OACI

Doc 9303

Documentos de viaje de lectura mecánica

Octava edición, 2021

Parte 12: Infraestructura de clave pública para los MRTD

Aprobado por la Secretaria General y publicado bajo su responsabilidad

ORGANIZACIÓN DE AVIACIÓN CIVIL INTERNACIONAL

OACI

Doc 9303

Documentos de viaje de lectura mecánica

Octava edición, 2021

Parte 12: Infraestructura de clave pública para los MRTD

Aprobado por la Secretaria General y publicado bajo su responsabilidad

ORGANIZACIÓN DE AVIACIÓN CIVIL INTERNACIONAL

Publicado por separado en español, árabe, chino, francés, inglés y ruso
por la ORGANIZACIÓN DE AVIACIÓN CIVIL INTERNACIONAL
999 Robert-Bourassa Boulevard, Montréal, Québec, Canada H3C 5H7

En el sitio web www.icao.int/security/mrtd pueden obtenerse descargas
e información adicional.

Documentos de viaje de lectura mecánica **(Doc 9303)**
Parte 12 — *Infraestructura de clave pública para los MRTD*
Núm. de pedido: 9303P12
ISBN 978-92-9265-540-2 (versión impresa)

ENMIENDAS

La publicación de enmiendas se anuncia en los suplementos del *Catálogo de productos y servicios*; el Catálogo y sus suplementos pueden consultarse en el sitio web de la OACI: www.icao.int. Las casillas en blanco facilitan la anotación de estas enmiendas.

REGISTRO DE ENMIENDAS Y CORRIGENDOS

ENMIENDAS		
Núm.	Fecha	Anotada por

CORRIGENDOS		
Núm.	Fecha	Anotado por

Las denominaciones empleadas en esta publicación y la forma en que aparecen presentados los datos que contiene no implican, de parte de la OACI, juicio alguno sobre la condición jurídica de ninguno de los países, territorios, ciudades o áreas, o de sus autoridades, ni respecto a la delimitación de sus fronteras o límites.

ÍNDICE

Página

1. ALCANCE

En la Parte 12 del Doc 9303 se define la infraestructura de clave pública (PKI) para la aplicación del eMRTD. Se especifican los requisitos para los Estados expedidores u organizaciones expedidoras, incluido el funcionamiento de una autoridad de certificación (CA) que expide certificados y listas de revocación de certificados (CRL). También se especifican los requisitos para los Estados receptores y sus sistemas de inspección que validan dichos certificados y CRL.

En la octava edición del Doc 9303 se incorporan las especificaciones para los sellos digitales visibles (conocidos como VDS) y para elementos opcionales como las aplicaciones de los registros de viaje, registros de visados y datos biométricos adicionales (conocidos como LDS2) como extensión de la aplicación del eMRTD obligatoria (conocida como LDS1).

El Doc 9303-12 se leerá conjuntamente con:

- Doc 9303-10 — *Estructura lógica de datos (LDS) para el almacenamiento de datos biométricos y de otro tipo en el circuito integrado (CI) sin contacto*;
- Doc 9303-11 — *Mecanismos de seguridad para los MRTD*; y
- Doc 9303-13 — *Sellos digitales visibles.*

2. RESEÑA DE LA INFRAESTRUCTURA DE CLAVE PÚBLICA

La infraestructura de clave pública (PKI) del eMRTD permite la creación y subsiguiente verificación de firmas digitales en objetos eMRTD, incluyendo el objeto de seguridad de documento (SO_D) para asegurar que los datos firmados son auténticos y no han sido modificados. La revocación de un certificado, la falla del procedimiento de validación de la ruta de certificación o la falla de la verificación de la firma digital, por sí solos, no hacen que el eMRTD se considere inválido. Dicha falla significa que la verificación electrónica de la integridad y autenticidad de los datos LDS ha fallado y que entonces podrían utilizarse otros mecanismos no electrónicos para efectuar esa determinación como parte de la inspección general del eMRTD.

La PKI del eMRTD es mucho más sencilla que las PKI más genéricas con muchas aplicaciones como la PKI de Internet definida en [RFC 5280]. En la PKI del eMRTD cada Estado expedidor o autoridad expedidora establece una única autoridad de certificación (CA) que expide todos los certificados directamente a entidades finales, incluyendo firmantes del documento. Estas CA se conocen como autoridades de certificación de firma de país (CSCA). No hay otras CA en la infraestructura. Los Estados receptores establecen la confianza directamente en las claves o certificados CSCA de cada Estado expedidor u organización expedidora.

La PKI del eMRTD se basa en normas PKI genéricas, incluyendo las [X.509] y [RFC 5280]. Estas normas básicas de PKI definen un gran conjunto de características opcionales y complejas relaciones de confianza entre las CA que no son pertinentes a la aplicación del eMRTD. En esta parte del Doc 9303 se especifica un perfil de dichas normas, adaptadas a la aplicación del eMRTD. Algunos de los aspectos singulares de la aplicación del eMRTD son los siguientes:

- hay exactamente una CSCA por Estado expedidor;
- las rutas de certificación comprenden exactamente un certificado (p. ej., firmante del documento);
- la verificación de la firma debe ser posible de 5 a 10 años después de la creación;

- se admite el cambio de nombre de la CSCA; y
- los certificados de enlace CSCA no se procesan como certificados intermedios en una ruta de certificación.

En su mayor parte, la infraestructura PKI del eMRTD cumple con [RFC 5280]. No obstante, el hecho de que las CSCA puedan experimentar un cambio de nombre impone requisitos singulares sobre la PKI del eMRTD que son incompatibles con algunos de los procedimientos de validación CRL definidos en [RFC 5280]. Estas diferencias se han mantenido en un mínimo y se identifican claramente.

En el caso del VDS y la LDS2, la PKI de la firma digital, que proporciona la autenticidad e integridad de los objetos de datos, es una extensión de la PKI de la LDS1. Los firmantes del VDS y la LDS2 son expedidos por la misma CSCA que expide los firmantes de la LDS1. Los cambios en los perfiles de certificado de estas nuevas aplicaciones se especifican en el presente documento. Esta infraestructura se conoce, en su conjunto, como la PKI del eMRTD.

La PKI de la firma digital comprende las entidades siguientes:

- CA de firma de país (CSCA);
- certificados de firmante del documento (CDS) que se utilizan para validar los objetos de seguridad de documento (SO_D);
- certificados de firmante LDS2, que abarca los elementos siguientes:
  - firmante LDS2-TS – firma los sellos de viaje LDS2;
  - firmante LDS2-V – firma los visados electrónicos LDS2; y
  - firmante LDS2-B – firma los datos biométricos adicionales LDS2;
- certificados de firmante del código de barras (BSC), para los que, en el presente documento, se definen los dos tipos específicos siguientes:
  - certificados de firmante del visado (VSC); y
  - certificados de firmante del documento de viaje de emergencia (ESC);
- certificados de firmante de la lista maestra (MSC) utilizados para firmar listas maestras;
- certificados de firmante de la lista de desviaciones (DLSC) utilizados para firmar listas de desviaciones;
- lista de revocación de certificados (CRL).

Los distintos tipos de certificados están firmados por la misma CSCA. La CSCA también firma la CRL, que contiene cualquier certificado revocado, independientemente del tipo de certificado. Todos los certificados expedidos por la CSCA se conocen en su conjunto como **certificados de firmante**.

En el caso de las aplicaciones LDS2, se define una **PKI de autorización** separada. La PKI de autorización permite que el Estado expedidor o la organización expedidora del eMRTD controle y gestione los Estados extranjeros a los que se concede autorización para escribir objetos de datos LDS2 en sus eMRTD y leer dichos objetos de datos. Un Estado extranjero que prevea leer o escribir datos LDS2 debe obtener un certificado de autorización directamente del Estado expedidor u organización expedidora del eMRTD.

La PKI de autorización utiliza una estructura de certificado diferente (certificado verificable mediante tarjeta ISO 7816) y, por lo tanto, requiere componentes de infraestructura adicionales.

La LDS2 requiere que el terminal demuestre al CI sin contacto del eMRTD que tiene derecho a escribir objetos de datos LDS2 en el CI sin contacto o que tiene derecho a leer objetos de datos LDS2. Dicho terminal está equipado con al menos una clave privada y el correspondiente certificado de terminal, que codifica la clave pública del terminal y los derechos de acceso. Después de que el terminal haya demostrado conocer esta clave privada, el chip del MRTD concede al terminal el acceso para leer/escribir datos LDS2, según lo indicado en el certificado del terminal.

La PKI de autorización LDS2 comprende las entidades siguientes:

- autoridades de certificación de verificación de país (CVCA);
- verificadores del documento (DV);
- terminales; y
- punto de contacto único (SPOC).

La distribución y gestión de los certificados de autorización entre las CVCA de un Estado y las DV de otros Estados se realiza a través de un punto de contacto único (SPOC) en cada Estado.

En esta Parte 12 del Doc 9303 se especifica el perfil de la PKI del eMRTD, el perfil de la PKI de autorización y los objetos correspondientes, entre los cuales:

- funciones y responsabilidades de las entidades en la infraestructura;
- algoritmos criptográficos y gestión de claves;
- contenido de certificados y CRL;
- mecanismos de distribución de certificados y CRL; y
- validación de la ruta de certificación.

3. FUNCIONES Y RESPONSABILIDADES

En esta sección se describen las entidades, las funciones y las responsabilidades relativas a la PKI del eMRTD y la PKI de autorización.

3.1 La PKI del eMRTD

La autenticidad e integridad de los datos almacenados en los eMRTD está protegida por autenticación pasiva. Este mecanismo de seguridad se basa en firmas digitales y consiste en las siguientes entidades PKI para las PKI de eMRTD:

- **CA de firma de país (CSCA):** Cada Estado expedidor o autoridad expedidora establece una única CSCA con su punto de confianza nacional en el contexto de los eMRTD. La CSCA expide certificados de clave pública para uno o más firmantes del documento (nacionales) y, con carácter opcional, para otras entidades finales como los firmantes de la lista maestra y firmantes de la lista de desviaciones. La CSCA también expide listas de revocación de certificado (CRL) periódicas indicando si alguno de los certificados expedidos ha sido revocado.
- **Firmante del documento (DS):** El firmante del documento firma digitalmente los datos que han de almacenarse en el eMRTD; esta firma se almacena en el eMRTD, en un objeto de seguridad de documento.

- **Firmantes LDS2**: Un firmante LDS2 firma digitalmente objetos de datos LDS2 de uno o más tipos.

- **Firmante del código de barras (BCS):** Un firmante del código de barras firma digitalmente los datos (encabezado y mensaje) codificados en el código de barras. La firma también se almacena en el código de barras. En este documento se detallan dos ejemplos para el uso del firmante del código de barras: visados y documentos de viaje de emergencia.

- **Sistema de inspección (IS):** El sistema de inspección verifica la firma digital, incluyendo la validación de la ruta de certificación para verificar la autenticidad e integridad de los datos electrónicos almacenados en el eMRTD como parte de la autenticación pasiva.

- **Firmante de la lista maestra:** El firmante de la lista maestra es una entidad opcional que firma digitalmente una lista de certificados CSCA (nacionales y extranjeros) en apoyo del mecanismo bilateral de distribución para certificados CSCA.

- **Firmantes de la lista de desviaciones:** Los firmantes de la lista de desviaciones se utilizan para firmar las listas de desviaciones. Las listas de desviaciones se definen en el Doc 9303-3.

Los medios protegidos de generación de pares de claves ESTARÁN bajo el control del Estado expedidor u organización expedidora. Cada par de claves incluye una clave “privada” y una clave “pública”. Las claves privadas y sistemas o medios conexos ESTARÁN bien protegidos con respecto a cualquier acceso del exterior o no autorizado mediante un diseño inherente y medios de seguridad del soporte físico.

Si bien el certificado CSCA permanece relativamente estático, con el tiempo se creará un gran número de certificados de firmante del documento.

La CSCA de cada Estado expedidor u organización expedidora actúa como punto de confianza para el Estado receptor. El Estado expedidor u organización expedidora distribuye su propia clave pública CSCA a los Estados receptores en forma de certificado. El Estado receptor establece que este certificado (y claves certificadas) son “de confianza” a través de medios fuera de banda y almacena un “punto de confianza” para dicha clave o certificado de confianza. Estos certificados CSCA SERÁN certificados autofirmados expedidos directamente por la CSCA. Los certificados CSCA NO DEBEN ser certificados subordinados o cruzados en una infraestructura PKI de mayores dimensiones. Los certificados de enlace CSCA autoexpedidos también pueden expedirse para ayudar al Estado receptor a establecer confianza en una nueva clave/certificado CSCA después de una renovación de claves.

Nota.— En algunos Estados se exige que un controlador de autoridad de certificación (CCA) centralizado sea la autoridad suprema para publicar certificados autofirmados para todas las aplicaciones. En estos casos, una posible solución es que la CSCA cree un certificado autofirmado (que satisfaga los requisitos del Doc 9303 de la OACI) y que dicho certificado sea refrendado por la CCA (satisfaciendo los propios requisitos CCA del Estado). No obstante, estos certificados refrendados no son parte de la PKI del eMRTD y no se distribuyen a los Estados receptores.

3.1.1 Autoridad de certificación de firma de país

SE RECOMIENDA que los pares de claves CSCA (KPu_{CSCA}, KPr_{CSCA}) se generen y almacenen en una infraestructura de CA fuera de línea y altamente protegida.

La clave privada CSCA (KPr_{CSCA}) se utiliza para firmar certificados de firmante del documento (C_{DS}), otros certificados y CRL.

Los certificados de autoridad de certificación de firma de país (C_{CSCA}) se utilizan para validar los certificados de firmante del documento, certificados de firmante de la lista maestra, certificados de firmante de la lista de desviaciones, CRL y otros certificados expedidos por la CSCA.

Todos los certificados y las CRL DEBEN cumplir con los perfiles especificados en la sección 7 y DEBEN distribuirse utilizando los mecanismos de distribución especificados en la sección 5.

En el caso de los participantes en el PKD, el expedidor de certificado también DEBE enviar cada certificado CSCA (C_{CSCA}) al PKD [para fines de validación de certificados de firmante del documento (C_{DS})].

Las CRL DEBEN expedirse con carácter periódico según se especifica en la sección 4.

3.1.2 Firmantes del documento

SE RECOMIENDA que los pares de claves de firmante del documento (KPu_{DS}, KPr_{DS}) se generen y almacenen en una infraestructura altamente protegida.

La clave privada de firmante del documento (KPr_{DS}) se utiliza para firmar los objetos de seguridad del documento (SO_D).

Los certificados de firmante del documento (C_{DS}) se utilizan para validar los objetos de seguridad de documento (SO_D).

Cada certificado de firmante del documento (C_{DS}) DEBE cumplir con el perfil de certificado definido en la sección 7 y DEBE almacenarse en el CI sin contacto de cada eMRTD que fue firmado con la correspondiente clave privada DS (véase información detallada en el Doc 9303-10). Esto asegura que el Estado receptor tiene acceso al certificado de firmante del documento pertinente a cada eMRTD.

El expedidor de certificado también DEBERÍA enviar a la OACI los certificados de firmante del documento de los participantes en el PKD para su publicación en el Directorio de Clave Pública (PKD) de la OACI.

3.1.3 Firmantes LDS2

Un firmante LDS2 firma digitalmente objetos de datos LDS2 de uno o más tipos.

Cuando sea necesario referirse a un firmante LDS2 como aquel que firma un tipo de objeto de datos LDS2 en concreto, se denominará de la siguiente manera:

- firmante LDS2-TS – firma los sellos de viaje LDS2;
- firmante LDS2-V – firma los visados electrónicos LDS2; y
- firmante LDS2-B – firma la biometría adicional LDS2.

SE RECOMIENDA que cada Estado no tenga más de un firmante LDS2-TS, un firmante LDS2-V y un firmante LDS2-B. Un firmante LDS2 también puede combinar algunas de estas funciones o todas ellas.

Si se requiere una mayor diferenciación, como el lugar en que se añadió el sello de viaje, la funcionaria o funcionario concreto que autorizó a una persona viajera, la funcionaria o funcionario que concedió un visado o el lugar en el que se añadieron datos biométricos adicionales, puede incluirse en un campo propio dentro del propio objeto de datos LDS2 correspondiente.

3.1.4 Firmantes del código de barras

SE RECOMIENDA que los pares de claves de firmante del código de barras (KPu_{BCS}, KPr_{BCS}) se generen y almacenen en una infraestructura altamente protegida.

La clave privada del firmante del código de barras (KPr_{BCS}) se utiliza para firmar los datos (encabezado y mensaje) codificados en el código de barras. La firma también se almacena en el código de barras.

Los certificados de firmante del código de barras (C_{BCS}) se utilizan para validar los datos (encabezado y mensaje) codificados en el código de barras.

Cada certificado de firmante del código de barras (C_{BCS}) DEBE cumplir con el perfil de certificado definido en la sección 7. Los certificados de firmante del código de barras no figuran en el propio sello digital. Por lo tanto, un país que expide documentos protegidos por sellos digitales DEBE publicar todos sus certificados de firmante del código de barras. El canal principal de distribución de los certificados de firmante del código de barras es el PKD/bilateral. Hay también canales secundarios que utilizan otras vías, como la publicación en un sitio web.

El expedidor de certificado también DEBERÍA enviar a la OACI los certificados de firmante del código de barras de los participantes en el PKD para su publicación en el Directorio de Clave Pública (PKD) de la OACI.

El firmante del visado (VS) y el firmante del documento de viaje de emergencia son casos especiales del firmante del código de barras.

3.1.5 Sistema de inspección

Los sistemas de inspección ejecutan la autenticación pasiva para asegurar la integridad y autenticidad de los datos almacenados en el CI sin contacto del eMRTD. Como parte de ese proceso, los sistemas de inspección DEBEN ejecutar la validación de ruta de certificación como se indica en la sección 6.

3.1.6 Firmante de la lista maestra

La clave privada del firmante de la lista maestra se utiliza para firmar las listas maestras de CSCA.

Los certificados de firmantes de la lista maestra se utilizan para validar las listas maestras de CSCA.

3.1.7 Firmante de la lista de desviaciones

La clave privada del firmante de la lista de desviaciones se utiliza para firmar las listas de desviaciones.

Los certificados de firmante de la lista de desviaciones se utilizan para validar las listas de desviaciones.

3.2 La PKI de autorización

El Estado expedidor o la organización expedidora escribe la aplicación de la LDS2 en el CI sin contacto en el momento de la personalización.

Antes de que otro Estado pueda escribir objetos LDS2 en ese CI sin contacto, DEBE obtener la debida autorización del Estado expedidor u organización expedidora. Cada objeto de datos LDS2 está firmado digitalmente por un firmante LDS2 en el Estado escritor y, posteriormente, escrito en el IC sin contacto por un terminal autorizado en ese Estado escritor. El proceso de dos pasos de firma por parte de un firmante y de escritura por parte de un terminal autorizado es parecido al concepto LDS1, en que el firmante del documento firma digitalmente los objetos de seguridad del documento, pero estos se escriben posteriormente en el CI sin contacto por medio del proceso de personalización, como se ilustra en la figura 1. La lectura posterior de los objetos LDS2 del IC sin contacto se realiza a través de los terminales autorizados para la lectura del tipo de objeto LDS2 en cuestión.

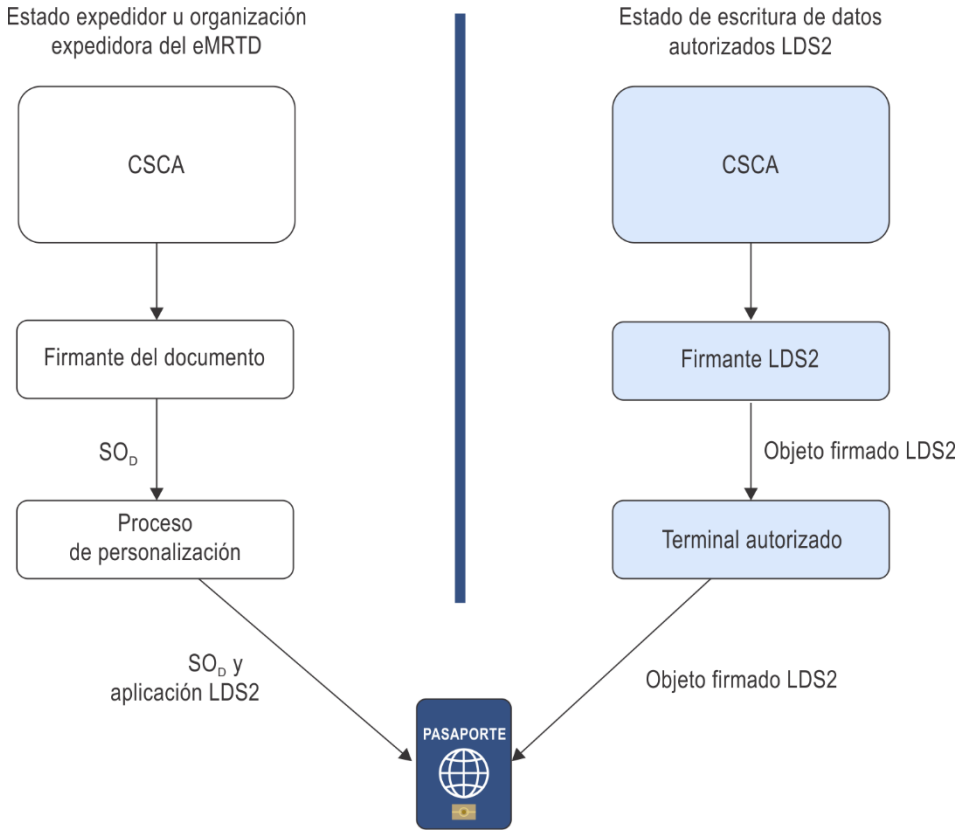

Figura 1. Modelo de confianza y arquitectura de escritura LDS2

La PKI de autorización permite que el Estado expedidor u organización expedidora controle el acceso (lectura y escritura) a los datos LDS2 en el IC sin contacto de los eMRTD que expide.

3.2.1 Autoridad de certificación de verificación de país

Cada Estado expedidor u organización expedidora que permite añadir datos LDS2 en sus eMRTD, DEBE establecer una única autoridad de certificación de verificación de país (CVCA). Esta CVCA es una autoridad de certificación (CA) que es el punto de confianza de la PKI de autorización de ese Estado expedidor u organización expedidora y abarca todas las aplicaciones LDS2. La CVCA puede ser una entidad autónoma o estar integrada en la CSCA de ese mismo Estado u organización. Sin embargo, incluso si comparten ubicación, la CVCA DEBE utilizar un par de claves distinto del de la CSCA. La CVCA determina los derechos de acceso que se concederán a todos los verificadores del documento (DV), extranjeros y nacionales, y expide los certificados que contienen las autorizaciones individuales a cada uno de esos DV.

