manera que el primer archivo elemental que se debe leer debe ser este. Primero se debe seleccionar EF.COM usando el comando SELECT y después se puede leerlo usando el comando READ BINARY. Para facilitar la explicación, los comandos mostrados en están sección no están protegidos con SM.

- Comando SELECT con el identificador del fichero '01 0E' para EF.COM en el campo datos (comando plano, sin protección SM), ver **Tabla 31**.

Tabla 31

Command y Response APDU para seleccionar cada fichero que se va a leer.

Command					
CLA	**INS**	**P1**	**P2**	**Lc**	**Data**
00	A4	02	0C	02	01 1E
Response					
Datos					**SW1-SW2**
-					90 00

Fuente: (ICAO, 2015)

- Comando READ BINARY para leer los primeros 4 bytes del contenido del archivo seleccionado (comando plano, sin protección SM), ver **Tabla 32**.

Tabla 32

Command y Response APDU para READ BINARY, para leer un fichero.

Command				
CLA	**INS**	**P1**	**P2**	**Le**
00	B0	00	00	04
Response				
Datos			**SW1-SW2**	
4 primeros bytes de EF.COM			90 00	

Fuente: (ICAO, 2015)

Luego de leer EF.COM (y conocer los grupos de datos presentes) se puede continuar con la lectura de cada grupo de datos almacenado en el chip RF, de una manera muy similar a como se realizó la lectura de EF.COM, es decir seleccionando primero el fichero elemental usando el identificador de fichero FID, para a continuación leer su contenido con READ BINARY. La

Tabla 33 muestra el identificador de archivo FID asignado a cada DG, así como el rótulo que debe tener dentro de EF.COM.

Tabla 33

Identificador de fichero FID y rótulo usado en EF.COM, para cada DG.

Nombre del EF	FID	Rótulo dentro de EF.COM
EF.COM	011E	60
EF.DG1	0101	61
EF.DG2	0102	75
EF.DG3	0103	63
EF.DG4	0104	76
EF.DG5	0105	65
EF.DG6	0106	66
EF.DG7	0107	67
EF.DG8	0108	68
EF.DG9	0109	69
EF.DG10	010A	6ª
EF.DG11	010B	6B
EF.DG12	010C	6C
EF.DG13	010D	6D
EF.DG14	010E	6E
EF.DG15	010F	6F
EF.DG16	0110	70
EF.SOD	011D	77
EF.CARDACCESS	011C	
EF.CardSecurity	011D	

Fuente: (ICAO, 2015)

Es fundamental, para el proceso de validación, que también que también de lea del chip RF el fichero EF.SOD, cuyo rótulo (77) no necesariamente se encontrará en EF.COM. EF.SOD debe seleccionarse con el FID ‘011D’, indicado en la **Tabla 33**.

5.3.5 Autenticación Pasiva (PA)

Una vez que se ha leído EF.SOD, este se debe decodificar, tomando en cuenta que se trata de una estructura de tipo SignedData, codificada usando la regla de codificación distinguida DER y que debe estar en perfil ASN.1. Una vez decodificado EF.SOD, se debe obtener de él lo siguiente:

- El algoritmo usado para el cálculo de las condensaciones de cada DG ‘hashAlgorithm’, obtenido de la secuencia ‘LDSSecurityObject’.

- El ‘encapsulatedContent’ que son los datos sobre los que se calculó la firma y el ‘Signature’ que es la firma en sí.
- El algoritmo usado para calcular la firma digital de ‘encapsulatedContent’, compuesto por ‘digestAlgorithm’ más ‘signatureAlgorithm’.
- Además se debe extraer el certificado de firmante de documentos (DS). Según el Doc 9303 (ICAO, 2015), el incluir el certificado de firmante de documento en EF.SOD es opcional para los estados, pero si no se lo incluye se lo puede proporcionar a los estados receptores mediante uno de los mecanismos de distribución mencionadas en la sección 2.5.2.
- La lista de los valores hash calculados sobre cada grupo de datos, que en el perfil ASN.1 de EF.SOD es una secuencia llamada ‘dataGroupHashValues’, en la que cada entrada de la secuencia está formada por un entero (identificador del grupo de datos) más un OCTET STRING (valor hash correspondiente).

Al proceso de Autenticación Pasiva descrito en 4.2.3, se lo puede resumir en tres sub etapas:

5.3.5.1 Verificación de la firma de EF.SOD

Para verificar la firma de EF.SOD se requiere como entrada el certificado de DS, el ‘encapsulatedContent’ y ‘Signature’. Se debe verificar que el resultado de la firma realizada sobre ‘encapsulatedContent’, usando la clave privada del firmante de documentos DS da como resultado el valor ‘Signature’, por lo tanto también se necesita conocer la combinación de los algoritmos ‘digestAlgorithm’ y ‘signatureAlgorithm’ que se usaron para firmar el ‘encapsulatedContent’ de EF.SOD.

5.3.5.2 Verificación de la ruta de certificación del certificado DS

Este paso requiere tener acceso a los certificados DS y CSCA y a la lista de revocación CRL. Para esta verificación puede usarse el algoritmo PKIX descrito en RFC 5280, con algunas consideraciones, tal como se indicó en la sección 4.2.3.1.

5.3.5.3 Verificación de la integridad de los datos

Para esta operación se requiere conocer el algoritmo usado para calcular las condensaciones de cada grupo de datos. Para cada grupo de datos contenido en el chip RF se debe calcular su condensación usando el algoritmo 'hashAlgorithm' y se debe comparar con el valor correspondiente contenido en la lista de valores hash 'dataGroupHashValues'. Si ambos valores son iguales se puede afirmar que el grupo de datos no ha sido modificado.

Un resultado negativo en alguno de los tres pasos anteriores significa un fallo en la Autentificación Pasiva.

5.3.6 Autenticación Activa (AA)

Si el chip contiene el Grupo de Datos 15, debe realizarse la Autenticación Activa como se explica en la sección 4.2.4, caso contrario se la omite. La AA se realiza mediante el comando INTERNAL AUTHENTICATE, ver **Tabla 34**.

Tabla 34

Command y Response APDU para INTERNAL AUTHENTICATE.

Command						
CLA	INS	P1	P2	Lc	Data	Le
	88	00	00	08	RND.IFD	-
Response						
Datos						SW1-SW2
Firma generada por el CI						90 00

Fuente: (ICAO, 2015)

Una respuesta diferente a ’9000’ al comando INTERNAL AUTHENTICATE significa un fallo en la AA. Sólo si la verificación de la firma es exitosa, la AA se considera exitosa.

5.3.7 Interpretación de los resultados y su presentación

En vista de que durante la validación se realizan múltiples comprobaciones, los resultados del proceso deben sintetizarse en tres posibles respuestas, con lo que se facilita la interpretación al oficial de control fronterizo. Las posibles respuestas son:

5.3.7.1 Autenticación Exitosa

Se debe mostrar cuando el control de acceso, la Autenticación Pasiva y la Autenticación Activa fueron exitosos, entonces se considera que todo el proceso de autenticación fue exitoso.

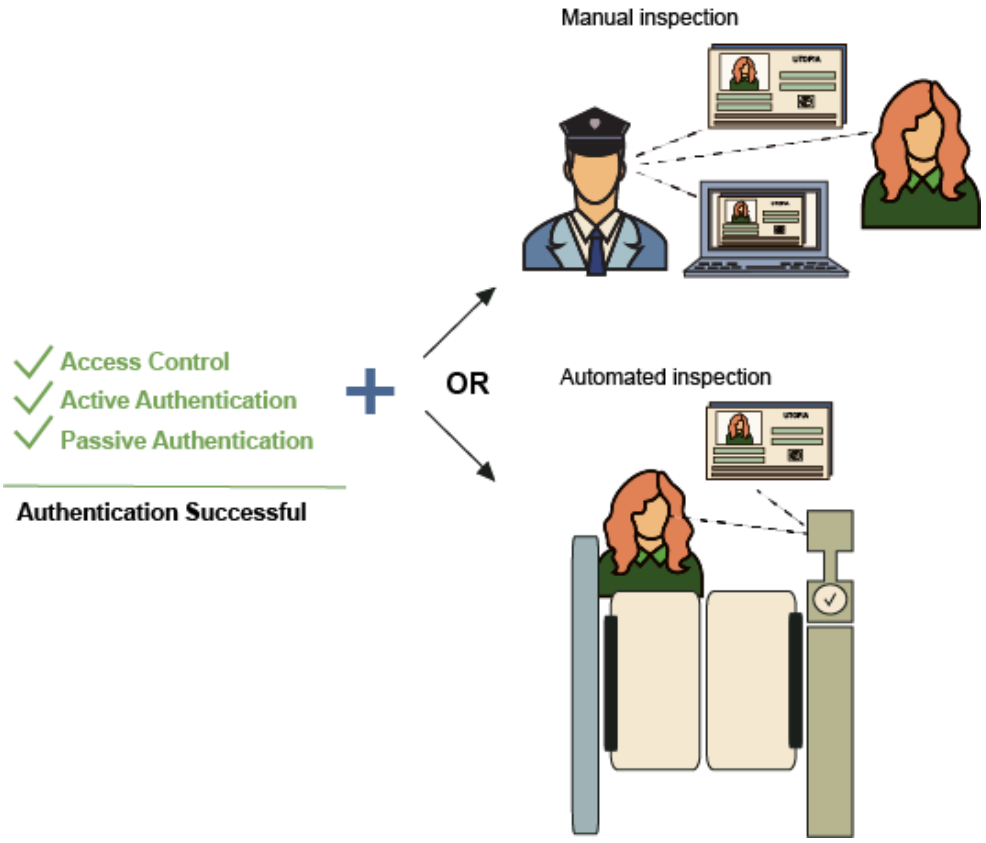

Figura 14: Resultado: “AUTENTICACIÓN EXITOSA” del Sistema de Inspección
Fuente: (ICAO, 2018)

Acción que debería realizar el oficial fronterizo: para completar la autenticación debe realizarse una comparación manual de los datos contenidos en el chip, contra los datos impresos en el documento (ICAO, 2018). La *Figura 14* esquematiza un resultado de AUTENTICACIÓN EXITOSA.

5.3.7.2 Sin chip / No es pasaporte electrónico

Se debe mostrar cuando el pasaporte no responde al lector, debido a que no se trata de un pasaporte electrónico o el chip-antena tienen un problema que no permite su lectura.

Acción que debería realizar el oficial fronterizo: verificar si el documento tiene el símbolo de micro plaqueta contenida (**Figura 1**), y en el caso afirmativo se debe buscar señales de que el documento fue manipulado para dañar el chip y si es necesario remitir al viajero a una inspección secundaria, en el caso negativo el oficial fronterizo debe realizar una inspección manual, es decir, como si se tratase de un pasaporte no electrónico (ICAO, 2018). La *Figura 15* esquematiza un resultado SIN CHIP/NO ES EPASSPORT.

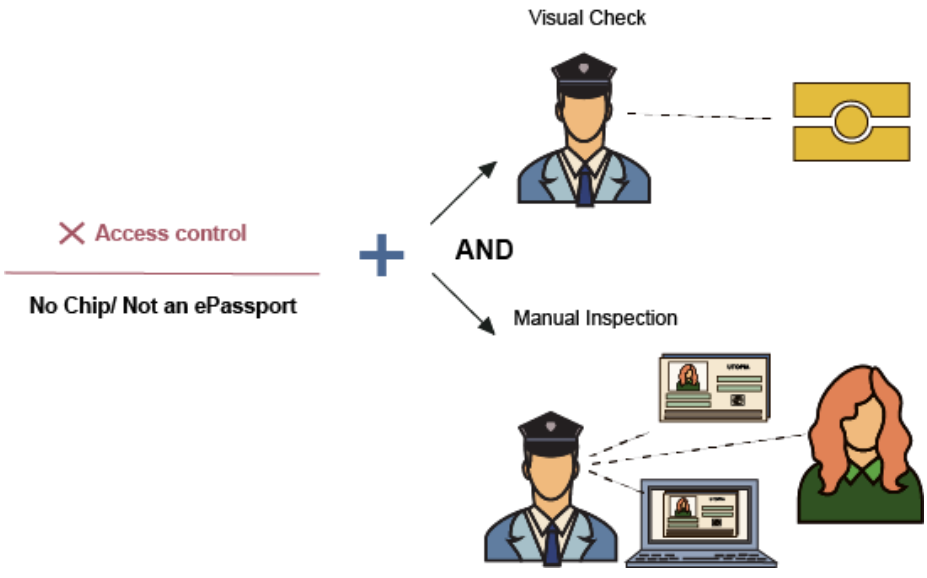

Figura 15: Resultado: "SIN CHIP/NO ES ePASSPORT" del Sistema de inspección
Fuente: (ICAO, 2018)

5.3.7.3 Autenticación Parcial

Este resultado se debe mostrar cuando alguna de las fases no fue satisfactoria en el proceso de validación del chip. Resulta útil para el oficial, encargado de realizar la inspección, conocer detalles del ¿por qué? de una autenticación parcial, por lo que, junto al resultado final se debería mostrar información adicional que brinde detalles sobre la etapa fallida. Una autenticación parcial se puede producir por cualquiera de los siguientes factores:

- Control de acceso fallido
- Autenticación Pasiva fallida
- Autenticación Activa fallida

Este resultado incluye aquellos casos en los que el contenido del chip no pudo ser leído como producto de un fallo en la ejecución de BAC. En cuyo caso no se pueden llevar a cabo las Autenticaciones Pasiva y Activa. Estamos ante un caso en el cual la validación del contenido de un Pasaporte Electrónico no se pudo llevar a cabo, sin embargo el documento no se puede considerar aún inválido, como se manifiesta en el Doc 9303 parte 11 (ICAO, 2015):

> En el caso de que los datos del CI sin contacto no puedan utilizarse, por ejemplo, como resultado de una revocación de certificados o de una verificación de firma inválida, o si el CI sin contacto fue dejado intencionalmente en blanco, el eMRTD no es necesariamente invalidado. En tales casos un Estado receptor PODRÍA basarse en otras características de seguridad del documento para fines de validación.

Acción que debería realizar el oficial fronterizo: Se deben seguir los procedimientos operativos correspondientes establecidos por el estado receptor, como podría ser una inspección secundaria más detallada (ICAO, 2018). La *Figura 16* esquematiza un resultado de AUTENTICACIÓN PARCIAL.