3.2.2 Verificador del documento

Un verificador del documento (DV) es una CA que, como parte de una dependencia institucional, gestiona un grupo de terminales (por ejemplo, los terminales manejados por la policía de fronteras de un Estado) y expide certificados de autorización para esos terminales. Un DV DEBE haber recibido ya un certificado de autorización de la CVCA

responsable antes de poder expedir certificados asociados a sus terminales. Los certificados expedidos por un DV a los terminales PUEDEN contener la misma autorización, o un subconjunto de esta, que se ha concedido al DV. NO DEBEN contener ninguna autorización más allá de la concedida a la DV.

3.2.3 Terminal/sistema de inspección

En el contexto de la PKI de autorización, un terminal es la entidad que accede al IC sin contacto del eMRTD y escribe en él un objeto de datos LDS2 firmado digitalmente o lee un objeto de datos LDS2. El terminal DEBE disponer de un certificado de autorización expedido por la DV local que concede la autorización requerida. El terminal también se denomina “sistema de inspección”.

3.2.4 Punto de contacto único (SPOC)

Cada Estado que participe en la PKI de autorización LDS2 DEBE establecer un solo SPOC. Este SPOC es la interfaz que se utiliza para todas las comunicaciones entre la CVCA de un Estado y los DV de otro Estado. Las solicitudes y respuestas de certificados se comunican entre los SPOC de cada Estado utilizando el protocolo del SPOC definido en la sección 8.

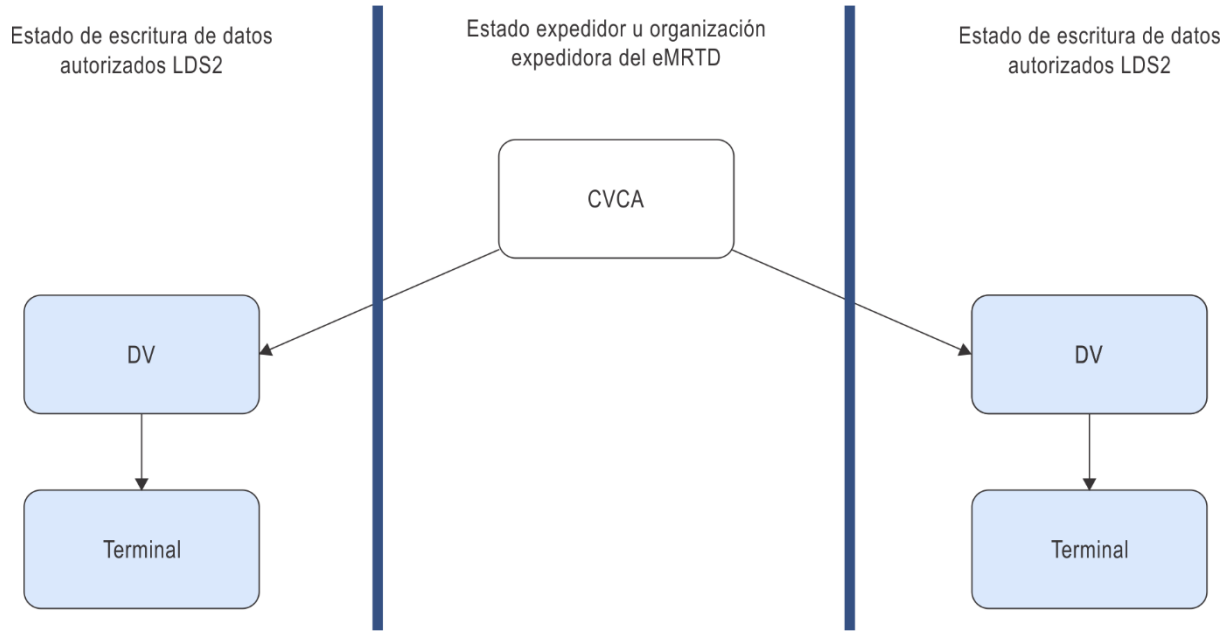

Figura 2. Modelo de confianza de la PKI de autorización

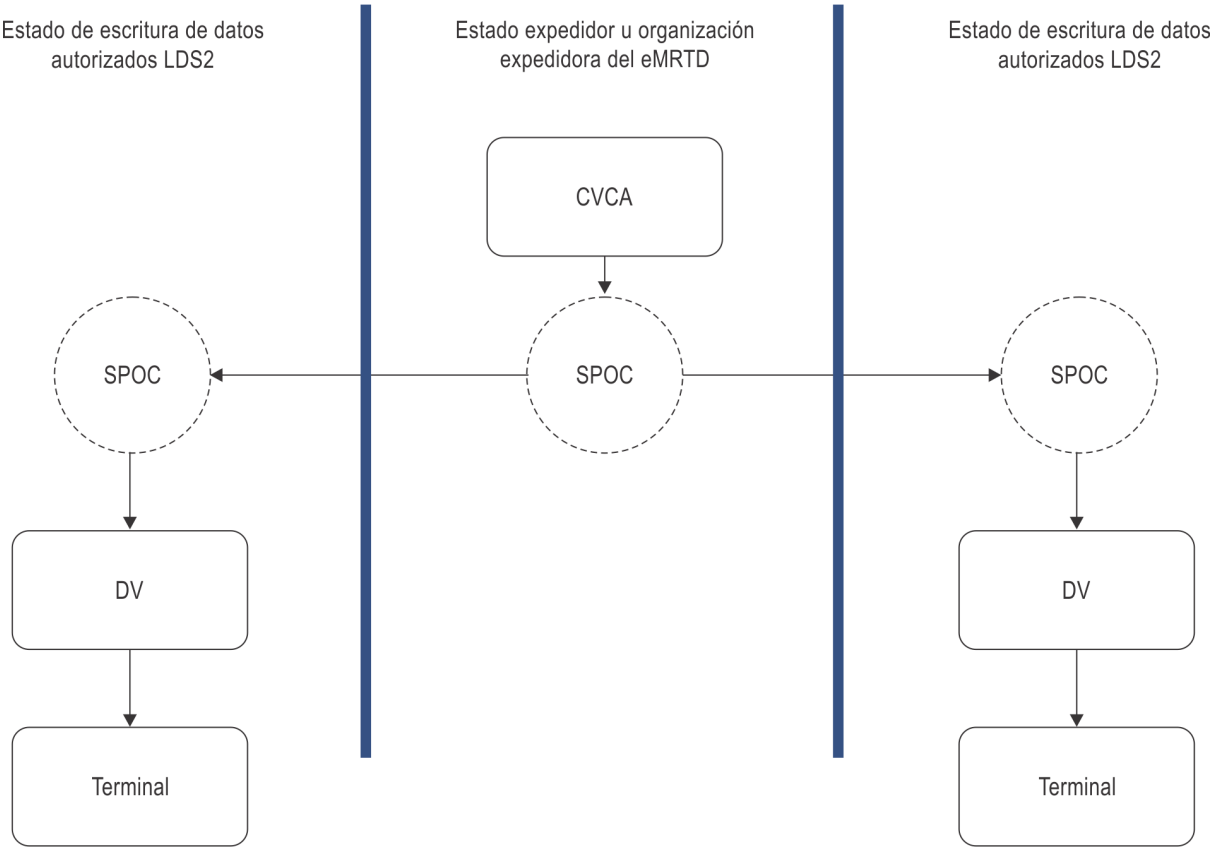

Figura 3. Función del SPOC

4. GESTIÓN DE CLAVES

La gestión de claves se define para las dos infraestructuras de clave pública por separado.

4.1 La PKI del eMRTD

Los Estados expedidores u organizaciones expedidoras TENDRÁN por lo menos dos tipos de pares de claves:

- par de claves de CA de firma de país; y
- par de claves de firmante del documento.

Los Estados expedidores u organizaciones expedidoras PUEDEN tener tipos de pares de claves adicionales, a saber:

- par de claves de firmante de la lista maestra;
- par de claves de firmante de la lista de desviaciones.
- par de claves de firmante LDS2.
- par de claves del cliente SPOC;

- par de claves del servidor SPOC; y
- par de claves de firmante del visado/par de claves de firmante del documento de viaje de emergencia (ambos son tipos de firmantes del código de barras).

Las claves públicas de los certificados de CA de firma de país, firmante del documento y SPOC se expiden utilizando certificados [X.509]. Las claves públicas contenidas en los certificados CSCA se utilizan para verificar la firma CSCA en los certificados de firmante expedidos, certificados SPOC y CSCA, y las CRL expedidas.

Para las claves y los certificados de firmante de la lista maestra, firmante de la lista de desviaciones y comunicaciones, la vida útil de la clave privada y el período de validez certificado quedan a discreción del Estado expedidor u organización expedidora.

Tanto los certificados CSCA como los certificados de firmante del documento se relacionan con el uso de una clave privada y un período de validez de la clave pública que se presenta en la tabla 1.

Tabla 1. Uso y validez de las claves

	Uso de la clave privada	***Validez de la clave pública (suponiendo pasaportes válidos por diez años)***
CA de firma de país	3-5 años	13-15 años
Firmante del documento	Hasta 3 meses[1]	Aproximadamente 10 años
Firmante LDS2-TS	1-2 años	10 años + 3 meses
Firmante LDS2-V	1-2 años	10 años + 3 meses
Firmante LDS2-B	1-2 años	10 años + 3 meses
Cliente SPOC	Sin especificar	6-18 meses
Servidor SPOC	Sin especificar	6-18 meses
Firmante del código de barras de visado	1-2 años	Período de uso de la clave privada + validez del visado
Firmante del código de barras de documento de viaje de emergencia	1 año + 2 meses (los 2 meses están pensados para una transición fluida)	Período de uso de la clave privada + validez del ETD
Firmante de la lista maestra	A discreción del Estado expedidor u organización expedidora	A discreción del Estado expedidor u organización expedidora
Firmante de la lista de desviaciones	A discreción del Estado expedidor u organización expedidora	A discreción del Estado expedidor u organización expedidora
Comunicación	A discreción del Estado expedidor u organización expedidora	A discreción del Estado expedidor u organización expedidora

1. Obsérvese que la correspondiente extensión `privateKeyUsage` en el certificado DS puede ser ligeramente más larga para dar cabida a superposiciones o requisitos de producción.

4.1.1 Claves y certificados de firmante del documento

El período de uso de una clave privada de firmante del documento es mucho más breve que el período de validez del certificado DS para la correspondiente clave pública.

4.1.1.1 Validez de la clave pública de firmante del documento

La vida útil, es decir el período de validez del certificado, de la clave pública del firmante del documento se determina por la concatenación de los dos períodos siguientes:

- el período en que se utilizará la clave privada correspondiente para expedir eMRTD;
- el período de validez más prolongado de cualquier eMRTD expedido con esa clave[2].

El certificado de firmante del documento (C_{DS}) SERÁ válido durante todo este período para permitir la verificación de la autenticidad de los eMRTD. No obstante, la clave privada correspondiente DEBERÍA utilizarse solamente para expedir documentos por un período limitado; una vez expirado el último documento en cuya expedición se utilizó, no se requiere más la clave pública.

4.1.1.2 Período de expedición de la clave privada de firmante del documento

Al emplear sus sistemas, los Estados expedidores u organizaciones expedidoras podrían tener en cuenta el número de documentos que serán firmados por cualquier clave privada de firmante del documento individual.

Un Estado expedidor u organización expedidora puede emplear uno o más firmantes del documento, cada uno con su propio par de claves exclusivo, que estén activos en cualquier momento.

Para minimizar los costos de continuidad de las operaciones en caso de que la clave del firmante del documento sea revocada, el Estado expedidor u organización expedidora que expida un gran número de eMRTD por día podría:

- utilizar un período de uso de clave privada muy breve; o
- emplear varios firmantes del documento o emplear varios firmantes del documento concurrentes que estén activos el mismo tiempo, cada uno con su propia clave privada y certificado de clave pública exclusivo.

Un Estado expedidor u organización expedidora que expida un pequeño número de eMRTD por día puede optar por emplear un único firmante del documento y también contar con un período de uso de clave privada ligeramente más prolongado.

Independientemente del número de eMRTD expedidos por día, o del número de firmantes del documento activos al mismo tiempo, SE RECOMIENDA que el período máximo en que se utilice la clave privada de firmante del documento para firmar los eMRTD sea de tres meses.

2. Algunos Estados expedidores u organizaciones expedidoras pueden expedir eMRTD antes de que sean válidos, por ejemplo, respecto a un cambio de nombre por matrimonio de la persona titular. En estos casos, el “período de validez más prolongado de cualquier eMRTD” comprende la validez real del eMRTD (p. ej., 10 años) más el tiempo máximo entre el momento de expedición del eMRTD y el momento en que cobra validez.

Una vez producido el último documento firmado con una clave privada determinada, SE RECOMIENDA que los Estados expedidores u organizaciones expedidoras borren la clave privada en una forma que se pueda auditar y rendir cuentas.

4.1.2 Claves y certificados de firmante LDS2

Los pares de claves de firmante LDS2 son parecidos a los pares de claves de firmante del documento en el sentido de que el período de uso de la clave privada es mucho más corto que el período de validez del certificado correspondiente. Los certificados DEBEN ser válidos durante la vida útil del eMRTD o el objeto LDS2 firmado (la que sea más larga). Dado que los objetos de datos firmados se escribirán en los eMRTD de varios Estados, estos certificados DEBEN ser válidos como mínimo durante la vida útil más larga del eMRTD (es decir, 10 años).

4.1.2.1 Validez de la clave pública de firmante LDS2

La vida útil, es decir el período de validez del certificado, de la clave pública del firmante LDS2 se determina por la concatenación de los dos períodos siguientes:

- el período en que se utilizará la clave privada correspondiente para firmar objetos LDS2;
- El período de validez que sea más largo de cualquiera de los elementos siguientes:
  - cualquier eMRTD que almacene un objeto LDS2 firmado con esa clave; o
  - cualquier objeto LDS2 firmado con esa clave. Obsérvese que en el caso de un visado electrónico LDS2, es posible que el período de validez de dicho visado se extienda más allá del período de validez del eMRTD en que figura.

4.1.3 Claves y certificados de firmante del código de barras

Un firmante del código de barras es un tipo específico de servidor de firma utilizado para firmar una única categoría de tipo de documento, como un visado o un documento de viaje de emergencia. A fin de seguir las mejores prácticas en este campo, SE RECOMIENDA que solo se utilice un número limitado de claves de firma (un número bajo, de un solo dígito) en paralelo para crear firmas para sellos digitales, salvo que los requisitos operacionales hagan absolutamente necesario un número mayor de claves. Para asegurar la disponibilidad del firmante del código de barras en caso de un incidente de seguridad relacionado con las claves de firma, SE RECOMIENDA disponer de medidas que velen por la continuidad de las operaciones (p. ej., preparación de claves de reserva, un sitio de respaldo, etc.).

Para facilitar la gestión de los certificados correspondientes (véase la sección 5), el número de claves de validación de firma publicadas DEBE limitarse a cinco claves de firma por año.

4.1.3.1 Validez de la clave pública de firmante del código de barras

Esta sección es aplicable a todos los firmantes del código de barras, incluido el firmante del visado y el firmante del documento de viaje de emergencia.

La vida útil, es decir el período de validez del certificado de la clave pública del firmante del código de barras se determina por la concatenación de los dos períodos siguientes:

- el período en que se utilizará la clave privada correspondiente para expedir un visado o ETD;
- el período de validez más prolongado de cualquier documento expedido con esa clave[3].

El certificado de firmante del código de barras SERÁ válido durante todo este período para permitir la verificación de la autenticidad del documento en cuestión. No obstante, la clave privada correspondiente DEBERÍA utilizarse solamente para expedir documentos por un período limitado; una vez expirado el último documento en cuya expedición se utilizó, no se requiere más la clave pública.

Período de uso de la clave privada:	en función del perfil de documento
Validez del certificado:	período de uso de la clave privada + validez del documento

Ejemplo

Nota.— Los períodos de validez reales utilizados para el cálculo en este ejemplo no implican ninguna recomendación.

Supongamos que se expiden documentos con un período de validez de 5 años, y que el período de uso de la clave privada del certificado de firmante del código de barras es de 1 año. Entonces, la validez del certificado de firmante del código de barras es de 1 + 5 = 6 años. Si el período de uso de la clave privada del certificado CSCA es de 3 años, la validez del certificado CSCA es de 3 + 6 = 9 años.

4.1.4 Claves y certificados CSCA

El período de uso de una clave privada CSCA es mucho más breve que el período de validez del certificado CSCA para la correspondiente clave pública.

4.1.4.1 Validez de la clave pública de CA de firma de país

La vida útil, es decir la validez de certificado, de la clave pública CSCA se determina concatenando los períodos siguientes:

- el período en que se utilizará la clave privada CSCA correspondiente para firmar cualquier certificado que dependa de la CSCA; y
- la vida útil máxima de la clave de cualquier certificado expedido por la CSCA.

4.1.4.2 Período de expedición de la clave privada de CA de firma de país

El período de uso de la clave privada CSCA para firmar certificados CRL constituye un delicado equilibrio entre los factores siguientes:

- en el caso improbable de que la clave privada CA de firma de país de un Estado expedidor u organización expedidora se vea comprometida, la validez de todos los eMRTD expedidos utilizando claves de firmante del documento cuyos certificados fueron firmados por la clave privada CSCA

3. Algunos Estados expedidores u organizaciones expedidoras pueden expedir eMRTD antes de que sean válidos, por ejemplo, respecto a un cambio de nombre por matrimonio de la persona titular. En estos casos, el “período de validez más prolongado de cualquier eMRTD” comprende la validez real del eMRTD (p. ej., 10 años) más el tiempo máximo entre el momento de expedición del eMRTD y el momento en que cobra validez.

comprometida puede resultar cuestionable. En consecuencia, los Estados expedidores u organizaciones expedidoras PODRÍAN hacer que dicho período de expedición sea relativamente breve;

- no obstante, si se mantiene el período de expedición muy breve, ello conduciría a la existencia, en cualquier momento determinado, de un gran número de claves públicas CSCA. Esto puede conducir a una compleja gestión de certificados en los sistemas de procesamiento fronterizo.

Por consiguiente, SE RECOMIENDA que el par de claves CSCA de un Estado expedidor u organización expedidora sea sustituido cada tres a cinco años.

4.1.4.3 Sustitución de clave CA de firma de país

Las claves CSCA proporcionan los puntos de confianza de todo el sistema y sin ellos el sistema se desmoronaría. Por consiguiente, los Estados expedidores u organizaciones expedidoras DEBERÍAN planificar cuidadosamente la sustitución de su par de claves CSCA. Una vez transcurrido el período de expedición de la clave privada inicial de firma CSCA, el Estado expedidor u organización expedidora siempre tendrá por lo menos dos certificados CSCA (C_{CSCA}) válidos en cualquier momento.

Los Estados expedidores u organizaciones expedidoras DEBEN notificar a los Estados receptores que se ha previsto una renovación de su clave CSCA. Esta notificación DEBE proporcionarse con 90 días de antelación respecto de la renovación de la clave. Una vez ocurrida la renovación de la clave se distribuye a los Estados receptores el nuevo certificado CSCA (certificando la nueva clave pública CSCA).

Si el certificado CSCA es un certificado autofirmado nuevo, la autenticación de dicho certificado debería efectuarse utilizando un método fuera de banda.

Cuando ocurre una renovación de clave CSCA, DEBE expedirse un certificado que enlace la nueva clave con la clave antigua para proporcionar una transición segura a las partes que confían en el certificado. En general, esto se consigue mediante la expedición de un certificado autoexpedido donde los campos del expedidor y del sujeto son idénticos pero la clave utilizada para verificar la firma representa el antiguo par de claves y la clave pública certificada representa el nuevo par de claves. No es necesario verificar estos certificados de enlace CSCA utilizando un método fuera de banda dado que la firma en el certificado de enlace CSCA se verifica utilizando una clave pública en que ya se confía para esa CSCA. Las listas maestras también pueden utilizarse para distribuir enlaces CSCA y certificados raíz CSCA autofirmados.

Los Estados expedidoras u organizaciones expedidoras deberían abstenerse de utilizar su nueva clave privada CSCA por los dos primeros días después de la renovación de la clave CSCA, para asegurar que el correspondiente nuevo certificado de clave pública CSCA se ha distribuido satisfactoriamente.

Los Estados expedidores u organizaciones expedidoras DEBEN usar la clave privada CSCA más reciente para firmar todos los certificados y las CRL.

4.1.5 Revocación de certificados

Los Estados expedidores u organizaciones expedidoras pueden tener que revocar certificados en caso de que ocurra un incidente (como una clave comprometida).

Todas las CSCA DEBEN producir información periódica de revocación en forma de listas de revocación de certificados (CRL).

Las CSCA DEBEN expedir por lo menos una CRL cada 90 días, incluso si no se ha revocado ningún certificado desde que se expidió la CRL anterior. Las CRL PUEDEN expedirse con una frecuencia mayor que cada 90 días pero no mayor que cada 48 horas.

Si se revoca un certificado, se DEBE distribuir una CRL indicando dicha revocación dentro de las 48 horas siguientes.

Solo pueden revocarse los certificados y no los objetos de seguridad de documento. El uso de las CRL se limita a las notificaciones de certificados revocados que han sido expedidos por la CSCA que publicó la CRL (incluyendo avisos de revocación para certificados CSCA, certificados DS, certificados de firmante de la lista maestra, certificados de firmante de la lista de desviaciones y cualquier otro tipo de certificado expedido por dicha CA).

En la aplicación del eMRTD no se utilizan CRL particionadas. Todos los certificados revocados por una CSCA, incluyendo certificados DS, certificados CSCA, certificados de firmante de la lista maestra y certificados de firmante de la lista de desviaciones se enumeran en la misma CRL. Aunque la CRL siempre se firma con la clave privada de firma CSCA más reciente (en vigor), la CRL incluye avisos de revocación para certificados firmados con esa misma clave privada así como certificados firmados con claves privadas de firma CSCA anteriores.

4.1.5.1 Revocación de certificados CSCA

La revocación de un certificado CSCA es tanto extrema como difícil. Después de informar a un Estado receptor de que se ha revocado un certificado CSCA, todos los otros certificados expedidos utilizando la correspondiente clave privada CSCA quedan efectivamente revocados.

Cuando se ha firmado un certificado de enlace CSCA utilizando una clave privada CSCA antigua para certificar una clave pública CSCA nueva (véase “Sustitución de clave de CA de firma de país” en 4.1.4.3), la revocación del antiguo certificado CSCA también REVOCARÁ el nuevo certificado CSCA.

Si se debe revocar un certificado CSCA, la CSCA pertinente puede expedir una CRL firmada con la clave privada que corresponde a la clave pública que se revoca, dado que esta es la única clave que los usuarios de la CRL podrán verificar en esa oportunidad. La clave pública CSCA debería considerarse válida solo para el fin de verificar esa firma de CRL. Una vez que el usuario de la CRL ha verificado la firma de CRL, la clave privada de firma CSCA se considera comprometida y el certificado se revoca para todas las verificaciones futuras.