Figura 16: Resultado: "AUTENTICACIÓN PARCIAL" del Sistema de inspección
Fuente: (ICAO, 2018)

5.4 Herramienta para validar ePassports (ePassportValidator)

La herramienta que se va a desarrollar se basara en los procedimientos definidos en la metodología propuesta y constituirá un instrumento que permita automatizar su aplicación.

5.4.1 Implementación de la Herramienta ePassportValidator

Una consideración importante que se debe tomar en cuenta durante la implementación de la metodología es el hardware con el que se cuenta para efectuar la lectura de los pasaportes electrónicos. Para el presente trabajo se cuenta con lector RFID genérico de Smart Card Contactless, y no posee la capacidad de lectura óptica de texto OCR. Por tal motivo la información contenida en el MRZ, necesaria para el acceso al chip mediante BAC, no podrá obtenerse automáticamente mediante un escaneo de la página de datos del documento y por lo tanto deberá ser ingresada manualmente por quién esté usando la aplicación. Esto implica que la aplicación debe ofrecer la funcionalidad necesaria para ingresar: el número de documento, la fecha de nacimiento y la fecha de expiración.

El diseño y desarrollo de la herramienta no se basará en metodologías dirigidas por la experiencia de usuario como es el caso de la metodología de Diseño Centrado en el Usuario, UCD por sus siglas en ingles. Para poder usar la metodología UCD se requiere conocer al usuario

al cual estará destinado el desarrollo y además tener contacto con él. Por otro lado, la herramienta que aquí se desarrolla, tiene como objetivo automatizar la aplicación de una metodología planteada (para validar pasaportes) a fin de poder probarla, por lo tanto, no está destinada a usuarios específicos, sino que su ámbito de uso se limita al propósito investigativo del presente proyecto. Además, hasta el término de este trabajo, Ecuador no emite aún pasaportes electrónicos y por lo tanto actualmente su disponibilidad es muy limitada, lo que reduce en gran medida los posibles escenarios en los que se pueda usar dicha herramienta. De todas maneras, tratando de asegurar la utilidad de la herramienta, en el apartado 5.4.3, se hace un breve análisis de su usabilidad, usando para ello usuarios sin características particulares, es decir, que no pertenecen a un grupo particular dentro del ámbito de uso de la aplicación.

La herramienta a desarrollar, va a estar compuesta por una funcionalidad básica (automatización de la metodología de validación de ePassports), sobre la que se pueden hacer incrementos que aporten el resto de la funcionalidad requerida (ej. interfaz gráfica, manejo de logs, etc.). En ese sentido una metodología de desarrollo de software que se adapta bien a estas necesidades es SCRUM, la misma que requiere como documentación mínima: el Product Backlog y los Sprint Backlogs.

5.4.1.1 Product Backlog

Contiene los elementos de la lista priorizada del producto llamados Product Backlog Items, que son las historias de usuario. La

Tabla 35 muestra el Product Backlog, que contiene seis historias de usuario que van a guiar el desarrollo de la herramienta de validación de pasaporte biométricos.

Tabla 35

Historias de Usuario que reúnen los requisitos para el desarrollo de la herramienta.

ID	Historia de Usuario	Descripción	Criterios Aceptación	Prioridad
HU1	COMO usuario de la herramienta QUIERO ver el resultado de la validación de un pasaporte, reducido a 3 posibles resultados PARA que así se facilite la comprensión del proceso	Mostrar el resultado de la aplicación de la metodología, ya sea: Autenticación exitosa, Sin chip/No es ePassport o Autenticación Parcial	Al terminar la validación de un pasaporte, se debe mostrar el resultado final del proceso, ya sea AUTENTICACIÓN EXITOSA, SIN CHIP/NO ES EPASSPORT o AUTENTICACIÓN PARCIAL	2
HU2	COMO usuario de la herramienta QUIERO ver el registro de los pasos que forman la metodología PARA conocer el paso que se ejecuta en determinado instante y su resultado	Log durante el proceso de aplicación de la metodología, donde se indique el paso que se está llevando a cabo en cada momento y su resultado	Ya sea durante un proceso de validación o al final de este, se debe mostrar un registro de los pasos efectuados y su estado	3
HU3	COMO usuario de la herramienta QUIERO validar el contenido de un ePassport en base a la metodología propuesta PARA conocer el resultado de la validación	La aplicación constituye en sí la automatización de la aplicación de la metodología	Si un pasaporte con determinada condición (ej. falsificado, clonado, etc.) se presenta a la aplicación, esta, luego de la validación, debería mostrar un resultado acorde a la condición del pasaporte evaluado (ej. Autenticación Parcial: Fallo en AA)	1
HU4	COMO usuario de la herramienta QUIERO ingresar los datos: número de documento, fecha de nacimiento y fecha de expiración PARA poder acceder al contenido del chip	Ingreso de la información mínima requerida para el acceso (BAC) al chip	Debe existir un manera de que el usuario ingrese los 3 datos requeridos para el acceso BAC	4

CONTINÚA

HU5	COMO usuario de la herramienta QUIERO indicar la ruta en la que se encuentran: el certificado CSCA y la CRL PARA que se pueda efectuar la Autenticación Pasiva	Por defecto la aplicación debe tener acceso al CSCAc y la CRL (de prueba) pero también debe brindar la opción de usar otro CSCA y CRL, requeridos para PA	La aplicación debe brindar la opción de escoger otra CSCA o CRL, diferente a las de Prueba	6
HU6	COMO usuario de la herramienta QUIERO que se desplieguen los datos contenidos en el chip (si se pudieron leer) PARA poder compararlos con la impresión en el documento	Si el usuario puede ver los datos contenidos en el chip, entonces podrá comparar el contenido del chip con la impresión en el documento, que es una medida adicional de seguridad, que prueba que chip y documento físico se corresponden	Cuando se pueda acceder y leer el contenido del chip, los Grupos de Datos contenidos en él deben mostrarse en pantalla	5

5.4.1.2 Sprint Backlog

Constituye la lista de tareas pendientes de realizar en cada Sprint o incremento y conforman la funcionalidad que se entregará en el próximo incremento. La historia de usuario HU3 es la que presenta mayor complejidad, teniendo en cuenta representa en sí el proceso de validación, por lo que se la ha descompuesto en tareas más sencillas de implementar, cada una de las cuales constituye una entrada de un Sprint Backlog.

Se proveen 4 Sprints para la implementación de la herramienta, el Sprint Backlog correspondiente a casa uno de ellos se muestra a continuación.

Sprint 1: duración 10 días

Tabla 36

Sprint 1

ID	Tarea	HU	Responsable	Horas
I1	Conexión con el documento (PC/SC), selección de la Aplicación ICAO y ejecución de BAC	HU3	JLP	16h
I2	Lectura de los Grupos de Datos contenidos del chip	HU3	JLP	16h
I3	AA	HU3	JLP	16h
I4	Control de flujo del proceso	HU3	JLP	16h

Sprint 2: duración 10 días

Tabla 37

Sprint 2

ID	Tarea	HU	Responsable	Horas
II1	PA: verificación de la firma de EF.SOD	HU3	JLP	16h
II2	PA: verificación de la integridad de cada DG.	HU3	JLP	16h
II3	PA: validación de la ruta de certificación del certificado DSc con el algoritmo PKIX	HU3	JLP	16h
II4	PA: Verificación de la revocación del certificado DSc	HU3	JLP	16h

Sprint 3: duración 10 días

Tabla 38

Sprint 3

ID	Tarea	HU	Responsable	Horas
II1	Mostar uno de los 3 resultados posible al final del proceso, en caso de AUTENTICACIÓN PARCIAL indicar el proceso en el cual falló	HU1	JLP	16h
II2	Implementación del log de la aplicación, para que además muestre el tiempo tomado en cada sub-proceso	HU2	JLP	16h
II3	Control de excepciones provocadas cuando se intenta leer Grupos de Datos protegidos por EAC	HU3	JLP	16h
II4	Control del flujo, con documentos que no contienen chip	HU3	JLP	16h

Sprint 4: duración 10 días

Tabla 39

Sprint 4

ID	Tarea	HU	Responsable	Horas
II1	Control de excepciones ante un fallo en alguno de los procesos: BAC, PA, o AA	HU3	JLP	16h
II2	Permitir el ingreso de los datos para el acceso BAC y validación de la entrada	HU4	JLP	16h
II3	Permitir la elección tanto del CSCAc como de la CRL a usar en la PA	HU5	JLP	16h
II4	Mostar los datos leídos del chip (ej. Foto, firma)	HU6	JLP	16h

5.4.2 Visión General de la Herramienta ePassportValidator

La herramienta desarrollada permite realizar la validación del contenido de un pasaporte electrónico en base a los lineamientos proporcionados por la metodología propuesta. Fue desarrollada en el lenguaje JAVA versión de JDK 1.8, y consta de 3 paquetes, mostrados en la *Figura 17*:

a) com.epassport: contiene sólo una clase llamada LoggerInit.java que es la encargada de inicializar las configuraciones requeridas por el registro log de la herramienta.

b) com.epassport.gui: contiene dos clases:

- ✓ ApplicationMainView.java: es la interfaz gráfica principal de la aplicación, que además contiene los controles gráficos necesarios para las entradas del usuario, *Figura 18*.
- ✓ DGViewer.java: interfaz gráfica que muestra el resultado de la validación y la información que se puede leerse de un ePassport. La primera pestaña, llamada Resultados muestra el log del proceso de validación que se acaba de ejecutar, esta pestaña se muestra en todos los casos, aun cuando BAC falle o no se trate de un

ePassport, véase *Figura 19*. Si se puedo leer el contenido del chip, esta interfaz contendrá una pestaña con el contenido de cada Grupo de Datos leído del chip, véase *Figura 20*, *Figura 21* y *Figura 22*.

c) com.epassport.validation: aloja el núcleo de la funcionalidad de la herramienta. Consta de 4 clases:

- ✓ EFMetaInfo.java: representa los metadatos de un archivo elemental (EF), nombre del archivo, número de grupo de datos e identificador de archivo FID.
- ✓ PasiveAuthentication.java: proporciona métodos para realizar las diferentes tareas de las que consta la Autenticación Pasiva. La *Figura 23* es un extracto de la clase PasiveAuthentication.java, que muestra el algoritmo PKIX para la validación de la ruta de certificación del certificado DS_C.
- ✓ ActiveAuthentication.java: implementa la Autenticación Activa, usando SHA-1 con RSA.
- ✓ InspectionTerminal.java: proporciona el punto de acceso al proceso de validación de pasaportes, a través del método inspectionProcess(). Implementa la metodología en sí.