Para expedir nuevos documentos, el Estado expedidor o la organización expedidora DEBE reiniciar su proceso de autenticación desde el comienzo expidiendo un nuevo certificado raíz CSCA, distribuyendo dicho certificado a los Estados receptores y apoyando la confirmación fuera de banda de que el certificado recibido por cada Estado receptor es en verdad el certificado CSCA auténtico vigente.

4.1.5.2 Revocación de otros certificados

Cuando un Estado expedidor u organización expedidora desee revocar un certificado de firmante expedido por la CSCA, no tiene por qué esperar hasta que expire el período `nextUpdate` de la CRL vigente para expedir una nueva CRL. SE RECOMIENDA que se expida la nueva CRL dentro de las 48 horas siguientes a la notificación de revocación.

4.1.6 Algoritmos criptográficos

Los Estados expedidores u organizaciones expedidoras PUEDEN admitir diferentes algoritmos para usarse en sus claves CSCA y de firma de documentos. Por ejemplo, es posible que la CSCA se haya expedido utilizando RSA, pero los certificados de firmante podrían ser DSA de curva elíptica (ECDSA) y viceversa.

Los Estados expedidores u organizaciones expedidoras ELEGIRÁN longitudes de clave apropiadas que ofrezcan protección contra ataques. DEBERÍAN tenerse en cuenta los catálogos criptográficos adecuados.

Los Estados receptores DEBEN admitir todos los algoritmos en los puntos donde deseen validar la firma en los eMRTD.

Para uso en sus claves CSCA, de firma y, cuando corresponda, objetos de seguridad de documento, los Estados expedidores u organizaciones expedidoras ADMITIRÁN uno de los algoritmos siguientes.

4.1.6.1 RSA

Los Estados expedidores u organizaciones expedidoras que implementen el algoritmo RSA para la generación de firmas y verificación de certificados y objetos de seguridad de documento (SO_D) UTILIZARÁN [RFC 4055]. La [RFC 4055] especifica dos mecanismos de firma: RSASSA-PSS y RSASSA-PKCS1_v15. SE RECOMIENDA que los Estados expedidores u organizaciones expedidoras generen firmas con arreglo a RSASSA-PSS, pero los Estados receptores también DEBEN prepararse para verificar firmas con arreglo a RSASSA-PKCS1_v15.

4.1.6.2 Algoritmo de firma digital (DSA)

Los Estados expedidores u organizaciones expedidoras que implementen el DSA para la generación o verificación de firma UTILIZARÁN [FIPS 186-4].

4.1.6.3 DSA de curva elíptica (ECDSA)

Los Estados expedidores u organizaciones expedidoras que implementen el ECDSA para la generación o verificación de firma UTILIZARÁN [X9.62] o [ISO/IEC 15946]. Los parámetros de dominio de curva elíptica utilizados para generar el par de claves ECDSA DEBEN describirse explícitamente en los parámetros de la clave pública, es decir, los parámetros deben ser del tipo ECParameters (sin curvas denominadas y sin parámetros implícitos) y DEBEN incluir el cofactor opcional. Los puntos ECPoints DEBEN tener formato no comprimido.

SE RECOMIENDA seguir la directriz [TR 03111].

4.1.6.4 Algoritmos de condensación

Los algoritmos SHA-224, SHA-256, SHA-384 y SHA-512 son los únicos algoritmos de condensación permitidos. Véase [FIPS 180-2].

4.1.7 Algoritmos criptográficos para certificados de firmante LDS2

Debido a que los certificados y los objetos firmados LDS2 se almacenan en el CI sin contacto, deben ser lo más compactos posible. Por lo tanto, los firmantes LDS2 DEBEN utilizar el ECDSA, independientemente del algoritmo utilizado en las claves CSCA y de firma del documento.

4.2 La PKI de autorización

Los Estados expedidores u organizaciones expedidoras que implementen la LDS2 TENDRÁN por lo menos los tipos de pares de claves siguientes:

- par de claves de CA de verificación de país (CVCA);
- par de claves del verificador del documento (DV); y
- par de claves de terminal.

Las claves públicas CVCA y DV son certificadas por la CVCA. Las claves públicas de terminal son certificadas por el DV. Los certificados de clave pública CVCA, DV y de terminal son certificados verificables mediante tarjeta que DEBEN cumplir con sus respectivos perfiles de certificado definidos en la sección 7. No hay mecanismo de revocación para los certificados CVCA, DV o de terminal. Por lo tanto, sus períodos de validez son mucho más cortos que lo de los tipos de certificado X.509.

El período de uso de clave privada no se especifica y está sujeto a la discreción del Estado. Sin embargo, el período de uso de la clave privada DEBE ser como máximo igual al período de validez de la clave pública. El período de validez de clave pública de los pares de claves CVCA, DV y de terminal se indica en la tabla 2.

Tabla 2. Validez del uso de clave de certificados verificables mediante tarjeta

	Validez de clave pública
CVCA	6 meses – 3 años
DV	2 semanas – 3 meses
Terminal	1 día – 1 mes

4.2.1 Algoritmos criptográficos para la autenticación del terminal

El algoritmo utilizado para la autenticación del terminal en la PKI de autorización es determinado por la CVCA del Estado expedidor del eMRTD. En una cadena de certificados SE DEBEN utilizar los mismos algoritmos de firma, parámetros de dominio y tamaños de clave (es decir, los certificados CVCA, DV y de terminal para una determinada autorización). Por lo tanto, los verificadores del documento y los terminales deberán disponer de varios pares de claves. Los certificados de enlace CVCA PUEDEN incluir una clave pública que se desvíe de los parámetros vigentes, es decir, la CVCA PUEDE cambiar a un nuevo algoritmo de firma, nuevos parámetros de dominio o tamaños de clave.

Para la autenticación del terminal SE PUEDE utilizar RSA o ECDSA. En el Doc 9303-11 figura información detallada.

4.2.2 Algoritmos criptográficos para el SPOC

En la tabla 3 se muestran los conjuntos de cifrado TLS que deben usarse para el protocolo del SPOC.

Tabla 3. Conjuntos de cifrado TLS

Conjunto de cifrado	Algoritmo de intercambio de certificados y claves
TLS_RSA_WITH_AES_128_CBC_SHA	RSA
TLS_DHE_RSA_WITH_AES_128_CBC_SHA	DHE_RSA
TLS_RSA_WITH_AES_256_CBC_SHA	RSA
TLS_DHE_RSA_WITH_AES_256_CBC_SHA	DHE_RSA
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA	ECDHE_ECDSA
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA	ECDHE_ECDSA

En el ámbito de la negociación del protocolo de enlace TLS, el cliente ADMITIRÁ todos los conjuntos de cifrado TLS definidos en la tabla 3. Tanto el servidor como el cliente ADMITIRÁN la autenticación basada en RSA y ECDSA. Está permitido que un servidor solicite y también que el cliente envíe un certificado de cliente de un tipo diferente al del servidor.

El uso de la negociación de claves ECDHE_ECDSA en el protocolo de enlace TLS es conforme a las adiciones definidas en [TLSECC], [TLS1.2] y [TLSEXT]. Tanto el cliente como el servidor ADMITIRÁN las extensiones de curvas elípticas apropiadas definidas en la especificación [TLSECC] en el ámbito del protocolo de enlace TLS. Las curvas elípticas y los formatos de puntos de curva elíptica admitidos se definen en la sección 5 de [TLSECC]. El uso de los conjuntos de cifrado TLS admitidos definidos en la tabla 3 que utilizan la norma de cifrado avanzado (AES) para el cifrado DEBE ser conforme a la especificación [TLSAES].

5. MECANISMOS DE DISTRIBUCIÓN

En el caso de la PKI del eMRTD, los objetos PKI deben distribuirse a los Estados receptores. Se utilizan varios mecanismos de distribución diferentes dependiendo del tipo de objeto y los requisitos operacionales. Es importante destacar que la distribución de estos objetos NO establece confianza en ellos ni en las claves privadas o públicas relacionados con ellos. Los mecanismos para establecer confianza se especifican en la sección 6.1.

El mecanismo de distribución de la PKI de autorización se trata en la sección 8.

Los objetos que deben distribuirse de los Estados expedidores u organizaciones expedidoras a los Estados receptores comprenden:

- certificados CSCA;
- certificados de enlace CSCA;
- certificados de firmante del documento;
- certificados de firmante LDS2;

- certificados CVCA iniciales;
- certificados de enlace CVCA;
- certificados DV;
- certificados de firmante del código de barras;
- CRL (nulas y no nulas);
- certificados de firmante de la lista maestra; listas maestras; y
- certificados de firmante de la lista de desviaciones; listas de desviaciones.

Los mecanismos de distribución utilizados en la PKI del eMRTD y de autorización comprenden:

- PKD;
- intercambio bilateral;
- SPOC;
- listas maestras;
- listas de desviaciones; y
- CI sin contacto del eMRTD.

Se especifica un mecanismo principal y secundario de distribución, cuando corresponda, para cada objeto según se muestra en la tabla 4.

Tabla 4. Distribución de objetos PKI

	IC sin contacto	**SPOC**	**Bilateral**	**PKD**	**Lista de desviaciones**	**Lista maestra**	**Notas**
Certificados CSCA			Y (primario)			Y (secundario)	
Certificados de firmante del documento	Y (primario)			Y (secundario)			Los certificados se escriben en el mismo momento que el $SO_{D.}$
Certificados de firmante LDS2	Y						Los certificados se escriben en el mismo momento que el objeto firmado.
Certificados CVCA iniciales	Y						El certificado se escribe en el momento de la personalización del eMRTD.

	IC sin contacto	SPOC	Bilateral	PKD	Lista de desviaciones	Lista maestra	Notas
Certificados de enlace CVCA	Y	Y					Los certificados se distribuyen a los DV a través del SPOC y el punto de confianza CVCA se actualiza en el IC sin contacto en la siguiente verificación.
Certificados DV		Y					Solo se distribuye al DV del sujeto.
CRL (nulas y no nulas)			Y (secundario)	Y (primario)			Las CRL expedidas por la CSCA comprenden información de revocación pertinente para los objetos PKI LDS2.
Certificados de firmante de la lista maestra						Y	
Certificados de firmante del código de barras			Y (secundario)	Y (primario)			Los firmantes del código de barras no están codificados en el propio código y, por lo tanto, la distribución debe asegurarse para la validación del código de barras.
Listas maestras			Y	Y			
Certificado de firmante de la lista de desviaciones					Y		

Desde el punto de vista de las operaciones, los Estados receptores no están obligados a usar la fuente principal y la secundaria. En las operaciones diarias del sistema de inspección, queda a discreción de la autoridad competente si corresponde utilizar la fuente principal o la secundaria. Si la autoridad del Estado receptor utiliza la fuente secundaria para un certificado o CRL en sus operaciones diarias, debería estar preparada para admitir también la fuente principal.

Los Estados expedidores u organizaciones expedidoras deben planificar sus estrategias de renovación de pares de claves para las claves CSCA y las claves de firmante del documento a efecto de permitir la propagación de certificados y CRL en los sistemas de control fronterizos de los Estados receptores en forma oportuna. La propagación ideal ocurrirá dentro de un plazo de 48 horas, pero algunos Estados receptores pueden tener puestos fronterizos distantes o

conectados en forma deficiente a los cuales puede insumir más tiempo propagar los certificados y CRL. Los Estados receptores DEBERÍAN hacer todo lo posible por distribuir estos certificados y CRL a todas las estaciones fronterizas dentro de un plazo de 48 horas.

Los Estados expedidores u organizaciones expedidoras deberían prever que los Estados receptores propagarán los certificados CSCA (C_{CSCA}) dentro de un plazo de 48 horas.

Los Estados expedidores u organizaciones expedidoras aseguran la propagación oportuna de los certificados de firmante del documento (C_{DS}) incluyendo el certificado de firmante del documento (C_{DS}) dentro del objeto de seguridad de documento (SO_D). También deberían prever que los certificados de firmante del documento (C_{DS}) publicados en el PKD también se propagarán a las estaciones fronterizas dentro de un plazo de 48 horas.

Los certificados de firmante del código de barras no figuran en el propio sello digital. Por lo tanto, un país que expide documentos protegidos por sellos digitales DEBE publicar todos sus certificados de firmante del código de barras. El canal principal de distribución de los certificados de firmante del código de barras es el PKD/bilateral. Hay también canales secundarios que utilizan otras vías, como la publicación en un sitio web.

En el caso de los firmantes del código de barras, la publicación DEBE cumplir los principios siguientes:

- en cuanto se cree un nuevo certificado, DEBE publicarse dentro de un plazo máximo de 48 horas; y
- los certificados DEBEN permanecer publicados hasta su caducidad o revocación.

Los Estados receptores DEBERÍAN hacer todo lo posible, ya sea por medios electrónicos o de otro tipo, por ocuparse de las CRL, incluidas las expedidas en circunstancias excepcionales.

La propagación oportuna de los certificados de firmante de la lista maestra se asegura incluyéndolos en cada lista maestra.

5.1 Mecanismo de distribución mediante el PKD

La OACI proporciona un servicio de Directorio de Claves Públicas (PKD). Este servicio ACEPTARÁ objetos PKI, incluyendo certificados, CRL y listas maestras, de participantes en el PKD, los almacenará en un directorio y los pondrá a disposición de todos los Estados receptores.

Los certificados CSCA (C_{CSCA}) no se almacenan individualmente como parte del servicio PKD de la OACI. No obstante, pueden estar presentes en el PKD si están contenidos en las listas maestras.

Cada certificado permanece en el PKD hasta que haya expirado su período de validez, independientemente de si se continúa o no utilizando la clave privada correspondiente.

Los certificados, CRL y listas maestras almacenados en el PKD por todos los participantes en dicho directorio SE PONDRÁN a disposición de todas las partes (incluyendo aquellas que no participen en el PKD) que necesiten esta información para validar la autenticidad e integridad de los datos eMRTD, objetos LDS2 y objetos VDS almacenados digitalmente.

5.1.1 Carga en el PKD

Solo los participantes en el PKD PUEDEN cargar en él certificados, CRL y listas maestras. Todos los certificados y CRL DEBEN cumplir con los perfiles de la sección 7. Todas las listas maestras DEBEN cumplir con las especificaciones de la sección 9.

El PKD consiste en un “directorio de escritura” y un “directorio de lectura”. Los participantes en el PKD UTILIZARÁN el protocolo ligero de acceso a directorios (LDAP) para cargar sus objetos en el directorio de escritura. Una vez verificada la firma digital de un objeto, y completadas otras verificaciones debidas, el objeto se publica en el directorio de lectura.

5.1.2 Descarga del PKD

El acceso de lectura a todos los certificados, CRL y listas maestras publicadas en el PKD SE PONDRÁ a disposición de los participantes en el PKD y de los no participantes. El control de acceso NO SE IMPLEMENTARÁ para acceso de lectura al PKD.

Es responsabilidad del Estado receptor distribuir los objetos descargados del PKD a sus sistemas de inspección y mantener una caché de CRL vigentes junto con los certificados necesarios para verificar las firmas en los datos del eMRTD.

5.2 Mecanismo de distribución por intercambio bilateral

Para las CRL y los certificados CSCA (C_{CSCA}), el canal principal de distribución será el intercambio bilateral entre Estados expedidores u organizaciones expedidoras y Estados receptores. También puede utilizarse el intercambio bilateral para distribuir listas maestras.

La tecnología específica empleada para dicho intercambio bilateral puede variar dependiendo de las políticas de cada Estado expedidor u organización expedidora que tenga necesidad de distribuir sus certificados, CRL y listas maestras, así como de las políticas de cada Estado receptor que necesite el acceso a dichos objetos. He aquí algunos ejemplos de tecnologías que pueden utilizarse en el intercambio bilateral:

- correo/valija diplomática;
- intercambio por correo electrónico;
- descarga de un sitio web relacionado con la CSCA expedidora; y
- descarga de un servidor LDAP relacionado con la CSCA expedidora.

La lista anterior no es exhaustiva y también pueden emplearse otras tecnologías.

5.3 Mecanismo de distribución por lista maestra

Las listas maestras son una tecnología de apoyo para el plan de distribución bilateral. Como tal, la distribución de certificados CSCA por las listas maestras es un subconjunto del plan de distribución bilateral.

Una lista maestra es una lista firmada digitalmente de certificados CSCA en los que “confía” el Estado receptor u organización receptora que expidió la lista maestra. Los certificados raíz CSCA autofirmados y los certificados de enlace CSCA pueden incluirse en una lista maestra. La estructura y el formato de la lista maestra se definen en la sección 8. La publicación de una lista maestra permite que otros Estados receptores u organizaciones receptoras obtengan un conjunto de certificados CSCA de una única fuente (el expedidor de la lista maestra) en vez de establecer un acuerdo de intercambio bilateral directo con cada una de las autoridades u organizaciones expedidoras o representadas en la lista.

La CSCA autoriza a un firmante de la lista maestra a que recopile, firme digitalmente y expida listas maestras. Las listas maestras NO DEBEN ser firmadas y expedidas directamente por la propia CSCA. Los certificados de firmante de la lista maestra DEBEN cumplir con el perfil de certificado que se define en la sección 7.

Antes de expedir una lista maestra, el firmante de la lista maestra expedidor DEBERÍA validar extensamente los certificados CSCA que han de refrendarse, incluyendo la garantía de que los certificados en verdad pertenecen a las CSCA identificadas. Los procedimientos aplicados para esta validación fuera de banda DEBERÍAN reflejarse en las políticas de certificado publicadas de la CSCA que expidió el certificado de firmante de la lista maestra.

Cada lista maestra DEBE incluir el certificado de firmante de la lista maestra que se utilizará para verificar la firma en dicha lista así como los certificados CSCA de la autoridad que expidió dicho certificado de firmante de la lista maestra.

Si un Estado receptor ha recibido nuevos certificados CSCA y sus procedimientos de validación se han completado, SE RECOMIENDA que se recopile y expida una nueva lista maestra.

El uso de una lista maestra permite una distribución más eficiente de certificados CSCA para algunos Estados receptores. No obstante, un Estado receptor que utilice listas maestras todavía DEBE determinar sus propias políticas para establecer confianza en los certificados contenidos en la lista (véase información detallada en la sección 6).

6. CONFIANZA Y VALIDACIÓN DE LA PKI

La confianza y validación de la PKI es distinta para la PKI del eMRTD y la PKI de autorización.

6.1 La PKI del eMRTD

En el entorno de la PKI del eMRTD, los sistemas de inspección en los Estados receptores desempeñan la función de partes que confían en la PKI. La verificación satisfactoria de la firma digital en el objeto de seguridad de documento de un eMRTD asegura la autenticidad e integridad de los datos almacenados en el CI sin contacto de dicho eMRTD. Este proceso de verificación de firma requiere que la parte que confía establezca que la clave pública de firmante del documento utilizada para verificar la firma es en sí “de confianza”.

Los diversos mecanismos de distribución definidos en la sección 5 permiten que los Estados receptores tengan acceso a los certificados y CRL que necesitan para verificar las firmas digitales en cuestión. No obstante, estos planes de distribución no establecen confianza en los certificados, CRL o en las claves públicas que se utilizarán para verificar las firmas en esos certificados y CRL.

Las claves públicas contenidas en los certificados CSCA (CCSCA) se utilizan para verificar la firma digital en los certificados y las CRL. Por consiguiente, para aceptar un eMRTD de otro Estado expedidor, el Estado receptor DEBE haber colocado ya en alguna forma de depósito de confianza, accesible por su sistema de control fronterizo, una copia “de confianza” del certificado CSCA (CCSCA) del Estado expedidor u organización expedidora, u otra forma de información sobre el punto de confianza para la clave pública CSCA obtenida del certificado.

Es responsabilidad del Estado receptor establecer confianza en los certificados CSCA (C_{CSCA}) y almacenar dichos certificados (o información obtenida en los mismos) como puntos de confianza en forma segura para que los utilicen sus sistemas de inspección fronterizos.

6.1.1 Gestión del punto de confianza

Como se especifica en [RFC 5280], debe establecerse un punto de confianza que pueda utilizarse para fundamentar el procedimiento de validación de un firmante del documento, firmante de la lista maestra, firmante de la lista de desviaciones u otro tipo de certificado.

Cada punto de confianza comprende una clave pública “de confianza” y sus metadatos conexos. Los puntos de confianza DEBEN incluir, como mínimo:

- la clave pública de confianza y cualesquiera parámetros de clave conexos;
- el algoritmo de clave pública;
- el nombre del propietario de la clave; y
- el valor de la extensión `SubjectAltName` del certificado CSCA que contiene el código de tres letras asignado por la OACI de la autoridad u organización expedidora. Aunque no se utilice en la ruta de certificación o procedimientos de validación de CRL, se utiliza en la autenticación pasiva definida en el Doc 9303-11.

En la aplicación del eMRTD, se establece un punto de confianza separado para cada clave pública de una determinada CSCA. Para la clave pública inicial obtenida de una CSCA, la confianza DEBE establecerse mediante un mecanismo fuera de banda. Por ejemplo, si un certificado CSCA fue descargado de un servidor relacionado con la CSCA, podría utilizarse una comunicación fuera de banda (p. ej., teléfono o correo electrónico) para verificar que el certificado descargado es en verdad el certificado auténtico de dicha CSCA. Además, la parte que confía puede analizar las políticas, procedimientos y prácticas de la CSCA expedidora para determinar si son suficientemente seguros como para satisfacer los requisitos locales sobre uso de certificados. Una vez establecido el punto de confianza inicial para una determinada CSCA, el proceso podría simplificarse para claves subsiguientes en la misma CSCA. Si esa autoridad expide un certificado de enlace CSCA, entonces podría omitirse la comunicación fuera de banda con la autoridad para verificar la autenticidad de un nuevo certificado porque la clave pública con confianza ya establecida para esa misma CSCA se utiliza para verificar la firma en el certificado de enlace de dicha autoridad.

La información sobre puntos de confianza puede almacenarse como copia “de confianza” en el propio certificado CSCA, o en algún otro formato fiable.

Dado que las firmas en los certificados expedidos por las CSCA deben ser verificables mucho después de que la CSCA haya actualizado su par de claves, un Estado receptor tendrá normalmente más de un punto de confianza para la misma CSCA en un momento dado. Si una CSCA ha experimentado un cambio de nombre, algunos de estos puntos de confianza contendrán el antiguo nombre de la CSCA y otros contendrán el nuevo nombre.

6.1.2 Validación de certificados/CRL y verificación de las revocaciones

Como parte del proceso de verificar la autenticidad e integridad de los objetos de datos en la aplicación del eMRTD (p. ej., objetos de seguridad de documento, listas maestras, listas de desviaciones, etc.), el Estado receptor:

- valida el certificado utilizado para verificar la firma del objeto de datos (p. ej., certificado de firmante del documento, certificado de firmante de la lista maestra, certificado de firmante de la lista de desviaciones);
- valida la CRL que se utiliza para verificar el estado de revocación del certificado en cuestión; y
- procesa la CRL para verificar el estado de revocación del certificado en cuestión.