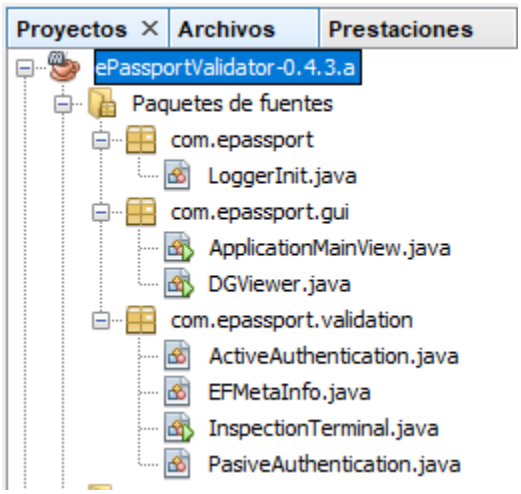

Figura 17: Estructura del proyecto Java ePassportValidator

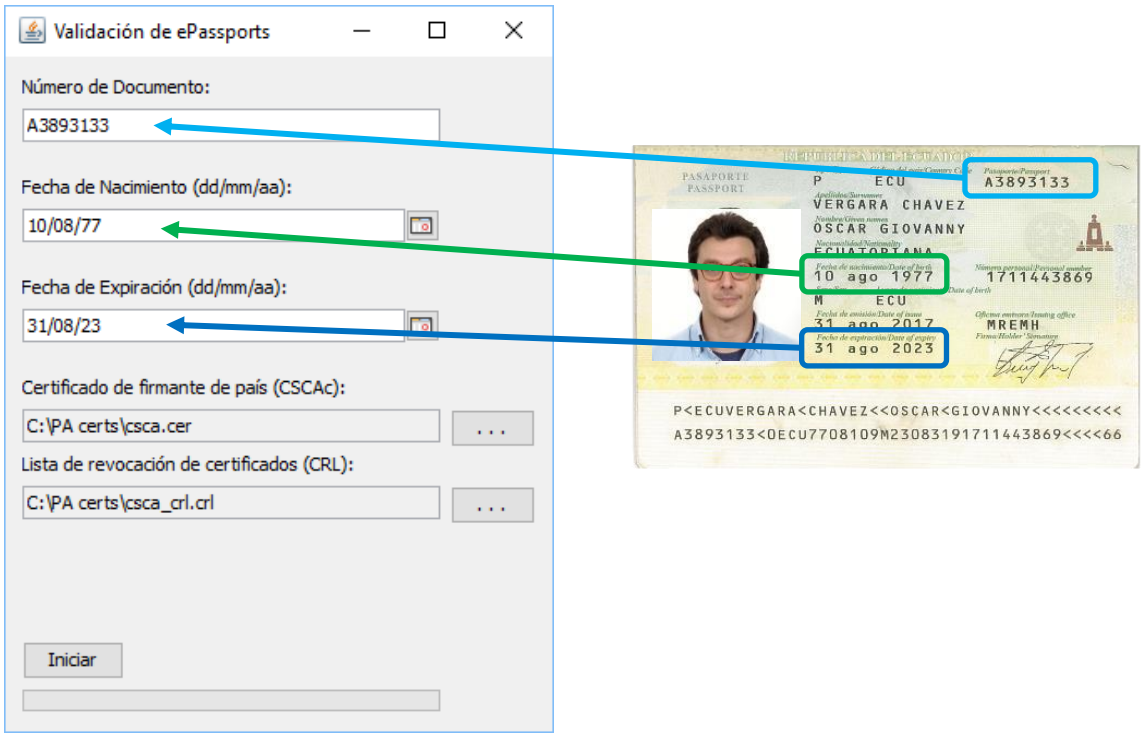

Figura 18: Interfaz principal donde se ingresan los datos para acceso BAC

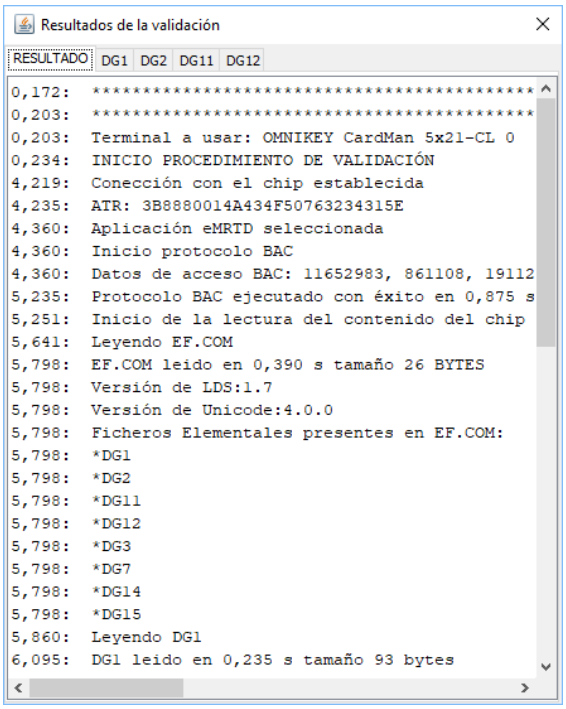

Figura 19: Interfaz DGViewer mostrando el log de la validación

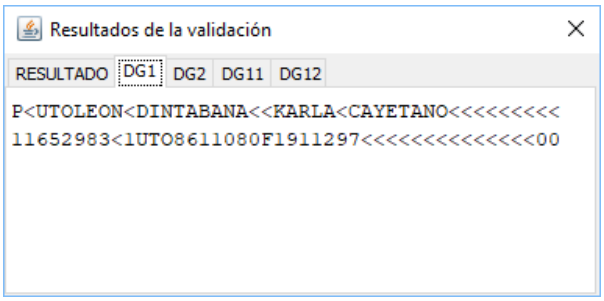

Figura 20: Interfaz DGViewer mostrando el contenido del Grupo de Datos 1 (MRZ)

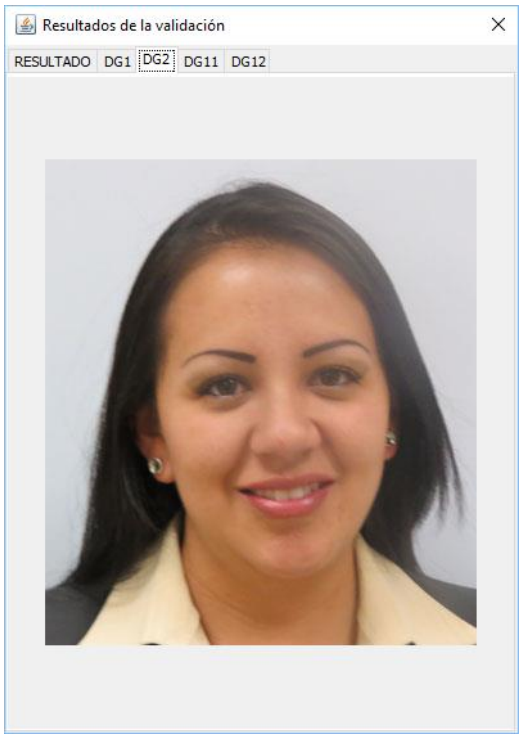

Figura 21: Interfaz DGViewer mostrando el contenido del Grupo de Datos 2 (rostro)

Figura 22: Interfaz DGViewer mostrando el contenido del Grupo de Datos 7 (firma)

```
public void validateDSCertificationPath() throws CertificateException, InvalidAlgorithmE
    loadCRL();
    CertificateFactory cf = CertificateFactory.getInstance("X.509");
    //set the list of path certs, exclude root CA
    CertPath certPath = cf.generateCertPath(Arrays.asList(dsCert));
    //the trust anchor
    TrustAnchor trustAnchor = new TrustAnchor(cscaCert, null);
    //params for RFC 5280 algorithm
    PKIXParameters params = new PKIXParameters(Collections.singleton(trustAnchor));
    //to specify subject name constraints DSc issuer = CSCAc subject
    X509CertSelector targetConstraints = new X509CertSelector();
    targetConstraints.setIssuer(cscaCert.getSubjectX500Principal());
    params.setTargetCertConstraints(targetConstraints);
    //Doc9303: "initial-policy-mapping-inhibit: Set to inhibit policy mapping;"
    params.setPolicyMappingInhibited(true);
    //Doc9303: "initial-explicit-policy: This should NOT be set"
    params.setExplicitPolicyRequired(false);
    //Doc9303: "initial-any-policy-inhibit: Set to inhibit processing of the any-policy
    params.setAnyPolicyInhibited(true);
    //Doc9303: "user-initial-policy-set: Set to the special value "any-policy"
    params.setInitialPolicies(null);
    //Revocation check implementing in another method
    params.setRevocationEnabled(false);

    //the CertPathValidator for do validation operation
    CertPathValidator cpv = CertPathValidator.getInstance("PKIX");
    try {
        PKIXCertPathValidatorResult result
                = (PKIXCertPathValidatorResult) cpv.validate(certPath, params);
```

Figura 23: Algoritmo PKIX para validación de la ruta de certificación

La herramienta ha sido probada únicamente en sistemas operativos Windows, y funciona con cualquier dispositivo lector contactless estándar -ISO 14443 compatible con la especificación PC/SC, ya que hace uso de la API de javax.smartcardio, que en el caso de Windows trabaja sobre la API WinSCard, en el caso de Linux debería trabajar sobre la librería PC/SC –Lite.

Para el acceso y la decodificación de la información en el chip se ha usado la API open source JMRTD, que proporciona la funcionalidad requerida para BAC y Mensaje Seguro (SM). La ***Figura 24*** muestra los componentes involucrados en la arquitectura de ePassportValidator.

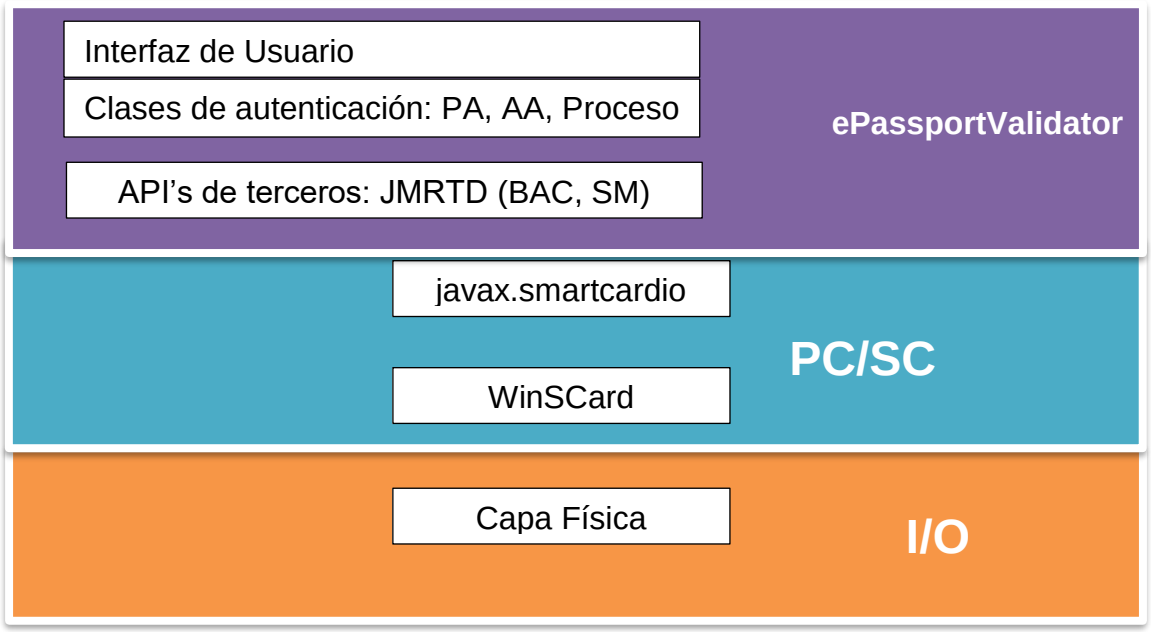

Figura 24: Componentes de la herramienta ePassportValidator

5.4.3 Usabilidad de la Herramienta

La norma (ISO 9241, 1997) define la usabilidad como: "la capacidad que tiene un producto para ser usado por determinados usuarios con el fin de alcanzar unos objetivos concretos con efectividad, eficiencia y satisfacción dentro de un contexto de uso específico". En donde la efectividad se refiere a capacidad del sistema de ofrecer la funcionalidad para la que fue creado, la eficiencia se refiere al esfuerzo necesario para poder llevar a cabo dicha funcionalidad, y la satisfacción se refiere a la sensación que tiene el usuario que usa la herramienta (Marco et al., 2006). Con el fin de garantizar que la herramienta cumple el objetivo para el cual fue creada, se debe analizar su usabilidad, para lo cual se diseñó un cuestionario, con preguntas que proporcionen información cualitativa relativa a la utilidad, facilidad, rapidez, efectividad y satisfacción en el uso de la herramienta. Posteriormente se eligieron aleatoriamente 10 usuarios, que no estaban familiarizados con los conceptos de pasaporte electrónico, y se les explico en forma breve el objetivo de la herramienta. Sin indicarles cómo operar la herramienta se les permitió usarla por alrededor de 5 minutos, al final de los cuales se les aplico el cuestionario mostrado en la Tabla 40.

Tabla 40

Cuestionario para evaluar la usabilidad de la herramienta desarrollada

Cuestionario para medir su percepción acerca del uso de la herramienta de validación de pasaportes electrónicos. Por favor evalué cada pregunta en una escala de 1 a 5, en donde 1 significa insuficiente y 5 excelente.					
Pregunta	**1 Insuficiente**	**2 Regular**	**3 Bueno**	**4 MuyBueno**	**5 Excelente**
¿Cómo califica la rapidez ofrecida, por la herramienta, para llevar a cabo su propósito?					
¿Cómo califica la facilidad del uso de la herramienta?					
¿Cómo califica el grado de satisfacción, que obtuvo, del uso de la herramienta?					
¿En qué medida es efectiva la herramienta, es decir, que cumple el objetivo para el cual fue creada?					
¿Qué parte de la herramienta considera que se podría mejorar, para que su uso sea más satisfactorio para usted?					

Los resultados de aplicar el cuestionario han mostrado que los usuarios están conformes con la rapidez, facilidad de uso y efectividad de la herramienta (*Figura 25*), sin embargo el grado de satisfacción derivado de su uso es bajo. La respuesta, de los usuarios, a la última pregunta

proporciona información del elemento que produce esta baja satisfacción, que el 70% de las veces, resultó ser la pantalla que muestra el resultado final del proceso.

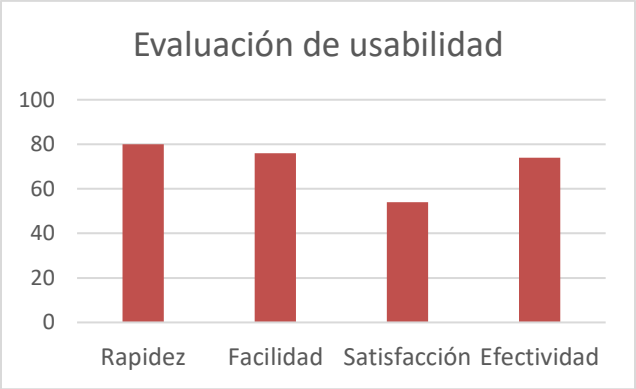

***Figura 25*:** Resultados de la encuesta de usabilidad

Con el fin de mejorar la usabilidad de la herramienta se realizaron modificaciones a la pantalla de resultados, de tal manera que la información que proporciona sea más fácil de entender. Se agregaron iconos, se simplifico la información que muestra y se añadió un botón que muestra el registro log, en caso de que se desee ver información más detallada, resultando en la captura mostrada en la ***Figura 26***, recuérdese que el aspecto original de esta pantalla se muestra en la *Figura 19*.

Figura 26: Pantalla de resultados mejorada

5.5 Aplicación de la metodología

5.5.1 Preparativos

El proceso de verificar la autenticidad de un pasaporte ecuatoriano requiere el acceso a los certificados (CSCA y DS) y CRL de país, sin embargo el Ecuador actualmente no emite aún pasaportes electrónicos y tampoco cuenta con una PKI para emisión de documentos de viaje. Para poder probar la metodología propuesta y la herramienta desarrollada ha sido necesaria la creación de certificados provisionales, que para los fines pertinentes remplacen a los del Estado Ecuatoriano, así como la personalización (datos del ciudadano grabados en el chip, e impresos en el documento) de varios pasaportes para las pruebas. Los siguientes preparativos fueron necesarios:

- Se han creado un certificado de firmante de país CSCA (de prueba) para el ecuador, así como un certificado de firmante de documentos DSc (firmado por el certificado CSCA anterior), y una lista de revocación CRL (también firmada por el certificado CSCA anterior).
- Usando el certificado de firmante de documentos DSc creado se personalizaron varios pasaportes.

Para las pruebas que se realicen se establece el punto de confianza en el certificado CSCA creado, y su clave pública, de tal manera que es necesario asumir que dicho certificado es el certificado auténtico de firmante de país del Ecuador.

5.5.2 Prueba de la Metodología

El propósito de la prueba es validar la metodología propuesta, es decir que frente a un documento con unas condiciones iniciales, la aplicación de la metodología proporcione la

respuesta correcta. Por ejemplo si se presenta, para su verificación, un documento cuyo objeto de seguridad del documento SOD ha sido firmado por un certificado DSc que a su vez no está firmado por la clave privada de la CSCA en la cual se estableció el punto confianza, la aplicación de la metodología debería arrojar como resultado que la autenticación no fue satisfactoria en su totalidad (AUTENTICACIÓN PARCIAL) debido a un fallo en la Autenticación Pasiva. En tal virtud se debe definir un caso de prueba tanto para el flujo normal del proceso (Autenticación exitosa), como para cada uno de los flujos alternos que se pueden presentar en la metodología, es decir un total de seis casos de prueba, los mismos que se muestran en la **Tabla 41**. Nótese que en esta tabla se omiten los casos en los que el control de acceso falle, pero la Autenticación Pasiva y/o Activa se desarrollen correctamente, esto es debido a que un fallo en el control de acceso implica el no acceso al contenido del chip.