Se dispone de ejemplos de algoritmos para estos procesos, como los especificados en [RFC 5280]. Los Estados receptores no tienen que implementar el algoritmo específico definido en RFC 5280, pero DEBEN proporcionar funciones equivalentes al comportamiento externo resultante de este procedimiento. Una implementación particular puede utilizar cualquier algoritmo siempre que este obtenga el resultado correcto.

En el apéndice D se proporciona orientación para los Estados receptores que optan por basar su algoritmo en el especificado en [RFC 5280].

6.1.3 Autoridad de validación del código de barras

La autoridad de validación del código de barras valida un sello digital aplicando una política de validación. En el Doc 9303-13 especifican en detalle los criterios de validación y los algoritmos para generar un estado de validación.

En la figura 4 se ilustra la arquitectura funcional de la autoridad de validación del código de barras. La autoridad de validación del código de barras se basa en un *software* de validación que puede instalarse en cualquier computadora utilizada por las autoridades de control fronterizo.

El *software* de validación está conectado con un lector que toma una imagen del código de barras para recuperar el código de barras y la ZLM del documento, y también una imagen del documento para recuperar su ZLM. Para verificar la validez de la firma del sello digital, el *software* de validación DEBERÍA sincronizarse con el punto de publicación de la PKI como mínimo cada 24 horas para recuperar los últimos certificados de firmante del código de barras y las CRL.

El *software* de validación del código de barras descodifica el sello digital y las ZLM de cualquier documento asociado (p. ej., un visado o pasaporte), valida la firma del sello digital y aplica una política de validación (véase el Doc 9303-13) para generar un estado de validación del documento.

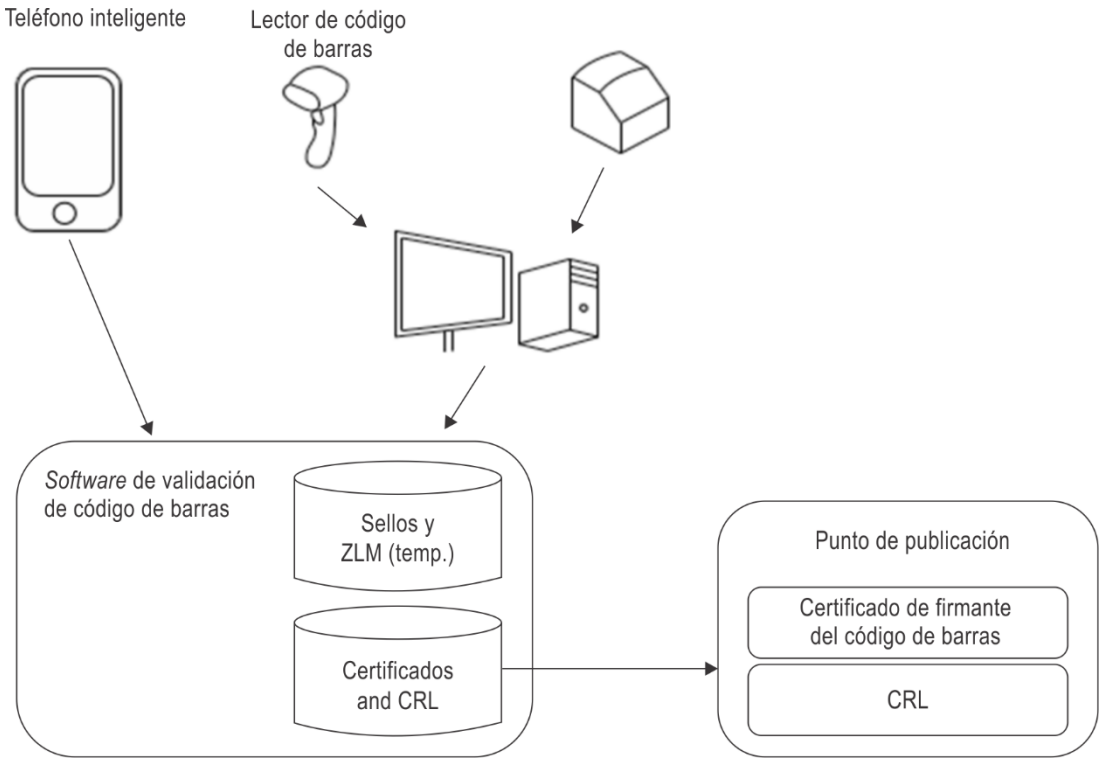

Figura 4. Validación del código de barras

En los contextos móviles, el *software* de validación también puede ejecutarse directamente en un teléfono inteligente. Mientras que la validez del sello puede verificarse mediante el *software* en el teléfono inteligente, la comparación entre los datos (firmados) dentro del sello y las ZLM impresas (p. ej., del visado o pasaporte) DEBE hacerse manualmente o mediante OCR de las ZLM a partir de la imagen capturada, aunque esto último suele ser un problema difícil en la práctica.

El *software* de validación del código de barras procesa los datos siguientes:

- datos de entrada proporcionados por dispositivos de lectura, p. ej., imágenes de visados o pasaportes; y
- certificados y CRL.

6.2 La PKI de autorización

En el caso de la PKI de autorización, el punto de confianza y la validación se gestionan de otra forma.

6.2.1 Validación de certificados verificables mediante tarjeta

Para los certificados DV y de terminal en la PKI de autorización, el punto de confianza es la clave pública más reciente de la CVCA del Estado que expidió el eMRTD. El punto de confianza inicial SE ALMACENARÁ de forma segura en el CI sin contacto del eMRTD en la fase de producción o de personalización (previa). A medida que cambia el par de claves utilizado por la CVCA, se producen certificados de enlace CVCA. El IC sin contacto del eMRTD DEBE actualizar internamente su punto o puntos de confianza de conformidad con los certificados de enlace válidos recibidos. Debido a la programación de los certificados de enlace CVCA, se almacenarán como máximo dos puntos de confianza CVCA en el CI sin contacto en cualquier momento dado.

Para validar un certificado de terminal, el IC sin contacto del eMRTD DEBE disponer de una cadena de certificados que comience en un punto de confianza almacenado en el IC sin contacto del eMRTD.

El procedimiento de validación de los certificados DV y de terminal es específico del protocolo de autenticación del terminal LDS2 y se detalla en el Doc 9303-11.

7. PERFILES DE CERTIFICADO Y DE CRL

Los perfiles de certificado se definen tanto para la PKI del eMRTD como la PKI de autorización.

7.1 La PKI del eMRTD

Los Estados expedidores u organizaciones expedidoras DEBEN expedir certificados y CRL que se ajusten a los perfiles especificados a continuación. Todos los certificados y CRL DEBEN producirse en formato de regla de codificación distinguida (DER) para conservar la integridad de las firmas en ellos. Los perfiles para certificados CSCA y DS que se incluyeron en la sexta edición de estas especificaciones difieren en algunos aspectos de los perfiles actuales. Los sistemas de inspección DEBEN ser capaces de tramitar certificados que se expidieron con arreglo a esos perfiles anteriores (véase el apéndice C), así como con los perfiles actuales.

Estos perfiles se basan en el requisito de que cada Estado expedidor u organización o entidad expedidora CREARÁ una CSCA única para la firma de todos los eMRTD que se ajusten al Doc 9303.

Los perfiles de certificado se definen en la sección 7.1 para los siguientes tipos de certificados:

- CA de firma de país;
- firmante del documento;
- firmante de la lista maestra CSCA;
- firmante de la lista de desviaciones; y
- comunicaciones, aunque no se necesita estrictamente en la actualidad, ya que se trata de una futura etapa de prueba. Estos certificados pueden utilizarse para acceder al PKD o para comunicaciones LDAP/EMAIL/HTTP entre Estados. Se recomienda que estos certificados sean expedidos por la CSCA.

En la sección 3 se definen los objetos CA de firma de país, firmante del documento y firmante de la lista maestra CSCA.

El perfil de CRL se define en la sección 4.

Los perfiles emplean la terminología siguiente para los requisitos de presencia de cada una de los componentes o extensiones:

m obligatorio — el campo DEBE estar presente;
x no utilizar — el campo NO DEBE estar presente;
o opcional — el campo PUEDE estar presente.
c condicional — el campo PODRÍA estar presente en determinadas condiciones.

Los perfiles emplean la terminología siguiente para los requisitos de criticidad de las extensiones que pueden o deben incluirse:

c crítico — las aplicaciones receptoras DEBEN poder procesar esta extensión;
nc no críticos — las aplicaciones receptoras que no comprendan esta extensión PUEDEN ignorarla.

Algunos de los requisitos identificados en estos perfiles son heredados de los perfiles de base de referencia (p. ej., RFC 5280). Para facilitar su consulta, el texto pertinente del perfil de base que abarca el requisito específico se reproduce en una tabla en el apéndice B.

7.1.1 Perfiles de certificado

En la tabla 5 se definen los requisitos de perfil comunes a todos los certificados para los campos del cuerpo del certificado. En la tabla 6 se definen los requisitos para las extensiones de certificado.

Tabla 5. Perfil de campos del certificado

Componente del certificado	*Presencia*	*Comentarios*
`Certificate`	m	
`TBSCertificate`	m	Véase la tabla 6.
`signatureAlgorithm`	m	El valor insertado aquí depende del algoritmo seleccionado.
`signatureValue`	m	El valor insertado aquí depende del algoritmo seleccionado.
`TBSCertificate`		
`version`	m	DEBE ser `v3`

Componente del certificado	*Presencia*	*Comentarios*
serialNumber	m	DEBE ser un entero positivo y un máximo de 20 octetos. DEBE utilizar codificación de complemento 2 y representarse con el menor número de octetos.
signature	m	El valor insertado aquí DEBE ser el mismo que en el componente signatureAlgorithm de la secuencia Certificate.
issuer	m	countryName y serialNumber, si están presentes, DEBEN ser PrintableString. Otros atributos que tienen sintaxis DirectoryString DEBEN ser PrintableString o UTF8String. countryName DEBE estar en mayúsculas. En 7.1.1.1 figuran las convenciones de nombres.
validity	m	DEBE terminar con Zulu (Z). El elemento segundos DEBE estar presente. Las fechas hasta 2049 DEBEN estar en UTCTime. UTCTime DEBE estar representado como AAMMDDHHMMSSZ. Las fechas en 2050 y posteriores DEBEN estar en GeneralizedTime. GeneralizedTime NO DEBE tener fracciones de segundo. GeneralizedTime DEBE estar representado como AAAAMMDDHHMMSSZ.
subject	m	countryName y serialNumber, si están presentes, DEBEN ser PrintableString. Otros atributos que tienen sintaxis DirectoryString DEBEN ser PrintableString o UTF8String. countryName DEBE estar en mayúsculas. Los countryName en los campos issuer y subject DEBEN corresponderse. En 7.1.1.1 figuran las convenciones de nombres.
subjectPublicKeyInfo	m	
issuerUniqueID	x	
subjectUniqueID	x	
extensions	m	Véase la tabla 6 sobre las extensiones que DEBERÍAN estar presentes. Los valores por defecto para las extensiones NO DEBEN codificarse.

Tabla 6. Perfil de extensiones del certificado

Nombre de la extensión	***Raíz autofirmada CSCA***		***Enlace CSCA***		***Firmante del documento***		***Firmante de la lista maestra y firmante de la lista de desviaciones***		***Comunicación***		***Comentarios***
	Presencia	***Criticidad***	***Presencia***	***Criticidad***	***Presencia***	***Criticidad***	***Presencia***	***Criticidad***	***Presencia***	***Criticidad***	
`AuthorityKeyIdentifier`	**o**	**nc**	**m**	**nc**	**m**	**nc**	**m**	**nc**	**m**	**nc**	
`keyIdentifier`	m		m		m		m		m		
`authorityCertIssuer`	o		o		o		o		o		
`authorityCertSerialNumber`	o		o		o		o		o		
`SubjectKeyIdentifier`	**m**	**nc**	**m**	**nc**	**o**	**nc**	**o**	**nc**	**o**	**nc**	
`subjectKeyIdentifier`	m		m		m		m		m		
`KeyUsage`	**m**	**c**	**m**	**c**	**m**	**c**	**m**	**c**	**m**	**c**	
`digitalSignature`	x		x		m		m		o		Algunos certificados de comunicación (p. ej., certificados TLS) exigen que los bits de `keyUsage` se establezcan con arreglo al conjunto de cifrado particular que se utilice. Algunos conjuntos de cifrado exigen que se establezca el bit `digitalSignature` y otros no lo exigen.
`nonRepudiation`	x		x		x		x		x		
`keyEncipherment`	x		x		x		x		o		
`dataEncipherment`	x		x		x		x		x		
`keyAgreement`	x		x		x		x		o		
`keyCertSign`	m		m		x		x		x		
`cRLSign`	m		m		x		x		x		
`encipherOnly`	x		x		x		x		x		
`decipherOnly`	x		x		x		x		x		
`PrivateKeyUsagePeriod`	**m**	**nc**	**m**	**nc**	**m**	**nc**	**o**	**nc**	**o**	**nc**	
`notBefore`	o		o		o		o		o		Por lo menos un `notBefore` o `notAfter` DEBE estar presente. DEBE codificarse como `generalizedTime`.
`notAfter`	o		o		o		o		o		

Nombre de la extensión	*Raíz autofirmada CSCA*		*Enlace CSCA*		*Firmante del documento*		*Firmante de la lista maestra y firmante de la lista de desviaciones*		*Comunicación*		*Comentarios*
CertificatePolicies	**o**	**nc**	**o**	**nc**	**o**	**nc**	**o**	**nc**	**o**	**nc**	
PolicyInformation	m		m		m		m		m		
policyIdentifier	m		m		m		m		m		
policyQualifiers	o		o		o		o		o		
PolicyMappings	**x**		**x**		**x**		**x**		**x**		Véase la nota 1.
SubjectAltName	**m**	**nc**	**m**	**nc**	**m**	**nc**	**m**	**nc**	**m**	**nc**	Véase 7.1.1.2.
IssuerAltName	**m**	**nc**	**m**	**nc**	**m**	**nc**	**m**	**nc**	**m**	**nc**	Véase 7.1.1.2.
SubjectDirectoryAttributes	**x**		**x**		**x**		**x**		**x**		
Basic Constraints	**m**	**c**	**m**	**c**	**x**		**x**		**x**		
cA	m		m		x		x		x		
PathLenConstraint	m		m		x		x		x		DEBE ser siempre '0'.
NameConstraints	**x**		**x**		**x**		**x**		**x**		Véase la nota 1.
PolicyConstraints	**x**		**x**		**x**		**x**		**x**		Véase la nota 1.
ExtKeyUsage	**x**		**x**		**x**		**m**	**c**	**m**	**c**	Véase 7.1.1.3.
CRLDistributionPoints	**m**	**nc**	**m**	**nc**	**m**	**nc**	**m**	**nc**	**o**	**nc**	
distributionPoint	m		m		m		m		m		DEBE ser el ldap, http o https. Véase 7.1.1.4.
reasons	x		x		x		x		x		
cRLIssuer	x		x		x		x		x		
InhibitAnyPolicy	**x**		**x**		**x**		**x**		**x**		Véase la nota 1.
FreshestCRL	**x**		**x**		**x**		**x**		**x**		Véase la nota 2.
privateInternetExtensions	**o**	**nc**	**o**	**nc**	**o**	**nc**	**o**	**nc**	**o**	**nc**	Véase la nota 3.
NameChange	**o**	**nc**	**o**	**nc**	**x**		**x**		**x**		Véase 7.1.1.5.
DocumentType	**x**		**x**		**m**	**nc**	**x**		**x**		Véase 7.1.1.6.
Netscape Certificate Type	**x**		**x**		**x**		**x**		**x**		Véase la nota 4.
other private extensions	**o**	**nc**	**o**	**nc**	**o**	**nc**	**o**	**nc**	**o**	**nc**	

Nota 1.— Por definición, la extensión solo puede aparecer en certificados CA intermedios (certificados expedidos por una CA a otra CA). Los certificados CA intermedios no se utilizan en la PKI del eMRTD. Por consiguiente, esta extensión está prohibida en los certificados eMRTD.

Nota 2.— La extensión de CRL más reciente se utiliza para indicar una CRL delta. Las CRL delta no se admiten en la PKI del eMRTD. Por consiguiente, esta extensión está prohibida.

Nota 3.— Hay dos extensiones Internet privadas (Authority Information Access y Subject Information Access) que se definen en RFC 5280 y se utilizan para indicar información sobre el expedidor o el sujeto de un certificado. Estas extensiones no se requieren en la PKI del eMRTD. No obstante, dado que no inciden en la interoperabilidad y no son críticas, pueden incluirse en forma opcional en los certificados eMRTD.

Nota 4.— La extensión de tipo de certificado Netscape puede utilizarse para limitar las finalidades para las cuales puede utilizarse un certificado. Las extensiones `extKeyUsage` y `basicConstraints` son ahora las extensiones estándar para esos fines y se utilizan en la aplicación del eMRTD. Debido al posible conflicto entre valores en las extensiones estándar y en la extensión propia Netscape, se prohíbe esta última.

7.1.1.1 Requisitos relativos a los campos Expedidor y Sujeto

Los campos Expedidor y Sujeto son comunes a todos los certificados, pero en los certificados de firmante LDS2 se aplican restricciones específicas.

7.1.1.1.1 Requisitos generales

SE EXIGEN las siguientes convenciones de nombres y direcciones para los campos `Issuer` y `Subject`.

- `countryName`. DEBE estar presente. Este valor contiene un código de país que DEBE seguir el formato de código de país de dos letras especificado en el Doc 9303-3.
- `commonName`. DEBE estar presente.

También PUEDEN incluirse otros atributos a discreción del Estado expedidor u organización expedidora.

7.1.1.1.2 Requisitos relativos a los certificados de firmante LDS2

Los certificados de firmante LDS2 DEBEN cumplir con el perfil de certificado de firmante del documento definido anteriormente, con las excepciones definidas en 7.1.2.

7.1.1.2 Requisitos relativos a los nombres alternativos de Expedidor y Sujeto

Debido a que las funciones servidas por nombres alternativos en la aplicación del eMRTD son específicas de esta aplicación y diferentes de las definidas para la PKI Internet en [RFC 5280], los valores en la extensión de nombres alternativos de sujeto de los certificados eMRTD no identifican por lo general en forma clara el sujeto del certificado.

En la aplicación del eMRTD, los nombres alternativos sirven las dos funciones siguientes.

La primera función es proporcionar información de contacto para el sujeto o el expedidor del certificado. Para ese fin DEBERÍA incluir por lo menos uno de los siguientes:

- `rfc822Name`;
- `dNSName`; o
- `uniformResourceIdentifier`.

La segunda función consiste en proporcionar una cadena de directorio integrada por códigos de país asignados por la OACI. Para este fin, los certificados expedidos utilizando este perfil DEBEN incluir además un nombre de directorio que se construye como sigue:

- `localityName` que contiene el código de país de la OACI como aparece en la ZLM; y
- si este código de país no define unívocamente el Estado expedidor u organización expedidora, el atributo `stateOrProvinceName` SE UTILIZARÁ para indicar el código de tres letras asignado por la OACI para el Estado expedidor u organización expedidora.
- No se permiten otros atributos.

En los certificados raíz autofirmados CSCA, las extensiones `IssuerAltName` y `SubjectAltName` DEBEN ser idénticas. En los certificados de enlace CSCA, los valores PUEDEN ser diferentes. Por ejemplo, si ha ocurrido un cambio en el `rfc822Name` de la CSCA inmediatamente antes de la expedición del certificado de enlace CSCA, la extensión `IssuerAltName` contendría el `rfc822Name` antiguo y la extensión `SubjectAltName` contendría el nuevo `rfc822Name`. Todo certificado de enlace CSCA subsiguiente contendría el nuevo `rfc822Name` en ambas extensiones.

7.1.1.3 Requisitos relativos a la extensión de uso de clave ampliado

El identificador de objeto (OID) que debe incluirse en la extensión `extendedKeyUsage` para los certificados de firmante de la lista maestra es `2.23.136.1.1.3`.

El identificador de objeto (OID) que debe incluirse en la extensión `extendedKeyUsage` para los certificados de firmante de la lista de desviaciones es `2.23.136.1.1.8`.

Para los certificados de comunicación, el valor de esta extensión depende del protocolo de comunicación que se utilice (véase RFC 5280, sección 4.2.1.12).

7.1.1.4 Requisitos relativos a la extensión de puntos de distribución de la CRL

Las CSCA pueden publicar sus CRL en varios lugares, incluyendo el PKD, su propio sitio web, etc.

Para las CRL que se publiquen en lugares distintos del PKD (p. ej., sitio web o servidor LDAP local), los valores que han de incluirse en esta extensión son controlados por la CSCA que expide los certificados y las CRL en cuestión.

Para las CRL presentadas al PKD, los participantes en este directorio PUEDEN incluir dos valores URL para su CRL utilizando la plantilla siguiente (sustitúyase “CountryCode” por el código de tres letras de Estado expedidor u organización expedidora asignado por la OACI). Si este código de país no identifica unívocamente al Estado expedidor u organización expedidora, se creará la anotación adjuntando el símbolo “_” al código de país de tres letras en la ZLM, y luego el código de tres letras asignado por la OACI al Estado expedidor u organización expedidora que identifique unívocamente dicho Estado u organización:

https://pkddownload1.icao.int/CRLs/CountryCode.crl
https://pkddownload2.icao.int/CRLs/CountryCode.crl

Esta es una extensión obligatoria y la verificación del estado de revocación es una parte obligatoria del procedimiento de validación. Por consiguiente, DEBE rellenarse por lo menos un valor:

- Los valores del PKD pueden ser los únicos valores en la extensión;
- Puede haber valores adicionales (p. ej., una CSCA puede también optar por publicar sus CRL en un sitio web e incluir una referencia a esa fuente); o
- La CSCA también puede optar por influir solo un único valor (p. ej., una referencia a su sitio web como fuente), incluso si también presenta sus CRL al PKD.

En los siguientes ejemplos se ilustran los valores del PKD que se rellenarían en los certificados expedidos por la autoridad expedidora en los casos de Singapur y Hong Kong:

Ejemplo PKD en el caso de Singapur:

https://pkddownload1.icao.int/CRLs/SGP.crl
https://pkddownload2.icao.int/CRLs/SGP.crl

Ejemplo PKD en el caso de Hong Kong:
https://pkddownload1.icao.int/CRLs/CHN_HKG.crl
https://pkddownload2.icao.int/CRLs/CHN_HKG.crl

7.1.1.5 Extensión de cambio de nombre

Cuando tiene lugar una renovación de clave CSCA, DEBE expedirse un certificado que enlace la clave pública antigua con la nueva clave pública para proporcionar una transición segura a las partes que confían. En general, esto se logra mediante la expedición de un certificado autoexpedido donde los campos `issuer` y `subject` son idénticos pero la clave utilizada para verificar la firma representa el antiguo par de claves y la clave pública certificada representa el nuevo par de claves.

SE RECOMIENDA que las CSCA no cambien su nombre distinguido (DN) innecesariamente dado que ello tiene consecuencias adversas para las partes que confían (deben conservar tanto el antiguo nombre como el nuevo como CSCA válidas para el mismo Estado expedidor u organización expedidora hasta que todos los eMRP firmados con el nombre antiguo hayan caducado). No obstante, si es necesario efectuar un cambio de nombre, este DEBE comunicarse a las partes que confían mediante la expedición de un certificado de enlace CSCA donde el campo `issuer` contiene el nombre antiguo y el campo `subject` contiene el nuevo nombre. Este certificado de enlace CSCA también contiene una renovación de clave donde la clave utilizada para verificar la firma representa el antiguo para de claves y la clave pública certificada representa el nuevo par de claves. Los certificados que contienen un cambio de nombre de CSCA y una renovación de clave para dicha CSCA DEBEN incluir la extensión `NameChange` para identificar ese tipo de certificado. Esto no tiene consecuencias para `PathLengthConstraint`; el valor sigue siendo `'0'`.

Además, la extensión `NameChange` también PUEDE incluirse en el nuevo certificado autofirmado CSCA creado tras el cambio del DN de la CSCA. En un certificado raíz CSCA autofirmado de ese tipo, tanto el campo `issuer` como el `subject` contienen el nuevo DN. A diferencia del certificado de enlace autofirmado CSCA que contienen el DN antiguo y el nuevo de la CSCA, la inclusión de la extensión `NameChange` en un certificado raíz autofirmado CSCA indica simplemente que ha ocurrido un cambio de nombre y no enlaza el DN antiguo con el nuevo.

Una CSCA NO DEBE volver a utilizar los números de serie del certificado. Cada certificado expedido por una CSCA, independientemente de si la autoridad ha experimentado un cambio de nombre o no, DEBE ser único.

Sintaxis ASN.1 para la extensión de cambio de nombre:

```
nameChange EXTENSION ::= {
    SYNTAX NULL
    IDENTIFIED BY id-icao-mrtd-security-extensions-nameChange}

id-icao-mrtd-security-extensions OBJECT IDENTIFIER ::= {id-icao-
mrtd-security 6}
id-icao-mrtd-security-extensions-nameChange OBJECT IDENTIFIER ::=
{id-icao-
mrtd-security-extensions 1}
```

7.1.1.6 Extensión de tipo de documento

La extensión `DocumentType` DEBE utilizarse para indicar los tipos de documento, como aparecen en la ZLM, que el correspondiente firmante del documento está autorizado a producir. Esta extensión siempre DEBE establecerse como no crítica.

Sintaxis ASN.1 para la extensión de lista de tipo de documento:

```
documentTypeList EXTENSION ::= {
    SYNTAX DocumentTypeListSyntax
    IDENTIFIED BY id-icao-mrtd-security-extensions-documentTypeList}

DocumentTypeListSyntax ::= SEQUENCE {
    version             DocumentTypeListVersion,
    docTypeList   SET OF DocumentType }

DocumentTypeListVersion ::= INTEGER {v0(0)}

-- Document Type as contained in MRZ, e.g. "P" or "ID" where a
-- single letter denotes all document types starting with that letter
DocumentType ::= PrintableString(SIZE(1..2))

id-icao-mrtd-security-extensions-documentTypeList OBJECT
IDENTIFIER ::= {id-icao-mrtd-security-extensions 2}
```

7.1.2 Perfil de certificado de firmante LDS2

Los certificados de firmante LDS2 DEBEN cumplir con el perfil de certificado de firmante del documento definido en 7.1.1, con las excepciones siguientes:

Campo Sujeto:
El campo de sujeto de los certificados de firmante LDS2 DEBE rellenarse como sigue:

- `countryName`: DEBE estar presente. Este valor contiene un código de país que DEBE seguir el formato de código de país de dos letras especificado en el Doc 9303-3.
- `commonName`: DEBE estar presente. El valor de este atributo NO DEBE superar la longitud de 9 caracteres.
- NO DEBEN incluirse otros atributos.

Extensiones de certificado:
Los certificados de firmante LDS2 DEBEN contener las extensiones de certificado que se indican en la tabla 7. NO DEBEN incluirse todas las demás extensiones de certificado.

Tabla 7. Extensiones de certificado obligatorias para LDS2

Nombre de la extensión	Firmante LDS2		Comentarios
	Presencia	Criticidad	
AuthorityKeyIdentifier	**m**	**nc**	
keyIdentifier	m		
authorityCertIssuer	o		
authorityCertSerialNumber	o		
ExtKeyUsage	**m**	**c**	Véase la nota 1.

Nota 1.— La extensión EKU de cada certificado de firmante LDS2 DEBE rellenarse como se indica a continuación. Obsérvese que puede autorizarse que un único firmante LDS2 firme varios tipos de objetos de datos LDS2. En ese caso, la extensión EKU contendría todos los OID pertinentes para ese firmante:

```
id-icao-mrtd-security-lds2 OBJECT IDENTIFIER ::= {id-icao-mrtd-security 9}
id-icao-lds2Signer OBJECT IDENTIFIER ::= {id-icao-mrtd-
security-lds2 8}
```

- LDS2 Travel Stamp Signer (LDS2-TS) certificates
  id-icao-tsSigner OBJECT IDENTIFIER ::= { id-icao-lds2Signer 1}

- LDS2 Visa Signer (LDS2-V) certificates:
  id-icao-vSigner OBJECT IDENTIFIER ::= { id-icao-lds2Signer 2}

- LDS2 Biometrics Signer (LDS2-B) certificates:
  id-icao-bSigner OBJECT IDENTIFIER ::= { id-icao-lds2Signer 3}

Nota 2.— Los certificados de firmante LDS2 deben cumplir con las restricciones de tamaño impuestas por EF.Certificates que se indican en el Doc 9303-10.

A pesar de que en estos certificados no se incluye la extensión de puntos de distribución de la CRL, es obligatorio que se compruebe el estado de revocación de cada certificado como parte del proceso habitual de validación. La CRL expedida por la CSCA que expidió el certificado en cuestión es la CRL utilizada para verificar su estado de revocación.

7.1.3 Perfil de certificado de firmante del código de barras

Los certificados de firmante del código de barras DEBEN cumplir con el perfil de certificado de firmante LDS2. Dado que los certificados de firmante del código de barras tienen una función distinta a la de los certificados LDS2, su perfil se desvía en algunos aspectos. En concreto, hay requisitos específicos para el subjectDN del certificado de firmante del código de barras y el número de serie (véase el Doc 9303-13).

Campo Sujeto:
El campo de sujeto de los certificados de firmante del código de barras DEBE rellenarse como sigue:

- `commonName`: DEBE estar presente. DEBE constar de dos caracteres en mayúsculas, en formato printableString, que definan unívocamente al firmante del código de barras de un país, y DEBE coincidir con las letras 3 y 4 del identificador de firmante en el código de barras, como se especifica en el Doc 9303-13.
- `countryName`: DEBE constar del código de país de dos letras (véase el Doc 9303-3) del firmante del código de barras, en mayúsculas y en formato printableString, y DEBE coincidir con las letras 1 y 2 del identificador de firmante en el código de barras, como se especifica en el Doc 9303-13.
- NO DEBEN incluirse otros atributos.

Extensiones de certificado:
Los certificados de firmante del código de barras DEBEN contener las extensiones de certificado que se indican en la tabla 8. NO DEBEN incluirse todas las demás extensiones de certificado.

Tabla 8. Extensiones permitidas para certificados de firmante del código de barras

Nombre de la extensión	Firmante LDS2		Comentarios
	Presencia	**Criticidad**	
AuthorityKeyIdentifier	**m**	**nc**	
keyIdentifier	m		
authorityCertIssuer	o		
authorityCertSerialNumber	o		
DocumentType	**o**		Esta extensión indica el tipo de documento que el firmante del código de barras está autorizado a producir.
ExtKeyUsage	**m**	**c**	Véase la nota.

Nota 1.— La extensión EKU de cada certificado de firmante del código de barras DEBE rellenarse como se indica a continuación.

```
id-icao-mrtd-security-vds OBJECT IDENTIFIER ::= {id-icao-mrtd-security 11}
id-icao-vdsSigner OBJECT IDENTIFIER ::= {id-icao-mrtd-security-vds 1}
```

7.1.4 Perfil de CRL

En la tabla 9 se definen los requisitos de perfil de CRL para los campos en el cuerpo de la CRL. En la tabla 10 se definen los requisitos de perfil de la CRL para las extensiones de CRL y CRL Entry.

Tabla 9. Perfil de campos de la CRL

Componente de la lista de certificados	***CRL de CSCA***	***Comentarios***
`CertificateList`	m	
`tBSCertList`	m	Véase la tabla 10.
`signatureAlgorithm`	m	El valor insertado aquí depende del algoritmo seleccionado.
`signatureValue`	m	El valor insertado aquí depende del algoritmo seleccionado.
`tBSCertList`		
`Version`	m	DEBE ser `v2`.
`Signature`	m	El valor insertado aquí DEBE ser el mismo que en el componente `signatureAlgorithm` de la secuencia `CertificateList`.
`Issuer`	m	`countryName` y `serialNumber`, si están presentes, DEBEN ser `PrintableString`. Otros atributos que tienen sintaxis `DirectoryString` DEBEN ser `PrintableString` o `UTF8String`. `countryName` DEBE estar en mayúsculas.
`thisUpdate`	m	DEBE terminar con Zulu (Z). El elemento segundos DEBE estar presente. Las fechas hasta 2049 DEBEN estar en `UTCTime`. `UTCTime` DEBE estar representado como AAMMDDHHMMSSZ. Las fechas en 2050 y posteriores DEBEN estar en `GeneralizedTime`. `GeneralizedTime` NO DEBE tener fracciones de segundo. `GeneralizedTime` DEBE estar representado como AAAAMMDDHHMMSSZ.
`nextUpdate`	m	DEBE terminar con Zulu (Z). El elemento segundos DEBE estar presente. Las fechas hasta 2049 DEBEN estar en `UTCTime`. `UTCTime` DEBE estar representado como AAMMDDHHMMSSZ. Las fechas en 2050 y posteriores DEBEN estar en `GeneralizedTime`. `GeneralizedTime` NO DEBE tener fracciones de segundo. `GeneralizedTime` DEBE estar representado como AAAAMMDDHHMMSSZ.
`revokedCertificates`	c	DEBE estar presente si hay certificados revocados. NO DEBE estar presente si no hay certificados revocados. Si está presente, NO DEBE estar vacío.

Componente de la lista de certificados	*CRL de CSCA*	*Comentarios*
crlExtensions	m	Véase la tabla 10 sobre las extensiones que DEBERÍAN estar presentes. Los valores por defecto para las extensiones NO DEBEN codificarse.

Tabla 10. Perfil de extensiones de CRL y de entrada de CRL

Nombre de la extensión	*CRL de CSCA*	*Criticidad*	*Comentarios*
Extensiones de CRL			
authorityKeyIdentifier	m	**nc**	Este DEBE ser el mismo valor que el campo subjectKeyIdentifier en el certificado de expedidor de la CRL.
keyIdentifier	m		
authorityCertIssuer	o		
authorityCertSerialNumber	o		
issuerAlternativeName	o	**nc**	Véase la nota 1.
cRLNumber	m	**nc**	DEBE ser un entero no negativo y un máximo de 20 octetos. DEBE utilizar codificación de complemento 2 y representarse con el menor número de octetos.
deltaCRLIndicator	x		
issuingDistributionPoint	x		
freshestCRL	x		
CRL Entry Extensions			
reasonCode	x		
holdInstructionCode	x		
invalidityDate	x		
certificateIssuer	x		
other private extensions	o	**nc**	

Nota 1.— Si una CSCA ha experimentado un cambio de nombre, esta extensión PUEDE incluirse en las CRL expedidas después de dicho cambio de nombre. Si están presentes, los valores de esta extensión DEBEN ser idénticos al campo `issuer` *de los certificados expedidos por la CSCA con el nombre anterior. Una vez que todos los certificados expedidos con un nombre de CSCA anterior al expirado, el nombre de la CSCA puede excluirse de las CRL subsiguientes. No se exige que los sistemas de inspección procesen esta extensión. Dado que en el Doc 9303 de la OACI se exige una única CSCA por país, el componente* `countryName` *del campo de expedidor es suficiente para identificar unívocamente la CSCA. La clave pública más reciente de dicha CSCA se utiliza para verificar la firma de la CRL. Dado que una CSCA expide una única CRL, esta CRL abarca todos los certificados expedidos con ese* `countryName`*. Además de esa verificación obligatoria, también PUEDE realizarse una verificación opcional de que el campo* `issuer` *del certificado es igual al campo* `issuer` *de la CRL o uno de los valores de la extensión* `issuerAltName` *de la CRL.*

Nota 2.— Es posible que la CRL contenga otra información sobre revocación, por ejemplo, relativa a certificados de operador del sistema o autoridad de registro.

7.2 La PKI de autorización

La PKI de autorización incluye certificados X.509 para el SPOC y certificados verificables mediante tarjeta para la CVCA, el DV y los terminales. En esta sección se especifican los perfiles de certificado SPOC, CVCA, DV e IS. Se ofrece una visión general de los objetos de datos contenidos en los certificados verificables mediante tarjeta, así como la codificación de dichos objetos.

7.2.1 Perfil de certificado SPOC

Se puede utilizar una configuración de CA separada para expedir directamente certificados SPOC con las siguientes restricciones al perfil de certificado CA autofirmado:

- El certificado CA DEBE ser conforme a [RFC 5280];
- Los algoritmos SHA-224, SHA-256, SHA-384 y SHA-512 son los únicos algoritmos de condensación permitidos; y
- `countryName` DEBE estar presente en el campo Sujeto.

Los certificados SPOC (cliente y servidor) LDS2 DEBEN cumplir con el perfil de certificado de comunicación definido en la sección 7.1, con las restricciones siguientes.

Campo Expedidor:
Los certificados SPOC son expedidos por la CSCA o por una CA independiente creada específicamente para expedir certificados SPOC.

Campo Sujeto:
En el caso de los certificados SPOC LDS2, el campo Sujeto DEBE rellenarse como sigue:

- `countryName`: DEBE estar presente. Este valor contiene un código de país que DEBE seguir el formato de código de país de dos letras especificado en el Doc 9303-3.
- `commonName`: DEBE estar presente. En el caso de los certificados de cliente TLS del SPOC, el valor DEBERÍA ser “SPOC TLS client”. En el caso de los certificados de servidor TLS del SPOC, el valor DEBERÍA SER “SPOC TLS server”.
- También PUEDEN incluirse otros atributos a discreción del Estado expedidor u organización expedidora.

Extensiones de uso de clave
En el caso de los certificados SPOC, el valor (o valores) depende del conjunto de cifrado utilizado.

Extensiones de nombres alternativos de sujeto
Además de los valores indicados en el perfil de certificado de comunicación, los certificados de servidor TLS del SPOC también DEBEN contener un valor dNSName que es la parte principal de la URL del SPOC.

Extensiones de uso de clave ampliado
En el caso de los certificados de cliente y servidor del SPOC, DEBE incluirse el valor pertinente que se indica a continuación.

- Certificados de cliente SPOC: El OID es 2.23.136.1.1.10.1;
- Certificados de servidor SPOC: El OID es 2.23.136.1.1.10.2.

Extensiones de puntos de distribución de la CRL
Esta extensión es obligatoria en los certificados de cliente y servidor del SPOC.

7.2.2 Perfiles de certificado CVCA, DV y de terminal

Los certificados de enlace CVCA, certificados DV y certificados de terminal son validados por el CI. Debido a las restricciones computacionales de esos chips, los certificados DEBEN estar tener un formato verificable mediante tarjeta (certificados CV).

SE UTILIZARÁN el formato y perfil de certificado indicados en la tabla 11. En el Doc 9303-11 figura información detallada sobre los valores de codificación.

Tabla 11. Perfil de certificado CV

Objeto de datos	Presencia en el certificado
Certificado CV	m
Cuerpo del certificado	m
Identificador del perfil de certificado	m
Referencia de la autoridad de certificación	m
Clave pública	m
Referencia de la persona titular de certificado	m
Plantilla de autorización de la persona titular de certificado	m
Fecha de entrada en vigor del certificado	m
Fecha de caducidad del certificado	m
Extensiones de certificado	o
Firma	m

7.2.2.1 Identificador del perfil de certificado

El identificador del perfil de certificado indica la versión del perfil. SE UTILIZARÁ la versión 1, que se identifica con el valor de 0.

7.2.2.2 Referencia de la autoridad de certificación y referencia de la persona titular de certificado

Cada certificado CV DEBE contener dos referencias de clave pública (una referencia de la persona titular de certificado y una referencia de la autoridad de certificación).

La referencia de la autoridad de certificación es una referencia a la clave pública (externa) de la autoridad de certificación (CVCA o DV) que SE UTILIZARÁ para verificar la firma del certificado.

La referencia de la persona titular de certificado es un identificador de la clave pública proporcionada en el certificado que SE UTILIZARÁ para hacer referencia a esta clave pública.

Nota.— En consecuencia, la referencia de la autoridad de certificación contenida en un certificado DEBE ser igual a la referencia de la persona titular de certificado que figura en el certificado correspondiente de la autoridad de certificación expedidora.

La referencia de la persona titular de certificado DEBE constar de los elementos concatenados siguientes: Código de país, secuencia mnemónica de la persona titular y número de secuencia. Estos elementos DEBEN elegirse de conformidad con la tabla 12 y las normas siguientes:

a) Código de país:

- El código de país SERÁ el código de dos letras que figura en el Doc 9303-3 del país de la persona titular de certificado.

b) Secuencia mnemónica de la persona titular:

- La secuencia mnemónica de la persona titular SE ASIGNARÁ como un identificador único de la siguiente manera:
  - la secuencia mnemónica de la persona titular de una CVCA DEBE ser asignada por la propia CVCA;
  - la secuencia mnemónica de la persona titular de un DV DEBE ser asignada por la CVCA nacional; y
  - la secuencia mnemónica de la persona titular de un IS DEBE ser asignada por el DV supervisor.

c) Número de secuencia:

- El número de secuencia DEBE ser asignado por la persona titular de certificado;
- El número de secuencia DEBE ser numérico o alfanumérico;
  - Un número de secuencia numérico CONSTARÁ de los caracteres “0…9”.
  - Un número de secuencia alfanumérico CONSTARÁ de los caracteres “0…9” y “A…Z”.
- El número de secuencia DEBE empezar con el código de país de dos letras que figura en el Doc 9303-3 de la autoridad de certificación, los otros tres caracteres SE ASIGNARÁN como un número de secuencia alfanumérico; y
- El número de secuencia PUEDE restablecerse si se agotan todos los números de secuencia disponibles.

Tabla 12. Referencia de la persona titular de certificado

	Codificación	Longitud
Código de país	Doc 9303-3	2F
Secuencia mnemónica de la persona titular	ISO/IEC 8859-1	9V
Número de secuencia:	ISO/IEC 8859-1	5F

7.2.2.3 Clave pública

Este campo contiene la clave pública que se está certificando.

Los certificados autofirmados CSCA DEBEN contener parámetros de dominio. Los certificados de enlace CVCA PUEDEN contener parámetros de dominio, salvo en el caso de que dichos parámetros hayan cambiado. En ese caso, los certificados de enlace DEBEN contener los nuevos parámetros de dominio.

Los certificados DV y de terminal NO DEBEN contener parámetros de dominio. Los parámetros de dominio de las claves públicas de DV y terminal SE HEREDARÁN de la clave pública CVCA respectiva.

7.2.2.4 Plantilla de autorización de la persona titular de certificado

La función y la autorización de la persona titular de certificado SE CODIFICARÁN en la plantilla de autorización de la persona titular del certificado. Esta plantilla es una secuencia que consta de los objetos de datos siguientes:

a) un identificador de objeto que especifica el tipo de terminal y el formato de la plantilla; y

b) un objeto de datos discrecionales que codifica la autorización relativa, es decir, la función y autorización de la persona titular del certificado con respecto a la autoridad de certificación.

Los valores específicos se definen en el Doc 9303-10.

7.2.2.5 Fecha de entrada en vigor del certificado y fecha de caducidad del certificado

La combinación de estas dos fechas indica el período de validez del certificado. La fecha de entrada en vigor del certificado DEBE ser la fecha en que se genera. La fecha de caducidad del certificado es la fecha tras la cual caduca.

7.2.2.6 Extensiones de certificado (extensiones de autorización)

Las extensiones de autorización PUEDEN incluirse en los certificados CVCA, DV y de terminal. Estas extensiones transmiten autorizaciones adicionales a las de la plantilla de autorización de la persona titular de certificado.

Una extensión de autorización es una secuencia de plantillas de datos discrecionales, donde cada plantilla de datos discrecionales DEBE contener una secuencia de los objetos de datos siguientes, que también se muestran en la tabla 13:

a) un identificador de objeto que especifica el contenido y el formato de la extensión; y

b) un objeto de datos específico del contexto que contiene la autorización codificada.

Tabla 13. Extensiones de certificado

Objeto de datos
Extensiones de certificado
Plantilla de datos discrecionales
Identificador de objeto
Objeto de datos específico del contexto
Plantilla de datos discrecionales
Identificador de objeto
Objeto de datos específico del contexto
…

Nota.— El procedimiento de validación de certificados que se describe en el Doc 9303-11 no tiene en cuenta las extensiones de certificado. Por lo tanto, las extensiones son atributos no críticos y el CI NO DEBE rechazar certificados debido a extensiones desconocidas.

7.2.2.7 Firma

La firma del certificado SE CREARÁ sobre el cuerpo del certificado codificado (es decir, incluyendo la etiqueta y la longitud).

La referencia de la autoridad de certificación IDENTIFICARÁ la clave pública que se utilizará para verificar la firma.

7.2.3 Objetos de datos

En la tabla 14 se presenta un resumen de las etiquetas, longitudes y valores de los objetos de datos utilizados en los certificados CVCA, DV y de terminal.

Tabla 14. Resumen de los objetos de datos (clasificados por etiqueta)

Nombre	Etiqueta	Long.	Valor	Comentario
Identificador de objeto	0x06	V	Identificador de objeto	–
Referencia de la autoridad de certificación	0x42	16V	Cadena de caracteres	Identifica la clave pública de la autoridad de certificación expedidora de un certificado.
Plantilla de datos discrecionales	0x53	V	Cadena de octetos	Contiene datos arbitrarios.
Referencia de la persona titular de certificado	0x5F20	16V	Cadena de caracteres	Asocia la clave pública contenida en un certificado con un identificador.