Tabla 41

Los 6 casos de prueba que cubren las diferentes opciones que pueden presentarse

Caso No.	Condición	Resultado esperado
1	Documento sin chip o chip dañado.	No hay interacción en el chip: "SIN CHIP/NO ES PASAPORTE ELECTRÓNICO"
2	Existe un error en los datos para acceso BAC, o estos no se proporcionan.	BAC debe fallar y no se debería poder acceder al contenido del chip: "AUTENTIFICACIÓN PARCIAL: BAC->FALLÓ, NO SE EFECTÚO PA NI AA"
3	Alguno/s de: CSCAc, DSc o CRL, no está/n disponible/s o no es/son el/los que corresponde/n.	BAC correcto, PA debe fallar, AA correcta: "AUTENTIFICACIÓN PARCIAL: BAC->OK, PA->FALLÓ, AA->OK"

CONTINÚA

4	**Documento con datos de AA inconsistentes (la clave pública en DG15 no se corresponde a la clave privada en la memoria segura del chip).**	**BAC correcto, PA correcta, AA debe fallar: "AUTENTIFICACIÓN PARCIAL: BAC->OK, PA->OK, AA->FALLÓ"**
5	Alguno de: CSCAc, DSc o CRL no está disponible o no es el que corresponde, además hay una inconsistencia entre la clave pública (DG15) y la privada de AA.	BAC correcto, PA y AA deben fallar: "AUTENTIFICACIÓN PARCIAL: BAC->OK, PA-> FALLÓ, AA->FALLÓ"
6	No se requiere condición (proceso normal)	Todos los procesos se ejecutan satisfactoriamente: "AUTENTIFICACIÓN EXITOSA"

Para poder realizar las pruebas previstas se requieren pasaportes personalizados, de tal manera que cumplan la condición requerida para cada caso de prueba, es decir:

- Para el Caso 1 se necesita un pasaporte que no sea electrónico o no contenga chip RF.
- Para el Caso 2 se requiere un pasaporte en el que el MRZ visual no coincida con el MRZ en DG1, de esta forma BAC fallará.
- Para el Caso 3 se requiere un pasaporte ecuatoriano cuyo objeto de datos SOD haya sido firmado por un DS que no es el auténtico.
- Para el Caso 4 se requiere un pasaporte que tenga grabada una clave pública en DG15, que no sea la correspondiente a la clave privada grabada en la memoria segura del chip RF.
- Para el Caso 5 se requiere un pasaporte que cumpla las condiciones de los cosas 3 y 4 al mismo tiempo.
- Para el Caso 6 se requiere un pasaporte que haya sido correctamente personalizado, y que por lo tanto sea validado exitosamente.

Los pasaportes necesarios (proporcionados por el IGM) han sido personalizados de acuerdo a las condiciones mencionadas, usando para ello una aplicación de personalización también proporcionada por el IGM.

5.5.3 Resultados de aplicar la metodología

Los resultados de la aplicación de la metodología, mediante el uso de la herramienta de validación de pasaportes electrónicos, demuestran que está es útil para su cometido, presentando en todos los casos de prueba el resultado esperado. La **Tabla 42** muestra la conformidad de la evaluación de los seis casos de prueba, usando la herramienta ePassportValidator, por lo que se puede considerar a la metodología, y a su vez a la herramienta desarrollada, como válidas para la tarea de validar la autenticidad de un pasaporte electrónico.

Tabla 42

Resultados obtenidos en la evaluación de los 6 casos de prueba

Caso No.	Condición	Resultado esperado	Resultado obtenido	
1	Documento sin chip o chip dañado.	Si no hay interacción en el chip: “SIN CHIP/NO ES PASAPORTE ELECTRÓNICO”	“SIN CHIP/NO ES PASAPORTE ELECTRÓNICO”	
2	Existe un error en los datos para acceso BAC, o estos no se proporcionan	BAC debe fallar y no se debería poder acceder al contenido del chip: “AUTENTIFICACIÓN PARCIAL: BAC->FALLÓ, NO SE EFECTÚO PA NI AA”	“AUTENTIFICACIÓN PARCIAL: BAC->FALLÓ, NO SE EFECTÚO PA NI AA”	
3	Alguno/s de: CSCAc, DSc o CRL, no está/n disponible/s o no es/son el/los que corresponde/n	BAC correcto, PA debe fallar, AA correcta: "AUTENTIFICACIÓN PARCIAL: BAC->OK, PA->FALLÓ, AA->OK"	"AUTENTIFICACIÓN PARCIAL: BAC->OK, PA->FALLÓ, AA->OK"	

CONTINÚA 

4	**Documento con datos de AA inconsistentes (la clave pública en DG15 no se corresponde a la clave pública en la memoria segura del chip)**	**BAC correcto, PA correcta, AA debe fallar: "AUTENTIFICACIÓN PARCIAL: BAC->OK, PA->OK, AA->FALLÓ"**	**"AUTENTIFICACIÓN PARCIAL: BAC->OK, PA->OK, AA->FALLÓ"**	
5	Alguno de: CSCAc, DSc o CRL no está disponible o no es el que corresponde, además hay una inconsistencia entre la clave pública (DG15) y la privada de AA.	BAC correcto, PA y AA deben fallar: "AUTENTIFICACIÓN PARCIAL: BAC->OK, PA-> FALLÓ, AA->FALLÓ"	"AUTENTIFICACIÓN PARCIAL: BAC->OK, PA-> FALLÓ, AA->FALLÓ"	
6	No se requiere condición (proceso normal)	Todos los procesos se ejecutan satisfactoriamente: "AUTENTIFICACIÓN EXITOSA"	"AUTENTIFICACIÓN EXITOSA"	

6 CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

- La inversión que el Estado Ecuatoriano realice para implementar la emisión de Pasaportes Electrónicos se justifica por los beneficios que estos aportan, ya que añaden una capa adicional a las seguridades físicas que ya tienen los pasaportes no electrónicos. Esta seguridad adicional está respaldada por una Infraestructura de Clave Pública especializada en la emisión de pasaportes biométricos. Otro beneficio que aporta en los sistemas de control fronterizo, es la automatización, que puede ofrecer, tanto de la validación de la autenticidad del documento, como de la identificación del portador del documento, esto último gracias a los datos biométricos que es capaz de contener: imagen facial y opcionalmente huellas dactilares e iris. Esta automatización ofrecida acarrea consigo una subsecuente agilidad en el proceso migratorio. Un ejemplo es la herramienta ePassportValidator, desarrollada en el capítulo 5, y capaz de llevar a cabo el proceso de validación de un ePassport en aproximadamente 10 segundos. Por último la emisión del Pasaporte Biométrico es uno de los requisitos de la Unión europea para la exención del visado Schengen, lo que promueve la libre movilidad de los ecuatorianos.
- Aun cuando ciertos mecanismos de seguridad avanzados, como la Autenticación Activa o la Autenticación del Chip, son opcionales para ICAO, su implementación va ayudar a incrementar la seguridad del documento de viaje nacional, haciéndolo aún más resistente contra intentos de falsificación o adulteración de identidad.
- La decisión acerca de si el pasaporte electrónico Ecuatoriano debe o no debe implementar el Control de Acceso Extendido (EAC), depende en gran medida de los acuerdos políticos

con otros estados, como es el caso del convenio con la Unión Europea para la exención del visado Schengen. Si el acuerdo impone que para el arribo al territorio europeo, la identidad de los ciudadanos ecuatorianos deba ser validada empleando la biometría de huellas dactilares, en ese caso será mandatorio el soporte de EAC para proteger la confidencialidad del Grupo de Datos 3. Sin embargo EAC es un complejo mecanismos de seguridad que también requiere el uso de una PKI adicional, por lo que su implementación conllevaría mayor tiempo y coste elevados.