Nombre	Etiqueta	Long.	Valor	Comentario
Fecha de caducidad del certificado	0x5F24	6F	Fecha	La fecha tras la cual caduca el certificado.
Fecha de entrada en vigor del certificado	0x5F25	6F	Fecha	La fecha en que se genera el certificado.
Identificador del perfil de certificado	0x5F29	1F	Entero sin signo	Versión del certificado y formato de solicitud de certificado.
Firma	0x5F37	V	Cadena de octetos	Firma digital producida por un algoritmo criptográfico asimétrico.
Extensiones de certificado	0x65	V	Secuencia	Anida extensiones de certificado.
Autenticación	0x67	V	Secuencia	Contiene objetos de datos relacionados con la autenticación.
Plantilla de datos discrecionales	0x73	V	Secuencia	Anida objetos de datos arbitrarios.
Certificado CV	0x7F21	V	Secuencia	Anida el cuerpo del certificado y la firma.
Clave pública	0x7F49	V	Secuencia	Anida el valor de la clave pública y los parámetros de dominio.
Plantilla de autorización de la persona titular de certificado	0x7F4C	V	Secuencia	Codifica la función de la persona titular de certificado (es decir, CVCA, DV, terminal) y asigna derechos de acceso de lectura/escritura.
Cuerpo del certificado	0x7F4E	V	Secuencia	Anida objetos de datos del cuerpo del certificado.

F: longitud fija (número exacto de octetos), V: longitud variable (hasta un determinado número de octetos).

7.2.3.1 Codificación de valores

Los tipos de valores básicos utilizados en esta especificación son los siguientes: enteros (sin signo), puntos de curva elíptica, fechas, cadenas de caracteres, cadenas de octetos, identificadores de objeto y secuencias.

7.2.3.1.1 Enteros sin signo

Todos los enteros utilizados en esta especificación son enteros sin signo. Un entero sin signo SE CONVERTIRÁ en una cadena de octetos utilizando la representación binaria del entero en formato big-endian. SE UTILIZARÁ el número mínimo de octetos, es decir, los octetos iniciales de valor 0x00 NO DEBEN utilizarse.

Nota.— En cambio, el ENTERO tipo ASN.1 es siempre un entero con signo.

7.2.3.1.2 Puntos de curva elíptica

La conversión de los puntos de curva elíptica en cadenas de octetos se especifica en [TR-03111]. SE UTILIZARÁ el formato sin comprimir.

7.2.3.1.3 Fechas

Una fecha se codifica en 6 dígitos “d1…d6” con el formato AAMMDD, utilizando la zona horaria GMT. Se convierte en una cadena de octetos “o1…o6” codificando cada dígito dj en un octeto oj como decimal codificado en binario (BCD) desempaquetado ($1 \leq j \leq 6$).

El año AA está codificado en dos dígitos y debe interpretarse como 20AA, es decir, que el año está en el intervalo de 2000 a 2099.

7.2.3.1.4 Cadenas de caracteres

Una cadena de caracteres “c1…cn” es una concatenación de n caracteres cj con $1 \leq j \leq n$. SE CONVERTIRÁ en una cadena de octetos “o1…on” convirtiendo cada carácter cj en un octeto oj utilizando el conjunto de caracteres ISO/IEC 8859-1.

Los códigos de caracteres 0x00-0x1F y 0x7F-0x9F no están asignados y NO DEBEN utilizarse. La conversión de un octeto a un carácter no asignado DARÁ error.

7.2.3.1.5 Cadenas de octetos

Una cadena de octetos “o1…on” es una concatenación de n octetos oj con $1 \leq j \leq n$. Cada octeto oj consta de 8 bits.

7.2.3.1.6 Identificadores de objetos

Un identificador de objeto “i1.i2…in” se codifica como una lista ordenada de n enteros sin signo ij con $1 \leq j \leq n$. SE CONVERTIRÁ en una cadena de octetos “o1…on−1” mediante el procedimiento siguiente:

1) Los dos primeros enteros i1 e i2 se empaquetan en un único entero i que posteriormente se convierte en la cadena de octetos o1. El valor se calcula como sigue:

$$i=i1 \cdot 40+i2$$

2) Los enteros restantes ij se convierten directamente en cadenas de octetos oj-1 con $3 \leq j \leq n$.
En [X.690] figuran más detalles sobre la codificación.

Nota.— Los enteros sin signo se codifican como cadenas de octetos utilizando el formato big-endian, como se describe en el Doc 9303-11, pero solo se utilizan los bits 1-7 de cada octeto. El bit 8 (el de más a la izquierda) establecido en uno se utiliza para indicar que este octeto no es el último de la cadena.

7.2.3.1.7 Secuencias

Una secuencia “D1…Dn” es una lista ordenada de n objetos de datos Dj con $1 \leq j \leq n$. La secuencia SE CONVERTIRÁ en una lista concatenada de cadenas de octetos“O1…On” mediante la codificación DER de cada objeto de datos Dj en una cadena de octetos Oj.

7.2.3.2 Codificación de objetos de datos de clave pública

Un objeto de datos de clave pública contiene una secuencia de un identificador de objeto y varios objetos de datos específicos del contexto:

- El identificador de objeto es específico de la aplicación y se refiere no solo al formato de la clave pública (es decir, los objetos de datos específicos del contexto) sino también a su uso.
- Los objetos de datos específicos del contexto están definidos por el identificador de objeto y contienen el valor de la clave pública y los parámetros de dominio.

A continuación se describe el formato de los objetos de datos de clave pública utilizados en esta especificación.

7.2.3.2.1 Claves públicas RSA

Los objetos de datos contenidos en una clave pública RSA se muestran en la tabla 15. El orden de los objetos de datos es fijo.

Tabla 15. Clave pública RSA

Objeto de datos	Abrev.	Etiqueta	Tipo	Certificado CV
Identificador de objeto		0x06	Identificador de objeto	m
Módulo compuesto	n	0x81	Entero sin signo	m
Exponente público	e	0x82	Entero sin signo	m

7.2.3.2.2 Claves públicas de curva elíptica

Los objetos de datos contenidos en una clave pública EC se muestran en la tabla 16. El orden de los objetos de datos es fijo, los parámetros de dominio CONDICIONALES DEBEN estar todos presentes, salvo el cofactor, o todos ausentes, como se define a continuación:

- Los certificados autofirmados CVCA DEBEN contener parámetros de dominio;
- Los certificados de enlace CVCA PUEDEN contener parámetros de dominio;
- Los certificados DV y de terminal NO DEBEN contener parámetros de dominio. Los parámetros de dominio de las claves públicas de DV y terminal SE HEREDARÁN de la clave pública CVCA respectiva; y
- Las solicitudes de certificado DEBEN contener siempre parámetros de dominio.

Tabla 16. Clave pública EC

Objeto de datos	Abrev.	Etiqueta	Tipo	Certificado CV
Identificador de objeto		0x06	Identificador de objeto	m
Módulo primo	p	0x81	Entero sin signo	c
Primer coeficiente	a	0x82	Entero sin signo	c
Segundo coeficiente	b	0x83	Entero sin signo	c
Punto de base	G	0x84	Punto de curva elíptica	c
Orden del punto	r	0x85	Entero sin signo	c
Punto público	Y	0x86	Punto de curva elíptica	m
Cofactor	f	0x87	Entero sin signo	c

8. PROTOCOLO DEL SPOC

El punto de contacto único (SPOC) es la única interfaz expuesta por un Estado para las operaciones de gestión de claves con Estados extranjeros para la PKI de autorización LDS2. El protocolo del SPOC es el protocolo de gestión de claves para la realización de operaciones entre CVCA y DV de diferentes Estados. Aunque el protocolo del SPOC PUEDE utilizarse también para las comunicaciones internas entre una CVCA y los DV a escala nacional y entre un DV y el conjunto de terminales nacionales que gestiona, no se trata de un requisito. Para la gestión de claves a escala nacional pueden utilizarse otros protocolos de gestión de claves.

El protocolo del SPOC se utiliza para intercambiar claves y certificados, a fin de que:

- un DV pueda enviar una solicitud de certificación a la CVCA extranjera;
- una CVCA pueda enviar el certificado expedido al DV solicitante;
- las CVCA y los DV puedan solicitar el conjunto de certificados válidos a una CVCA extranjera; y
- se puedan intercambiar mensajes generales entre los DV y las CVCA.

Dentro de un Estado:

- La CVCA UTILIZARÁ su SPOC nacional para aceptar las solicitudes de certificación extranjeras entrantes y para enviar los certificados resultantes o las notificaciones de fallo al solicitante;
- Los DV UTILIZARÁN su SPOC nacional para enviar las solicitudes de certificación a las CVCA extranjeras y para recibir los certificados resultantes o las notificaciones de fallo;
- El SPOC DEBE recopilar las solicitudes y respuestas de las CVCA y los DV nacionales y remitirlas al SPOC del Estado receptor; y
- El SPOC DEBE recopilar las solicitudes y respuestas de los SPOC de otros Estados y proporcionárselas a la CVCA/los DV nacionales correspondientes.

La comunicación del servicio web del SPOC UTILIZARÁ HTTPS con autenticación TLS tanto del cliente como del servidor.

Nota.— Los SPOC son centros de comunicación entre las entidades de la PKI de autorización que, por lo tanto, deben estar disponible de manera ininterrumpida las 24 horas del día y ser accesibles para los SPOC extranjeros.

Cada SPOC se registra por separado con los demás SPOC de interés, proporcionando al menos la siguiente información:

- Estado del SPOC – el Estado para el que el SPOC proporciona la interfaz de comunicación;
- URL del SPOC – la URL del WSDL que describe la interfaz del SPOC y la ubicación del servicio; y
- certificado CA del SPOC – certificado (o certificados) utilizado para verificar los certificados de comunicación del SPOC.

8.1 Estructuras relacionadas con el SPOC

Las estructuras siguientes están definidas para usarse en los mensajes del SPOC.

8.1.1 Estructura de solicitud de certificado

Las solicitudes de certificado son certificados reducidos verificables mediante tarjeta que pueden llevar una firma adicional. SE UTILIZARÁ el perfil de solicitud indicado en la tabla 17.

Tabla 17. Perfil de solicitud de certificado CV

Objeto de datos	Presencia en el certificado
Autenticación	c
Certificado CV	m
Cuerpo del certificado	m
Identificador de perfil de certificado	m
Referencia de autoridad de certificación	r
Clave pública	m
Referencia de la persona titular de certificado	m
Firma	m
Referencia de autoridad de certificación	c
Firma	c

8.1.1.1 Identificador de perfil de certificado

La versión es 1, identificada por un valor de 0.

8.1.1.2 Referencia de la autoridad de certificación

La referencia de la autoridad de certificación DEBERÍA utilizarse para informar a la autoridad de certificación sobre la clave privada que se espera que el solicitante utilice para firmar el certificado. Si la referencia de la autoridad de certificación contenida en la solicitud se desvía de la referencia de la autoridad de certificación contenida en el certificado expedido (es decir, el certificado expedido está firmado por una clave privada que el solicitante no espera), el certificado correspondiente de la autoridad de certificación DEBERÍA proporcionarse también al solicitante como respuesta.

8.1.1.3 Clave pública

Las solicitudes de certificado DEBEN contener siempre parámetros de dominio.

8.1.1.4 Referencia de la persona titular de certificado

La referencia de la persona titular de certificado se utiliza para identificar la clave pública contenida en la solicitud y el certificado resultante.

8.1.1.5 Firma(s)

Una solicitud de certificado puede tener hasta dos firmas: una interna y otra externa:

Firma interna (SE EXIGE)

El cuerpo del certificado está autofirmado, es decir, que la firma interna DEBE ser verificable con la clave pública contenida en la solicitud de certificado. La firma SE CREARÁ sobre el cuerpo del certificado codificado (es decir, incluyendo la etiqueta y la longitud).

Firma externa (CONDICIONAL)

- La firma es OPCIONAL si una entidad solicita el certificado inicial. En este caso, la solicitud PUEDE llevar una forma adicional de otra entidad de confianza de la autoridad de certificación receptora (p. ej., la CVCA nacional puede autenticar la solicitud de un DV enviada a una CVCA extranjera).

- La firma SE EXIGE si una entidad solicita un certificado sucesivo. En este caso, la solicitud DEBE llevar la firma adicional del solicitante utilizando un par de claves reciente registrado previamente en la autoridad de certificación receptora.

Si se utiliza la firma externa, SE UTILIZARÁ un objeto de datos de autenticación para anidar el certificado CV (solicitud), la referencia de la autoridad de certificación y la firma adicional. La referencia de la autoridad de certificación IDENTIFICARÁ la clave pública que se utilizará para verificar la firma adicional. La firma SE CREARÁ sobre la concatenación del certificado CV codificado y la referencia de la autoridad de certificación codificada (es decir, ambas incluyendo la etiqueta y la longitud).

8.2 Mensajes del protocolo del SPOC

En esta sección se especifican los mensajes utilizados en el protocolo del SPOC.

8.2.1 Mensaje de certificado de solicitud

Uso previsto:

El mensaje RequestCertificate es utilizado por un SPOC para solicitar la generación de un nuevo certificado para uno de sus DV a una CVCA extranjera.

Parámetros de entrada:

callerID: (obligatorio)
Este parámetro contiene el identificador del Estado originador de la solicitud. El valor SERÁ el código de país de dos letras que figura en el Doc 9303-3. El valor de callerID DEBE ser verificado por el SPOC receptor con el valor registrado desde el SPOC originador durante su registro.

messageID: (obligatorio)
Este parámetro contiene la identificación del mensaje y DEBE identificar el mensaje unívocamente dentro de todos los mensajes de ese emisor. Si se envía un mensaje de respuesta al emisor como resultado de este mensaje, el mensaje de respuesta contendrá el mismo messageID. Así, un mensaje de respuesta entrante puede asignarse al mensaje original correcto. La construcción y asignación del messageID puede ser decidida por el emisor y no es verificada por la parte receptora.

certReq: (obligatorio)
Este parámetro contiene la solicitud de certificado propiamente dicha. DEBE construirse de conformidad con la sección 8.1.1. La codificación DEBE seguir las especificaciones que se indican en la sección 7.2.3.1.

Parámetros de salida:

CertificateSeq: (condicional)
Este parámetro contendrá el resultado (uno o más certificados) después de que se procese el mensaje, si el mensaje ha sido procesado satisfactoriamente y de forma sincronizada por el receptor. SE EXIGE si los certificados deben enviarse con la respuesta. NO DEBE estar presente si los certificados no se enviarán con el mensaje.

Códigos de retorno:

- ok_cert_available: El mensaje se ha procesado satisfactoriamente y de forma sincronizada. El parámetro de salida certificateSeq contiene uno o más certificados.
- ok_reception_ack: Se acusa recepción del mensaje. El mensaje aún no se ha verificado. El procesamiento del mensaje se realizará de forma asíncrona. El resultado del procesamiento se enviará a la URL registrada utilizando el mensaje SendCertificates.
- failure_inner_signature: La verificación de la firma interna de la solicitud de certificado propiamente dicha ha fallado.
- failure_outer_signature: La verificación de la firma externa de la solicitud de certificado propiamente dicha ha fallado.
- failure_syntax: El mensaje no es correcto sintácticamente.
- failure_request_not_accepted: El mensaje se ha procesado correctamente pero la solicitud no ha sido aceptada.

- failure_request_syntax: La solicitud de certificado no es correcta (p. ej., sintaxis o formato de archivo)
- failure_expired: El certificado que debe utilizarse para verificar la firma externa de la solicitud ha caducado.
- failure_domain_parameters: Los parámetros de dominio contenidos en la solicitud no coinciden con los parámetros de dominio del certificado CVCA previsto para firmar el certificado DV solicitado.
- failure_internal_error: Error distinto de los anteriores.

Observaciones:

El cuerpo de la solicitud de certificado DEBERÍA contener una referencia de la autoridad de certificación (CAR) para informar a la CVCA de la clave privada que el solicitante espera que se utilice para firmar el certificado. Si la CAR de la solicitud difiere de la CAR del certificado expedido, también SE PROPORCIONARÁ en la respuesta el certificado correspondiente de la CVCA. En ese caso, y si el mensaje se procesa de forma sincrónica, el certificado CVCA DEBE formar parte del parámetro de salida certificateSeq. El certificado DV DEBE ser el primer certificado de la secuencia. Los certificados CVCA (raíz y/o de enlace) SE ORDENARÁN por fecha de entrada en vigor (ascendente) en la secuencia.

8.2.2 Mensaje de envío de certificados

Uso previsto:

El mensaje SendCertificates es utilizado por un SPOC para enviar el nuevo certificado o cadena de certificados al SPOC solicitante. Este mensaje SE GENERARÁ en respuesta a:

- RequestCertificate: al procesar satisfactoriamente la solicitud asíncrona después de que se expida el certificado;
- GetCACertificates

Además, el mensaje DEBE utilizarse cuando se crea un nuevo certificado (CVCA raíz y de enlace) para enviar los certificados al SPOC extranjero registrado.

Parámetros de entrada:

callerID: (obligatorio)
Este parámetro contiene el identificador del Estado originador. El valor SERÁ el código de país de dos letras que figura en el Doc 9303-3. El valor de callerID DEBE ser verificado por el SPOC receptor con el valor registrado desde el SPOC originador durante su registro.

messageID: (condicional)
Cuando el mensaje se genera en respuesta a un mensaje de solicitud, el parámetro DEBE contener el mismo valor que el parámetro messageID del mensaje de solicitud. Cuando la generación del mensaje se haya desencadenado sin intervención externa (cambio de claves del certificado CVCA), el valor de statusInfo SERÁ new_cert_available_notification y el parámetro messageID PUEDE omitirse y SE IGNORARÁ cuando esté presente.

statusInfo: (obligatorio)
Este parámetro contiene un código de estado sobre el resultado del procesamiento del mensaje correspondiente. Pueden darse los estados siguientes:

- new_cert_available_notification: El SPOC originador desea notificar que el nuevo certificado (o certificados) CVCA está disponible sin que se haya solicitado.

- ok_cert_available: La solicitud se ha procesado satisfactoriamente. El parámetro de entrada certificateSeq contiene uno o más certificados.

- failure_inner_signature: La verificación de la firma interna de la solicitud de certificado propiamente dicha ha fallado.

- failure_outer_signature: La verificación de la firma externa de la solicitud de certificado propiamente dicha ha fallado.

- failure_syntax: El mensaje correspondiente no es correcto sintácticamente.

- failure_request_not_accepted: El mensaje correspondiente se ha procesado correctamente pero la solicitud no ha sido aceptada.

- failure_certificate: Uno o varios de los certificados enviados no es correcto (sintaxis o firma).

- failure_internal_error: error distinto de certificateSeq (condicional).

Este parámetro SE EXIGE si los certificados deben enviarse con el mensaje. NO DEBE estar presente si los certificados no se enviarán con el mensaje. Los certificados DEBEN estar codificados en formato binario TLV DER, según se define en la sección 7.2.3.

Cuando el mensaje se genera en respuesta a un mensaje GetCACertificates, o porque hay un nuevo certificado, la secuencia CONTENDRÁ una lista de certificados CA. La lista DEBE estar ordenada. Los certificados CVCA (de enlace y/o de raíz) SE ORDENARÁN por fecha de entrada en vigor en la secuencia. Cuando la secuencia contenga certificados con diferentes parámetros de dominio, DEBE estar presente como mínimo un certificado con parámetros de dominio incluidos para cada variante de parámetros de dominio. SE INCLUIRÁN todos los certificados CA actuales.

Cuando el mensaje se genera en respuesta al mensaje RequestCertificate, el contenido de la secuencia es el mismo que el descrito para la respuesta sincrónica de RequestCertificate.

Parámetros de salida:

Ninguno

Códigos de retorno:

- ok_received_correctly: El mensaje se ha recibido correctamente.

- failure_syntax: El mensaje no es correcto sintácticamente.

- failure_messageID_unknown: El messageID contenido no coincide con un mensaje enviado anteriormente.

- failure_internal_error: Error distinto de los anteriores.

8.2.3 Mensaje de obtención de certificados CA

Uso previsto:

Este mensaje es enviado por un SPOC a un SPOC extranjero para obtener todos los certificados CVCA válidos (certificados de enlace y certificados autofirmados) de ese Estado.

Parámetros de entrada:

callerID: (obligatorio)
Este parámetro contiene el identificador del Estado originador. El valor SERÁ el código de país de dos letras que figura en el Doc 9303-3. El valor de callerID DEBE ser verificado por el SPOC receptor con el valor registrado desde el SPOC originador durante su registro.

messageID: (obligatorio)
Este parámetro contiene la identificación del mensaje. DEBE identificar el mensaje unívocamente dentro de todos los mensajes de ese emisor. Si se envía un mensaje de respuesta al emisor como resultado de este mensaje, el mensaje de respuesta contendrá el mismo messageID. Así, un mensaje de respuesta entrante puede asignarse al mensaje original correcto. La construcción y asignación del messageID puede ser decidida por el emisor.

Parámetros de salida:

certificateSeq: (condicional)
Este parámetro contendrá el resultado (uno o más certificados) después de que se procese el mensaje, si el mensaje ha sido procesado satisfactoriamente y de forma sincronizada por el receptor. SE EXIGE si los certificados deben enviarse con la respuesta. NO DEBE estar presente si los certificados no se enviarán con el mensaje.

Códigos de retorno:

- ok_cert_available: El mensaje se ha procesado satisfactoriamente y de forma sincronizada. El parámetro de salida certificateSeq contiene uno o más certificados CA.
- ok_reception_ack: Se acusa recepción del mensaje. El mensaje aún no se ha verificado. El procesamiento del mensaje se realizará de forma asíncrona. El resultado del procesamiento se enviará a la URL registrada utilizando el mensaje SendCertificates.
- failure_syntax: El mensaje no es correcto sintácticamente.
- failure_internal_error: Error distinto de los anteriores.

Observaciones:

Si el mensaje se procesa satisfactoriamente y se acepta, la CVCA DEBE enviar todos los certificados CVCA válidos dentro de la respuesta, bien en el parámetro de salida certificateSeq (procesamiento síncrono) o en el mensaje de respuesta correspondiente SendCertificates (procesamiento asíncrono).

8.2.4 Mensajes generales

Uso previsto:

Este mensaje es enviado por un SPOC a un SPOC extranjero para transmitir una notificación u otro mensaje de texto general de lectura por seres humanos.

Parámetros de entrada:

callerID: (obligatorio)
Este parámetro contiene el identificador del Estado originador. El valor SERÁ el código de país de dos letras que figura en el Doc 9303-3. El valor de callerID DEBE ser verificado por el SPOC receptor con el valor registrado desde el SPOC originador durante su registro, incluidas las características de seguridad del mensaje (el certificado de firma digital/certificado de cliente TLS está registrado para el Estado respectivo).

messageID: (obligatorio)
Este parámetro contiene la identificación del mensaje y DEBE identificar el mensaje unívocamente dentro de todos los mensajes de ese emisor. Si se envía un mensaje de respuesta al emisor como resultado de este mensaje, el mensaje de respuesta contendrá el mismo messageID. Así, un mensaje de respuesta entrante puede asignarse al mensaje original correcto. La construcción y asignación del messageID puede ser decidida por el emisor.

subject: (obligatorio)
Este parámetro contiene el asunto del mensaje. El asunto DEBE describir brevemente el contenido del cuerpo del mensaje. El asunto DEBE estar escrito en inglés.

body: (obligatorio)
Este parámetro contiene el cuerpo del mensaje. El cuerpo DEBE ser un texto claro de lectura por seres humanos y no estar concebido para un procesamiento automatizado directo. El cuerpo DEBE estar escrito en inglés.

Códigos de retorno:

- ok: El mensaje se ha aceptado para su entrega.
- failure_syntax: El mensaje no es correcto sintácticamente.
- failure_internal_error: Error distinto de los anteriores.

8.3 Servicio web

La interfaz del servicio web es la interfaz para el intercambio rutinario de datos por cable entre SPOC. La interfaz UTILIZARÁ el protocolo [SOAP] sobre [HTTPS]. La interfaz del servicio web del SPOC SE AJUSTARÁ al WSDL especificado en la sección 8.3.3.

8.3.1 Uso del SOAP

Se utilizará el protocolo [SOAP] sobre [HTTPS] puro para implementar las interfaces del servicio web. NO DEBE utilizarse ninguna otra extensión SOAP (p. ej., WS-Addressing, WS-Security, WS-Secure Conversation, WS-Authorization, WS-Federation, WSAuthorization, WS-Policy, WS-Trust, WS-Privacy, WS-Test ni otras extensiones de WS).

NO DEBE utilizarse el tipo de nodo SOAP intermediario. Solo SE UTILIZARÁ una configuración directa del SPOC cliente al SPOC servidor.

El elemento de fallo de SOAP SE UTILIZARÁ únicamente cuando se produzca un error de procesamiento de la capa de transporte que no esté cubierto por esta especificación. Los errores en el ámbito de la aplicación SE COMUNICARÁN como respuestas SOAP normales utilizando el mecanismo de error descrito para cada mensaje.

SE RECOMIENDA que la interfaz del servicio web se implemente de conformidad con [WS-IBP] y [WSI-SSBP].

La interfaz SOAP del SPOC DEBE ser conforme a las definiciones WSDL descritas en la sección 8.3.3.

8.3.2 Consideraciones relativas a la seguridad

La comunicación del servicio web del SPOC UTILIZARÁ un canal seguro y autenticado. SE UTILIZARÁ SOAP sobre HTTPS. SE UTILIZARÁ TLS v1.2.

El cliente TLS REALIZARÁ las verificaciones siguientes:

- el certificado de servidor SE VALIDARÁ completamente de acuerdo con [RFC5280], incluido el estado de revocación;
- la extensión ExtKeyUsage del certificado de servidor DEBE estar presente y CONTENDRÁ los OID correspondientes al certificado de servidor TLS del SPOC que se indican en la sección 7.2.1; y
- el país del certificado de servidor SERÁ igual al valor del parámetro callerID. En caso de fallo, el cliente TLS DEBE cerrar la conexión.

El servidor TLS REALIZARÁ las verificaciones siguientes:

- el cliente DEBE autenticarse totalmente mediante un certificado;
- el certificado de cliente SE VALIDARÁ completamente de acuerdo con [RFC5280], incluido el estado de revocación;
- la extensión ExtKeyUsage del certificado de cliente DEBE estar presente y CONTENDRÁ los OID correspondientes al certificado de cliente TLS del SPOC que se indican en la sección 7.2.1; y
- el país del certificado de cliente DEBE corresponderse con el país previsto.

En caso de que alguna de las verificaciones falle, la solicitud SE RECHAZARÁ utilizando el código de respuesta HTTP 401 Unauthorized.

En el ámbito de la negociación del protocolo de enlace TLS, el cliente ADMITIRÁ todos los conjuntos de cifrado TLS definidos en la sección 4.2.2. Tanto el servidor como el cliente ADMITIRÁN la autenticación basada en RSA y ECDSA. Está permitido que un servidor solicite y también que el cliente envíe un certificado de cliente de un tipo diferente al del servidor.

El uso del acuerdo de clave ECDHE_ECDSA en el protocolo de enlace TLS es conforme las adiciones definidas en [TLSECC], [TLS1.2] y [TLSEXT]. Tanto el cliente como el servidor ADMITIRÁN las extensiones de curvas elípticas apropiadas definidas en la especificación [TLSECC] en el ámbito del protocolo de enlace TLS. Las curvas elípticas y los formatos de puntos de curva elíptica admitidos se definen en la sección 5 de [TLSECC]. El uso de los conjuntos de cifrado TLS admitidos definidos en la sección 4.2.2 que utilizan la norma de cifrado avanzado (AES) para el cifrado DEBE ser conforme a la especificación [TLSAES].

8.3.3 WSDL para la interfaz del servicio web del SPOC

La interfaz SOAP de SPOC DEBE ajustarse a las definiciones WSDL siguientes:

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/
  " xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:SPOC="http://namespaces.icao.int/lds2"
  targetNamespace="http://namespaces. icao.int/lds2">

  <wsdl:types>
    <xs:schema xmlns="http://namespaces.icao.int/lds2"
      targetNamespace="http://namespaces."
      elementFormDefault="qualified" attributeFormDefault="unqualified">
      <xs:element name="certificateSequence">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="certificate" type="xs:base64Binary" minOccurs="1"
maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
      </xs:element>
      <xs:element name="RequestCertificateRequest">
    <xs:complexType>
      <xs:sequence>
            <xs:element name="callerID" type="xs:string"/>
            <xs:element name="messageID" type="xs:string"/>
            <xs:element name="certificateRequest" type="xs:base64Binary"/>
      </xs:sequence>
    </xs:complexType>
      </xs:element>
      <xs:element name="RequestCertificateResponse">
    <xs:complexType>
      <xs:sequence>
            <xs:element ref="certificateSequence" minOccurs="0" maxOccurs="1"/>
            <xs:element name="result">
          <xs:simpleType>
              <xs:restriction base="xs:string">
                <xs:enumeration value="ok_cert_available"/>
                <xs:enumeration value="ok_reception_ack"/>
                <xs:enumeration value="failure_inner_signature"/>
                <xs:enumeration value="failure_outer_signature"/>
                <xs:enumeration value="failure_syntax"/>
                <xs:enumeration value="failure_request_not_accepted"/>
                <xs:enumeration value="failure_request_syntax"/>
                <xs:enumeration value="failure_expired"/>
                <xs:enumeration value="failure_domain_parameters"/>
                <xs:enumeration value="failure_internal_error"/>
            </xs:restriction>
          </xs:simpleType>
```

```
    </xs:element>
  </xs:sequence>
</xs:complexType>
  </xs:element>
  <xs:element name="SendCertificatesRequest">
<xs:complexType>
  <xs:sequence>
       <xs:element name="callerID" type="xs:string"/>
       <xs:element name="messageID" type="xs:string" minOccurs="0" maxOccurs="1"/>
       <xs:element ref="certificateSequence" minOccurs="0" maxOccurs="1"/>
       <xs:element name="statusInfo">
     <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="new_cert_available_notification"/>
            <xs:enumeration value="ok_cert_available"/>
            <xs:enumeration value="failure_inner_signature"/>
            <xs:enumeration value="failure_outer_signature"/>
            <xs:enumeration value="failure_syntax"/>
            <xs:enumeration value="failure_request_not_accepted"/>
            <xs:enumeration value="failure_certificate"/>
            <xs:enumeration value="failure_internal_error"/>
       </xs:restriction>
     </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
  </xs:element>
  <xs:element name="SendCertificatesResponse">
<xs:complexType>
       <xs:sequence>
       <xs:element name="result">
     <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="ok_received_correctly"/>
            <xs:enumeration value="failure_syntax"/>
            <xs:enumeration value="failure_messageID_unknown"/>
            <xs:enumeration value="failure_internal_error"/>
       </xs:restriction>
     </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
  </xs:element>
  <xs:element name="GetCACertificatesRequest">
<xs:complexType>
  <xs:sequence>
       <xs:element name="callerID" type="xs:string"/>
       <xs:element name="messageID" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
  </xs:element>
  <xs:element name="GetCACertificatesResponse">
```

```
  <xs:complexType>
   <xs:sequence>
        <xs:element ref="certificateSequence" minOccurs="0" maxOccurs="1"/>
        <xs:element name="result">
       <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="ok_cert_available"/>
              <xs:enumeration value="ok_reception_ack"/>
              <xs:enumeration value="failure_syntax"/>
              <xs:enumeration value="failure_internal_error"/>
        </xs:restriction>
       </xs:simpleType>
     </xs:element>
   </xs:sequence>
  </xs:complexType>
   </xs:element>
   <xs:element name="GeneralMessageRequest">
  <xs:complexType>
   <xs:sequence>
        <xs:element name="callerID" type="xs:string"/>
        <xs:element name="messageID" type="xs:string"/>
        <xs:element name="subject" type="xs:string"/>
        <xs:element name="body" type="xs:string"/>
   </xs:sequence>
  </xs:complexType>
   </xs:element>
   <xs:element name="GeneralMessageResponse">
  <xs:complexType>
   <xs:sequence>
        <xs:element name="result">
       <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="ok"/>
              <xs:enumeration value="failure_syntax"/>
              <xs:enumeration value="failure_internal_error"/>
        </xs:restriction>
       </xs:simpleType>
     </xs:element>
   </xs:sequence>
  </xs:complexType>
</xs:element>
  </xs:schema>
</wsdl:types>

<wsdl:message name="RequestCertificateRequest">
  <wsdl:part name="RequestCertificateRequest" element="SPOC:RequestCertificateRequest"/>
</wsdl:message>
<wsdl:message name="RequestCertificateResponse">
  <wsdl:part name="RequestCertificateResponse" element="SPOC:RequestCertificateResponse"/>
</wsdl:message>
```

```
<wsdl:message name="SendCertificatesRequest">
  <wsdl:part name="SendCertificatesRequest" element="SPOC:SendCertificatesRequest"/>
</wsdl:message>
<wsdl:message name="SendCertificatesResponse">
  <wsdl:part name="SendCertificatesResponse" element="SPOC:SendCertificatesResponse"/>
</wsdl:message>

<wsdl:message name="GetCACertificatesRequest">
  <wsdl:part name="GetCACertificatesRequest" element="SPOC:GetCACertificatesRequest"/>
</wsdl:message>
<wsdl:message name="GetCACertificatesResponse">
  <wsdl:part name="GetCACertificatesResponse" element="SPOC:GetCACertificatesResponse"/>
</wsdl:message>

<wsdl:message name="GeneralMessageRequest">
  <wsdl:part name="GeneralMessageRequest" element="SPOC:GeneralMessageRequest"/>
</wsdl:message>
<wsdl:message name="GeneralMessageResponse">
  <wsdl:part name="GeneralMessageResponse" element="SPOC:GeneralMessageResponse"/>
</wsdl:message>

<wsdl:portType name="SPOCPortType">
  <wsdl:operation name="RequestCertificate">
    <wsdl:input message="SPOC:RequestCertificateRequest"/>
    <wsdl:output message="SPOC:RequestCertificateResponse"/>
  </wsdl:operation>
  <wsdl:operation name="SendCertificates">
    <wsdl:input message="SPOC:SendCertificatesRequest"/>
    <wsdl:output message="SPOC:SendCertificatesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GetCACertificates">
    <wsdl:input message="SPOC:GetCACertificatesRequest"/>
    <wsdl:output message="SPOC:GetCACertificatesResponse"/>
  </wsdl:operation>
  <wsdl:operation name="GeneralMessage">
    <wsdl:input message="SPOC:GeneralMessageRequest"/>
    <wsdl:output message="SPOC:GeneralMessageResponse"/>
  </wsdl:operation>
</wsdl:portType>

<wsdl:binding name="SPOCSOAPBinding" type="SPOC:SPOCPortType">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="RequestCertificate">
    <soap:operation soapAction="RequestCertificate"/>
<wsdl:input>
     <soap:body parts="RequestCertificateRequest" use="literal"/>
    </wsdl:input>
    <wsdl:output>
     <soap:body parts="RequestCertificateResponse" use="literal"/>
    </wsdl:output>
</wsdl:operation>
  <wsdl:operation name="SendCertificates">
```

```
      <soap:operation soapAction="SendCertificates"/>
      <wsdl:input>
        <soap:body parts="SendCertificatesRequest" use="literal"/>
    </wsdl:input>
      <wsdl:output>
        <soap:body parts="SendCertificatesResponse" use="literal"/>
      </wsdl:output>
  </wsdl:operation>
    <wsdl:operation name="GetCACertificates">
      <soap:operation soapAction="GetCACertificates"/>
      <wsdl:input>
        <soap:body parts="GetCACertificatesRequest" use="literal"/>
    </wsdl:input>
      <wsdl:output>
        <soap:body parts="GetCACertificatesResponse" use="literal"/>
      </wsdl:output>
  </wsdl:operation>
    <wsdl:operation name="GeneralMessage">
      <soap:operation soapAction="GeneralMessage"/>
      <wsdl:input>
        <soap:body parts="GeneralMessageRequest" use="literal"/>
    </wsdl:input>
      <wsdl:output>
        <soap:body parts="GeneralMessageResponse" use="literal"/>
      </wsdl:output>
  </wsdl:operation>
   </wsdl:binding>

   <wsdl:service name="SPOC">
    <wsdl:port name="SPOCPort" binding="SPOC:SPOCSOAPBinding">
      <soap:address location="http://spoc-server/SPOC"/>
    </wsdl:port>
   </wsdl:service>
  </wsdl:definitions>
```

9. ESTRUCTURA DE LISTA MAESTRA DE CSCA

Las listas maestras se implementan como casos del tipo `ContentInfo`, según se especifica en [RFC 5652]. La `ContentInfo` DEBE contener un único caso del tipo `SignedData` como se detalla a continuación. En `ContentInfo` no se incluye ningún otro tipo de datos. Todas las listas maestras DEBEN producirse en formato DER para conservar en ellas la integridad de las firmas.

9.1 Tipo SignedData (datos firmados)

Se aplican las reglas de procedimiento de [RFC 5652].

La especificación de estructura de lista maestra utiliza la siguiente terminología para los requisitos de presencia de cada campo.

- m obligatorio — el campo DEBE estar presente;
- r recomendado — el campo DEBERÍA estar presente;
- x no utilizar — el campo NO DEBE estar presente;
- o opcional — el campo PUEDE estar presente.

Tabla 18. Lista maestra

Valor		Comentarios
`SignedData`		
`Version`	m	Valor = `v3`
`digestAlgorithms`	m	
`encapContentInfo`	m	
`eContentType`	m	`id-icao-cscaMasterList`
`eContent`	m	Contenido codificado de una `cscaMasterList`.
`Certificates`	m	El certificado de firmante de la lista maestra DEBE incluirse y el certificado CSCA, que puede utilizarse para verificar la firma en el campo `signerInfos` DEBERÍA incluirse.
`Crls`	x	
`signerInfos`	m	SE RECOMIENDA que los Estados solo proporcionen 1 `signerinfo` dentro de este campo.
`SignerInfo`	m	
`Version`	m	El valor de este campo está dictado por el campo `sid`. Véanse en [RFC 5652] las reglas relativas a este campo.
`Sid`	m	

Valor		Comentarios
subjectKeyIdentifier	r	SE RECOMIENDA que se admita este campo en vez de issuerandSerialNumber.
digestAlgorithm	m	Identificador de algoritmo del algoritmo utilizado para producir el valor condensado sobre encapsulatedContent y SignedAttrs. Véase la nota.
signedAttrs	m	Pueden incluirse atributos adicionales. No obstante, estos no tienen que ser procesados por los Estados receptores excepto para verificar el valor de la firma. signedAttrs DEBE incluir la hora de la firma (véase [PKCS #9]).
signatureAlgorithm	m	Identificador de algoritmo del algoritmo utilizado para producir el valor de firma y cualquier parámetro conexo. Véase la nota.
signature	m	El resultado del proceso de generación de firma.
unsignedAttrs	o	Aunque este campo PUEDE incluirse, los Estados receptores pueden optar por ignorarlo.

Nota.— DigestAlgorithmIdentifiers DEBEN omitir parámetros "NULL", mientras que el SignatureAlgorithmIdentifier (definido en RFC 3447) DEBE incluir NULL como parámetro si no están presentes otros parámetros, incluso cuando se utilizan algoritmos SHA2 con arreglo a RFC 5754. Las implementaciones DEBEN aceptar DigestAlgorithmIdentifiers en ambas condiciones, parámetros ausentes o parámetros NULL.

9.2 Especificación ASN.1 de la lista maestra

```
CscaMasterList
{ joint-iso-itu-t(2) international-organization(23) icao(136) mrtd(1)
security(1) masterlist(2)}

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

IMPORTS

 -- Imports from RFC 5280 [PROFILE], Appendix A.1
     Certificate
       FROM PKIX1Explicit88
         { iso(1) identified-organization(3) dod(6)
           internet(1) security(5) mechanisms(5) pkix(7)
             mod(0) pkix1-explicit(18) };
```

```
-- CSCA Master List

CscaMasterListVersion ::= INTEGER {v0(0)}

CscaMasterList ::= SEQUENCE {
  version            CscaMasterListVersion,
  certList           SET OF Certificate }

-- Object Identifiers

id-icao-cscaMasterList OBJECT IDENTIFIER ::=
                                     {id-icao-mrtd-security 2}
id-icao-cscaMasterListSigningKey OBJECT IDENTIFIER ::=
                                     {id-icao-mrtd-security 3}
END
```

10. ESTRUCTURA DE LA LISTA DE DESVIACIONES

La lista de desviaciones se implementa como un tipo SignedData, según se especifica en [RFC 3852]. Todas las listas de desviaciones DEBEN producirse en formato DER para conservar en ellas la integridad de las firmas.

El intervalo de desviaciones estará delimitado por:

- intervalo de fechas (incluyendo la fecha de expedición y la de caducidad);
- nombre del expedidor y número de serie;
- el identificador subjectKeyIdentifier del DSC;
- lista de números de eMRTD.

Se utilizarán combinaciones apropiadas de estos valores para vincular con precisión el intervalo de MRTD afectados. Cuando se combinen los valores, estos deben procesarse unidos por “AND” (Y). No puede optarse por procesar valores unidos utilizando “OR” (O).

10.1 Tipo SignedData (datos firmados)

Se aplican las reglas de procesamiento de [RFC 3852].

m obligatorio — el campo DEBE estar presente;
r recomendado — el campo DEBERÍA estar presente;
x no utilizar — el campo NO DEBE estar presente;
o opcional — el campo PUEDE estar presente.

Tabla 19. Lista de desviaciones

Valor		Comentarios
SignedData		
version	m	Valor = v3
digestAlgorithms	m	
encapContentInfo	m	
eContentType	m	id-icao-DeviationList
eContent	m	La DeviationList del contenido codificado.
certificates	m	Los estados DEBEN incluir el certificado de firmante de la lista de desviaciones y DEBERÍAN incluir el certificado CSCA, que puede utilizarse para verificar la firma en el campo signerInfos.
crls	x	
signerInfos	m	SE RECOMIENDA que los Estados solo proporcionen 1 signerinfo dentro de este campo.
SignerInfo	m	
version	m	El valor de este campo está dictado por el campo sid. Véanse las reglas relativas a este campo en [RFC 3852] sección 5.3.
sid	m	
subjectKeyIdentifier	r	SE RECOMIENDA que los Estados admitan este campo más que issuerandSerialNumber.
digestAlgorithm	m	Identificador de algoritmo del algoritmo utilizado para producir el valor condensado sobre encapsulatedContent y SignedAttrs.
signedAttrs	m	Es posible que los Estados productores deseen disponer de atributos adicionales para su inclusión en la firma, pero estos no tienen que ser procesados por los Estados receptores, salvo para verificar el valor de la firma. signedAttrs DEBE incluir la hora de la firma (véase [PKCS #9]).
signatureAlgorithm	m	Identificador de algoritmo del algoritmo utilizado para producir el valor de firma y cualquier parámetro conexo.
signature	m	El resultado del proceso de generación de firma.
unsignedAttrs	x	

10.2 Especificación ASN.1

```
DeviationList
{ joint-iso-itu-t (2) international-organization(23) icao(136) mrtd(1) security(1)
deviationlist(7)}

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

IMPORTS

  -- Imports from RFC 3280 [PROFILE], Appendix A.1
       AlgorithmIdentifier
           FROM PKIX1Explicit88
                  { iso(1) identified-organization(3) dod(6)
                    internet(1) security(5) mechanisms(5) pkix(7)
                    mod(0) pkix1-explicit(18) }

  -- Imports from RFC 3852
       SubjectKeyIdentifier, Digest, IssuerAndSerialNumber
           FROM CryptographicMessageSyntax2004
                  { iso(1) member-body(2) us(840) rsadsi(113549)
                    pkcs(1) pkcs-9(9) smime(16) modules(0)
                    cms-2004(24) };

DeviationListVersion ::= INTEGER {v0(0)}

DeviationList ::= SEQUENCE {
 version      DeviationListVersion,
 digestAlgorithm AlgorithmIdentifier OPTIONAL,
 deviations   SET OF Deviation
}

Deviation ::= SEQUENCE{
 documents       DeviationDocuments,
 descriptions SET OF DeviationDescription
}

DeviationDescription ::= SEQUENCE{
 description     PrintableString OPTIONAL,
 deviationType   OBJECT IDENTIFIER,
 parameters   [0] ANY DEFINED BY deviationType OPTIONAL,
 nationalUse     [1] ANY OPTIONAL

 -- The nationalUse field is for internal State use, and is not governed
 -- by an ICAO specification.
}

DeviationDocuments ::= SEQUENCE {
 documentType    [0] PrintableString (SIZE(2)) OPTIONAL,
          -- per MRZ, e.g. 'P'
 dscIdentifier   DocumentSignerIdentifier OPTIONAL,
```

```
 issuingDate     [4] IssuancePeriod OPTIONAL,
 documentNumbers    [5] SET OF PrintableString OPTIONAL
}

DocumentSignerIdentifier ::= CHOICE{
 issuerAndSerialNumber [1] IssuerAndSerialNumber,
 subjectKeyIdentifier [2] SubjectKeyIdentifier,
 certificateDigest  [3] Digest   --  if  used,  digestAlgorithm  must  be  present  in
DeviationList
}

IssuancePeriod ::= SEQUENCE {
 firstIssued   GeneralizedTime,
 lastIssued   GeneralizedTime
}

-- CertField is used to define which part of a certificate is
  affected by a coding error. Parts of the Body are identified by
  the corresponding value of CertificateBodyField, extensions
  by the corresponding OID identifying the extension.