- La metodología planteada es válida para la verificación electrónica de pasaportes biométricos, de tal manera que se puede confiar en los resultados de su aplicación. Sin embargo hay que tomar en cuenta que la metodología se basa en tres resultados posibles, cuyo significado debe ser correctamente interpretado, para lo cual se ofrece una guía en la sección 5.3.7.
- La herramienta desarrollada, basada en los lineamientos de la metodología planteada, ha demostrado ser útil para la validación electrónica de pasaportes biométricos, esta afirmación es avalada por los resultados en las pruebas practicadas, en donde en todos los casos se obtuvo el resultado esperado.

6.2 Recomendaciones

- Antes de que el estado Ecuatoriano inicie la emisión de pasaportes biométricos, deben tomarse en cuenta aspectos de interés relacionados con la seguridad, como los son por ejemplo, los tamaños de claves y algoritmos para firma digital. En ese sentido se recomienda usar tamaños de claves cuya fortaleza en la seguridad sea mayor o igual a 112 bits, es decir claves mayores o iguales a 2048 bits para RSA o DSA y mayores o iguales a

224 bits para ECDSA, véase **Tabla 24**. Para los requerimientos actuales de seguridad, si se usan tamaños de claves apropiados, el algoritmo de firma que se decida usar (RSA, DSA o ECDSA) cobra menor relevancia, sin embargo ECDSA ofrece mayor seguridad con miras al futuro, aun cuando la verificación de la firma tarda poco más que RSA, pero este coste se ve más que recompensado con la mayor seguridad que brinda. En cuanto al algoritmo de hash, todos los disponibles (SHA-224, SHA-256, SHA-384 y SHA-512) tienen una fortaleza en la seguridad mayor o igual a 112 bits, por lo que se adaptan bien las condiciones de seguridad requeridas, de tal manera que se podría elegir cualquiera de ellos, ya que además las diferencias en la velocidad de cálculo son despreciables. SHA-512 sin embargo es el que más seguridad brinda.

- Se recomienda que el pasaporte biométrico ecuatoriano cuente con un mecanismo de autenticación del chip para evitar la clonación y garantizar que los datos se leen del chip original. Su implementación durante el proceso de emisión no demanda mayor complejidad ni costes adicionales (el sistema de emisión debe crear un par de claves, en forma segura, y almacenarlas en el chip) y los beneficios son evidentes. En la sección 2.5.1.3 se mencionan los tres mecanismos disponibles para autenticar el chip y en la sección 4.2.4 se explicó a detalle la Autenticación Activa.
- En cuanto al mecanismo de control de acceso, en esta trabajo se ha usado BAC por motivos de factibilidad, sin embargo se recomienda, de ser posible, el uso de PACE, que brinda una mejora sustancial con respecto al anterior.

REFERENCIAS

Alinor. (2010). *English Wikipedia*. Recuperado el 2 de Mayo de 2018, de https://en.wikipedia.org/wiki/File:Biometric_passports.png

Alonso, J. M. (2013). *Diseño e implementación de un lector PC/SC inalámbrico para tarjeta inteligente basado en plataformas móviles NFC.* Universidad de Cantabria, Santander, Cantabria, España.

Álvarez, O. E. (2013). Protocolos Anticolisión en RFID. *Telem@tica, 12*(1), 4-6.

Barker, E. (2016). *Recommendations for Key Management, Part 1: General (800-57 Pt1 Rev 4).* Obtenido de National Institute of Standards and Technology - NIST USA: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r4.pdf

Bender, J. K. (2009). Introducing the PACE solution. *Innovation, 26*, 1.

BSI. (2015). *TR-03110 - Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token.* Technical Guideline, German Federal Office for Information Security. Obtenido de https://www.bsi.bund.de/EN/Publications/TechnicalGuidelines/TR03110/BSITR03110.html?nn=6650344

BSI. (2018). *TR-03111 - Elliptic Curve Cryptography*. Technical Guideline, German Federal Office for Information Security. Obtenido de https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidelines/TR03111/BSI-TR-03111_pdf.html

Endrodi, C. &. (2002). Efficiency analysis and comparison of public key algorithms. *conference of PhD students in computer science.*

Guthery, S., & Jurgensen, T. (2002). *Smart Cards: The Developer's Toolkit.* Upper Saddle River, NJ, USA: Prentice Hall PTR.

ICAO. (2015). Doc 9303. *Parte 1: Introducción, Séptima edición*. Montreal, Quebec, Canada: ICAO.

ICAO. (2015). Doc 9303. *Parte 11: Mecanismos de seguridad para los MRTD, Séptima edición*. Montreal, Quebec, Canada: ICAO.

ICAO. (2018). *ePassport Validation Roadmap Tool*. Obtenido de https://www.icao.int/Security/FAL/PKD/BVRT/Pages/default.aspx

ISO 9241. (1997). *ISO 9241: Ergonomic Requirements for Office Work with Visual Display Terminals.* Géneve: International Organization for Standardization.

ISO/IEC 14443. (2008). *ISO/IEC 14443 Identification cards - Contactless integrated circuit card - Proximity cards.* Geneva: ISO.

ISO/IEC 7816. (2013). *ISO/IEC 7816-4 Identification cards - Integrated circuit cards: Organization, security and commands for interchange.* Geneva: ISO.

Jansma, N. &. (2004). Performance comparison of elliptic curve and rsa digital signatures. *nicj*.

Joynes, M. (2012). PKI Deployment & International Trust. En B. Kefauver (Ed.), *Eighth Symposium and Exhibition on ICAO MRTDS, Biometrics and Security Standards.* Montréal, Canada.

Jurišic, A. &. (1997). Elliptic curves and cryptograpy. *Dr. Dobb's Journal*, 26-36.

Kessler, G. C. (3 de Marzo de 2016). An Overview of Cryptography. *Updated Version*.

Khalique, A. S. (2010). Implementation of elliptic curve digital signature algorithm. *International journal of computer applications, 2*(2), 21-27.

Latinov, L. (3 de Mayo de 2018). *MD5, SHA-1, SHA-256 and SHA-512 speed performance*. Obtenido de https://automationrhapsody.com/md5-sha-1-sha-256-sha-512-speed-performance/

Martinez, A. (21 de febrero de 2018). Pasaporte biométrico en Ecuador: ¿Qué es y por qué adoptarlo? *Metro*. Obtenido de https://en.wikipedia.org/wiki/File:Biometric_passports.png

PC/SC Workgroup. (2013). Interoperability Specification for ICCs and Personal Computer Systems - Part 1: Introduction and Architecture Overview. *segunda edición*. Obtenido de https://www.pcscworkgroup.com/specifications/download/

Schalk, G. H., & Bienert, R. (2013). *RFID MIFARE and Contactless Cards in Application.* Stattegg, Austria: Elektor International Media BV.

Sivaraman, K. (2017). A comparison study of rsa and dsa algorithm in movile cloud computing. *International Journal of Pure and Applied Mathematics, 116*(8), 247-253.

Surós, A. C. (Agosto de 2013). Control de acceso extendido para pasaportes electrónicos. En R. Lorán (Ed.), *Innovation in Engineering, Technology and Education for Competitiveness and Prosperity.* Cancún, México.