CertField ::= CHOICE {
 body  CertificateBodyField,
 extension OBJECT IDENTIFIER
}
CertificateBodyField ::= INTEGER {
 generic(0), version(1), serialNumber(2), signature(3), issuer(4),
 validity(5), subject(6), subjectPublicKeyInfo(7),
 issuerUniqueID(8), subjectUniqueID(9)
}

Datagroup ::= INTEGER
              {dg1(1), dg2(2), dg3(3), dg4(4), dg5(5), dg6(6),
               dg7(7), dg8(8), dg9(9), dg10(10), dg11(11),
               dg12(12), dg13(13), dg14(14), dg15(15), dg16(16),
               sod(20), com(21)}

MRZField ::= INTEGER
              {generic(0), documentCode(1), issuingState(2), personName(3),
               documentNumber(4), nationality(5), dateOfBirth(6),
               sex(7), dateOfExpiry(8), optionalData(9)}

-- Base Object Identifiers

id-icao OBJECT IDENTIFIER ::= {2 23 136 }
id-icao-mrtd OBJECT IDENTIFIER ::= {id-icao 1}
id-icao-mrtd-security OBJECT IDENTIFIER ::= {id-icao-mrtd 1}
id-icao-DeviationList OBJECT IDENTIFIER ::= {id-icao-mrtd-security 7}
id-icao-DeviationListSigningKey OBJECT IDENTIFIER ::= {id-icao-mrtd-security 8}

-- Deviation Object Identifiers and Parameter Definitions
```

```
id-Deviation-CertOrKey OBJECT IDENTIFIER ::= {id-icao-DeviationList 1}
id-Deviation-CertOrKey-DSSignature OBJECT IDENTIFIER ::= {id-Deviation-CertOrKey 1}
id-Deviation-CertOrKey-DSEncoding OBJECT IDENTIFIER ::= {id-Deviation-CertOrKey 2}
id-Deviation-CertOrKey-CSCAEncoding OBJECT IDENTIFIER ::= {id-Deviation-CertOrKey 3}
id-Deviation-CertOrKey-AAKeyCompromised OBJECT IDENTIFIER ::= {id-Deviation-CertOrKey 4}
id-Deviation-LDS OBJECT IDENTIFIER ::= {id-icao-DeviationList 2}
id-Deviation-LDS-DGMalformed OBJECT IDENTIFIER ::= {id-Deviation-LDS 1}
id-Deviation-LDS-DGHashWrong OBJECT IDENTIFIER ::= {id-Deviation-LDS 2}
id-Deviation-LDS-SODSignatureWrong OBJECT IDENTIFIER ::= {id-Deviation-LDS 3}
id-Deviation-LDS-COMInconsistent OBJECT IDENTIFIER ::= {id-Deviation-LDS 4}

id-Deviation-MRZ OBJECT IDENTIFIER ::= {id-icao-DeviationList 3}
id-Deviation-MRZ-WrongData OBJECT IDENTIFIER ::= {id-Deviation-MRZ 1}
id-Deviation-MRZ-WrongCheckDigit OBJECT IDENTIFIER ::= {id-Deviation-MRZ 2}

id-Deviation-Chip OBJECT IDENTIFIER ::= {id-icao-DeviationList 4}

id-Deviation-NationalUse OBJECT IDENTIFIER ::= {id-icao-DeviationList 5}

END
```

11. REFERENCIAS (NORMATIVA)

FIPS 180-2	FIPS 180-2, Federal Information Processing Standards Publication (FIPS PUB) 180-2, *Secure Hash Standard*, agosto de 2002.
FIPS 186-4	FIPS 186-4, Federal Information Processing Standards Publication (FIPS PUB) 186-4, *Digital Signature Standard (DSS)*, julio de 2013 (sustituye FIPS PUB 186-3 con fecha de junio de 2009).
ISO 3166-1	ISO/IEC 3166-1: 2006, Códigos para la representación de nombres de países y sus subdivisiones — Parte 1: Códigos de país.
ISO/IEC 15946	ISO/IEC 15946: 2002, Tecnología de la información — Técnicas de seguridad — Técnicas criptográficas basadas en curvas elípticas.
RFC 3280	RFC 3280, R. Housley, W. Polk, W. Ford, D. Solo, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, abril de 2002.
RFC 4055	RFC 4055, J. Schaad, B. Kaliski, R. Housley, Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, junio de 2005.
RFC 5652	RFC 5652, R. Housley, Cryptographic Message Syntax, septiembre de 2009.
RFC 5280	RFC 5280, D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, W. Polk, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, mayo de 2008.

TR 03111	BSI TR-03111: Elliptic Curve Cryptography v 2.0, 2012.
X9.62	X9.62, Criptografía de clave pública para la industria de servicios financieros: Algoritmo digital de curva elíptica (ECDSA), 7 de enero de 1999.
X.509	ITU-T X.509 | ISO/IEC 9594-8, 2008: Tecnología de la información – Interconexión de sistemas abiertos – El directorio: marcos para certificar entre claves públicas y atributos.
X.690	ITU-T X.690 2008: Tecnología de la información – Reglas de codificación ASN.1: Especificación de las reglas de codificación básicas (BER), de las reglas de codificación canónica (CER) y de las reglas de codificación distinguida (DER).
RFC-RSA	Jonsson, Jakob y Kaliski, Burt RFC 3447, Normas de criptografía de clave pública (PKCS)#1: Especificaciones de criptografía RSA, versión 2.1, 2003.
PKCS#1	RSA Laboratories, Nota técnica de RSA Laboratories, PKCS#1 v2.2: Norma de criptografía RSA, 2012.
TLSAES	Chown, P., “Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security (TLS)”, RFC 3268, junio de 2002.
TLSECC	Blake-Wilson, S., Bolyard, N., Gupta, V., Hawk, C. y B. Moeller, “Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS)”, RFC 4492, mayo de 2006.
TLS1.2	Dierks, T. y E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.2”, RFC 5246, agosto de 2008.
TLSEXT	Blake-Wilson, S., Nystrom, M., Hopwood, D., Mikkelsen, J. y T. Wright, “Transport Layer Security (TLS) Extensions”, RFC 4366, abril de 2006.
SOAP	SOAP, versión 1.2, parte 1: Marco de mensajería (segunda edición), Recomendación W3C, 27 de abril de 2007.
HTTPS	E. Rescorla., “HTTP Over TLS”, RFC 2818, mayo de 2000.
WSI-BP	Perfil básico WS-I, disponible en: http://www.ws-i.org/Profiles/BasicProfile-1.1.html
WSI-SSBP	Perfil de enlace básico WS-I, disponible en: http://www.ws-i.org/Profiles/SimpleSoapBindingProfile-1.0.html

— — — — — — — —

Apéndice A de la Parte 12

VIDA ÚTIL (INFORMATIVO)

Los ejemplos siguientes ilustran el cálculo de los períodos de uso de las claves privadas y de la validez de los certificados de clave pública para diversas situaciones posibles según se describen en la sección 4.

A.1 EJEMPLO 1

El primer ejemplo ilustra una situación en la cual los eMRTD son válidos por cinco años. Debido a que se expide una cantidad relativamente grande de eMRTD por día, se ha decidido mantener los períodos de uso de clave privada y de validez de certificado de clave pública en un mínimo. Para este ejemplo, el período mínimo de uso de clave privada para certificados de firmante del documento es de un mes.

Elemento	***Período de uso/validez***
Validez del eMRTD	5 años
Período de uso de clave privada de firmante del documento	1 mes
Validez de certificado de firmante del documento	5 años + 1 mes
Período de uso de clave privada CSCA	3 años
Validez de certificado CSCA	8 años + 1 mes

Las consecuencias de este ejemplo son que para cuando el primer certificado CSCA pierde validez, se habrán expedido por lo menos 36 certificados de firmante del documento (uno para cada clave privada con período de uso de un mes). En los últimos meses antes de que el primer certificado CSCA pierda validez, habrá por lo menos otros dos certificados CSCA expedidos (uno para cada clave privada con período de uso de tres años).

A.2 EJEMPLO 2

El segundo ejemplo ilustra una situación en el que los eMRTD son válidos por diez años. La política consiste en mantener con una longitud media los períodos de uso de clave privada y de validez de certificado de clave pública.

Elemento	***Período de uso/validez***
Validez del eMRTD	10 años
Período de uso de clave privada de firmante del documento	2 meses
Validez de certificado de firmante del documento	10 años + 2 meses

Elemento	***Período de uso/validez***
Período de uso de clave privada CSCA	4 años
Validez de certificado CSCA	14 años + 2 meses

Las consecuencias de este ejemplo son que para cuando el primer certificado CSCA pierde validez, se habrán expedido por lo menos 24 certificados de firmante del documento (uno para cada clave privada con período de uso de dos meses). En los últimos meses antes de que el primer certificado CSCA pierda validez, habrá por lo menos otros tres certificados CSCA expedidos (uno para cada clave privada con período de uso de cuatro años).

A.3 EJEMPLO 3

Este ejemplo final ilustra una situación donde los eMRTD son válidos por diez años y la política consiste en utilizar los máximos períodos de uso de clave privada y de validez de certificado de clave pública.

Elemento	***Período de uso/validez***
Validez del eMRTD	10 años
Período de uso de clave privada de firmante del documento	3 meses
Validez de certificado de firmante del documento	10 años + 3 meses
Período de uso de clave privada CSCA	5 años
Validez de certificado CSCA	15 años + 3 meses

Las consecuencias de este ejemplo son que para cuando el primer certificado CSCA pierde validez, se habrán expedido por lo menos 20 certificados de firmante del documento (uno para cada clave privada con período de uso de tres meses). En los últimos meses antes de que el primer certificado CSCA pierda validez, habrá por lo menos otros tres certificados CSCA expedidos (uno para cada clave privada con período de uso de cuatro años).

— — — — — — — —

Apéndice B de la Parte 12

TEXTO DE REFERENCIA PARA PERFILES DE CERTIFICADO Y CRL (INFORMATIVO)

Los perfiles de certificado y CRL que se definen en la sección 7 se basan en definiciones y requisitos de perfiles básicos especificados en los documentos de referencia. En las tablas siguientes se reproducen breves extractos de algunas secciones pertinentes de dichos documentos de referencia (en el momento de redactarse este apéndice). Estos extractos se proporcionan para ayudar al lector o lectora a comprender los antecedentes en algunos de los requisitos especificados en los perfiles de certificado eMRTD y CRL. No están previstos para basarse en ellos en vez de en los documentos de referencia. En todos los casos, para obtener la especificación completa del componente/extensión de referencia y para obtener la especificación más actual, DEBEN utilizarse los documentos de referencia propiamente dichos.

Tabla B-1. Campos y extensiones de certificado

Componente/Extensión	***Referencia***	***Extractos pertinentes***
`Certificate`	RFC 5280 – 4.1.1	
`TBSCertificate`	RFC 5280 – 4.1.1.1	
`signatureAlgorithm`	RFC 5280 – 4.1.1.2	
`signatureValue`	RFC 5280 – 4.1.1.3	
`TBSCertificate`	RFC 5280 – 4.1.2	
`version`	RFC 5280 – 4.1.2.1	Cuando se utilicen extensiones, según se prevé en este perfil, la versión DEBE ser 3 (el valor es `2`).
`serialNumber`	RFC 5280 – 4.1.2.2	El número de serie DEBE ser un entero positivo asignado por la CA a cada certificado. DEBE ser único para cada certificado expedido por una CA determinada (es decir, el nombre y el número de serie del expedidor identifican un certificado único). Las CA DEBEN imponer que el `serialNumber` sea un entero no negativo. Dado los requisitos de unicidad mencionados anteriormente, puede preverse que los números de serie contengan enteros largos. Los usuarios de certificados DEBEN poder tramitar valores `serialNumber` de hasta 20 octetos. Las CA conformes NO DEBEN utilizar valores de `serialNumber` con longitudes mayores de 20 octetos.

Componente/Extensión	***Referencia***	***Extractos pertinentes***
	X.690 – 8.3.2	Si el contenido en octetos de una codificación de valor entero es mayor que uno, entonces los bits del primer octeto y el bit 8 del segundo octeto: a) no serán todos unos; y b) no serán todos ceros. *Nota*.— Estas reglas aseguran que un valor entero siempre se codifica en el menor número posible de octetos.
	X.690 – 8.3.3	El contenido en octetos será un número binario con complemento a dos igual al valor entero y consistirá en 8 a 1 del primer octeto, seguido de los bits 8 a 1 del segundo octeto, seguido de los bits 8 a 1 de cada octeto siguiente hasta el último octeto del contenido inclusive.
`signature`	RFC 5280 – 4.1.1.2	Este campo DEBE contener el mismo identificador de algoritmo que el campo `signatureAlgorithm` en la secuencia `Certificate`.
`issuer`	RFC 5280 – Apéndice A.1	`X520countryName ::= PrintableString (SIZE (2))` `X520SerialNumber ::= PrintableString (SIZE (1..ub-serial-number))`
	RFC 5280 – 4.1.2.4	Las CA conformes con este perfil DEBEN utilizar la codificación `PrintableString` o `UTF8String` de `DirectoryString`.
	ISO 3166-1	
`validity`	RFC 5280 – 4.1.2.5	Pueden codificarse `notBefore` y `notAfter` como `UTCTime` o `GeneralizedTime`. Las CA conformes a este perfil siempre DEBEN codificar la fecha de validez del certificado hasta el año 2049 como `UTCTime`. Las fechas de validez del certificado en 2050 o posteriores DEBEN codificarse como `GeneralizedTime`.
`(if encoded as UTCTime)`	X.690 – 11.8.1	La codificación terminará con una “Z”, como se describe en UIT-T X.680 | ISO/IEC 8824-1, cláusula sobre `UTCTime`.
	X.690 – 11.8.2	El elemento segundos estará siempre presente.
`(if encoded as GeneralizedTime)`	X.690 – 11.7.1	La codificación terminará con una “Z”, como se describe en UIT-T Rec. X.680 | ISO/IEC 8824-1, cláusula sobre `GeneralizedTime`.

Componente/Extensión	***Referencia***	***Extractos pertinentes***
	X.690 – 11.7.2	El elemento segundos estará siempre presente.
	RFC 5280 – 4.1.2.5.2	Los valores `GeneralizedTime` NO DEBEN incluir fracciones de segundos. Para los fines de este perfil, los valores `GeneralizedTime` DEBEN expresarse en tiempo medio de Greenwich (Zulu) y DEBEN incluir segundos (es decir, las horas son AAAAMMDDHHMMSSZ), incluso cuando el número de segundos sea cero.
`subject`	RFC 5280 – Apéndice A.1	`X520countryName ::= PrintableString (SIZE (2))` `X520SerialNumber ::= PrintableString (SIZE (1..ub-serial-number))`
	RFC 5280 – 4.1.2.6	Las CA conformes con este perfil DEBEN utilizar la codificación `PrintableString` o `UTF8String` de `DirectoryString`.
`subjectPublicKeyInfo`	RFC 5280 – 4.1.2.7	
`issuerUniqueID`	RFC 5280 – 4.1.2.8	Las CA conformes a este perfil NO DEBEN generar certificados con identificadores únicos.
`subjectUniqueID`	RFC 5280 – 4.1.2.8	Las CA conformes a este perfil NO DEBEN generar certificados con identificadores únicos.
`extensions`	X.690 – 11.5	La codificación de un valor `set` o valor `sequence` no incluirá una codificación de ningún valor de componente que sea igual a su valor por defecto.
`AuthorityKeyIdentifier`	RFC 5280 – 4.2.1.1	El campo `keyIdentifier` de la extensión `authorityKeyIdentifier` DEBE incluirse en todos los certificados generados por las CA conformes para facilitar la construcción de la ruta de certificación. Hay una excepción: cuando una CA distribuye su clave pública en forma de certificado “autofirmado”, PUEDE omitirse el identificador de clave de la autoridad.
`keyIdentifier`		
`authorityCertIssuer`		
`authorityCertSerialNumber`		

Componente/Extensión	*Referencia*	*Extractos pertinentes*
SubjectKeyIdentifier	RFC 5280 – 4.2.1.2	Para facilitar la construcción de la ruta de certificación, esta extensión DEBE aparecer en todos los certificados CA conformes, es decir, todos los certificados incluyen la extensión de limitaciones básicas (sección 4.2.1.9), donde el valor de cA es TRUE.
subjectKeyIdentifier		
KeyUsage	RFC 5280 – 4.2.1.3	La restricción de uso podría emplearse cuando una clave que podría utilizarse por más de una operación debe restringirse.
digitalSignature		El bit digitalSignature se habilita cuando la clave pública de sujeto se utiliza con un mecanismo de firma digital para apoyar servicio de seguridad distintos a la firma del certificado (bit 5), o firma de CRL (bit 6).
nonRepudiation		
keyEncipherment		
dataEncipherment		
keyAgreement		
keyCertSign		El bit keyCertSign se habilita cuando la clave pública de sujeto se utiliza para verificar una firma en certificado de clave pública.
cRLSign		El bit cRLSign se habilita cuando la clave pública de sujeto se utiliza para verificar una firma en una lista de revocación de certificado (p. ej., una CRL, CRL delta o una ARL). Este bit DEBE habilitarse en certificados utilizados para verificar firmas en CRL.
encipherOnly		
decipherOnly		
PrivateKeyUsagePeriod	RFC 3280 – 4.2.1.4	Las CA conformes a este perfil NO DEBEN generar certificados con extensiones de período de uso de clave privada, a menos que por lo menos uno de los dos componentes esté presente y la extensión no sea crítica.
notBefore		Cuando se utilicen, notBefore y notAfter se representan como GeneralizedTime y DEBEN especificarse e interpretarse como se define en la sección 4.1.2.5.2.
notAfter		

Componente/Extensión	***Referencia***	***Extractos pertinentes***
`CertificatePolicies`	RFC 5280 – 4.2.1.4	Si esta extensión es crítica, el soporte lógico de validación de ruta DEBE ser capaz de interpretar esta extensión (incluyendo el calificador opcional), o DEBE rechazar el certificado.
`PolicyInformation`		
`policyIdentifier`		
`policyQualifiers`		
`PolicyMappings`	RFC 5280 – 4.2.1.5	
`SubjectAltName`	RFC 5280 – 4.2.1.6	
`IssuerAltName`	RFC 5280 – 4.2.1.7	
`SubjectDirectoryAttributes`	RFC 5280 – 4.2.1.8	
`Basic Constraints`	RFC 5280 – 4.2.1.9	La extensión de restricciones básicas identifica si el sujeto de certificado es una CA y la profundidad máxima de las rutas de certificación válidas que incluyen este certificado. Las CA conformes DEBEN incluir esta extensión en todos los certificados CA que contienen claves públicas utilizadas para validar firmas digitales en certificados y DEBEN indicar que la extensión es crítica en tales certificados.
`cA`		El componente `cA` booleano indica si la clave pública certificada pertenece a una CA. Si el `cA` booleano no se habilita, entonces el bit `keyCertSign` en la extensión de uso de clave NO DEBE habilitarse.
`PathLenConstraint`		
`NameConstraints`	RFC 5280 – 4.2.1.10	
`PolicyConstraints`	RFC 5280 – 4.2.1.11	
`ExtKeyUsage`	RFC 5280 – 4.2.1.12	Esta extensión indica uno o más propósitos para los cuales puede utilizarse la clave pública certificada, además o en lugar de los propósitos básicos indicados en la extensión de uso de clave.

Componente/Extensión	*Referencia*	*Extractos pertinentes*
CRLDistributionPoints	RFC 5280 – 4.2.1.13	
distributionPoint		
reasons		
cRLIssuer		
InhibitAnyPolicy	RFC 5280 – 4.2.1.14	
FreshestCRL	RFC 5280 – 4.2.1.15	
privateInternetExtensions	RFC 5280 – 4.2.2	
NameChange		
DocumentType		
Netscape Certificate Type		
other private extensions		

Tabla B-2. Campos y extensiones de CRL

Componente/Extensión	*Referencia*	*Extractos pertinentes*
CertificateList	RFC 5280 – 5.1.1	
tBSCertList	RFC 5280 – 5.1.1.1	
signatureAlgorithm	RFC 5280 – 5.1.1.2	
signatureValue	RFC 5280 – 5.1.1.3	
	RFC 5280 – 5.1.2	
version	RFC 5280 – 5.1.2.1	Este campo opcional describe la versión de la CRL codificada. Cuando se utilizan extensiones, según se exige en este perfil, este campo DEBE estar presente y DEBE especificar la versión 2 (el valor entero es 1).
signature	RFC 5280 – 5.1.2.2	Este campo DEBE contener el mismo identificador de algoritmo que el campo de firma en la secuencia CertificateList.

Componente/Extensión	***Referencia***	***Extractos pertinentes***
`issuer`	RFC 5280 – Apéndice A.1	`X520countryName ::= PrintableString (SIZE (2))` `X520SerialNumber ::= PrintableString (SIZE 1..ub-serial-number))`
	RFC 5280 – 5.1.2.3 y 4.1.2.4	Las CA conformes con este perfil DEBEN utilizar la codificación `PrintableString` o `UTF8String` de `DirectoryString`.
`thisUpdate`	RFC 5280 – 5.1.2.4	Los expedidores de CRL conformes a este perfil DEBEN codificar `thisUpdate` como `UTCTime` para las fechas hasta el año 2049. Los expedidores de CRL conformes a este perfil deben codificar `thisUpdate` como `GeneralizedTime` para fechas en el año 2050 o posteriores.
`(if encoded as UTCTime)`	X.690 – 11.8.1	La codificación terminará con una “Z”, como se describe en UIT-T X.680 | ISO/IEC 8824-1, cláusula sobre `UTCTime`.
	X.690 – 11.8.2	El elemento segundos estará siempre presente.
`(if encoded as GeneralizedTime)`	X.690 – 11.7.1	La codificación terminará con una “Z”, como se describe en UIT-T Rec. X.680 | ISO/IEC 8824-1, cláusula sobre `GeneralizedTime`.
	X.690 – 11.7.2	El elemento segundos estará siempre presente.
	RFC 5280 – 4.1.2.5.2	Los valores `GeneralizedTime` NO DEBEN incluir fracciones de segundos. Para los fines de este perfil, los valores `GeneralizedTime` DEBEN expresarse en tiempo medio de Greenwich (Zulu) y DEBEN incluir segundos (es decir, las horas son AAAAMMDDHHMMSSZ), incluso cuando el número de segundos sea cero.
`nextUpdate`	5.1.2.5	Los expedidores de CRL conformes a este perfil DEBEN codificar `nextUpdate` como `UTCTime` para fechas hasta el año 2049 inclusive. Los expedidores de CRL conformes a este perfil DEBEN codificar `nextUpdate` como `GeneralizedTime` para fechas en el año 2050 o posteriores.
`(if encoded at UTCTime)`	X.690 – 11.8.1	La codificación terminará con una “Z”, como se describe en UIT-T X.680 | ISO/IEC 8824-1, cláusula sobre `UTCTime`.
	X.690 – 11.8.2	El elemento segundos estará siempre presente.

Componente/Extensión	***Referencia***	***Extractos pertinentes***
`(if encoded at GeneralizedTime)`	X.690 – 11.7.1	La codificación terminará con una “Z”, como se describe en UIT-T Rec. X.680 | ISO/IEC 8824-1, cláusula sobre `GeneralizedTime`.
	X.690 – 11.7.2	El elemento segundos estará siempre presente.
	RFC 5280 – 4.1.2.5.2	Los valores `GeneralizedTime` NO DEBEN incluir fracciones de segundos. Para los fines de este perfil, los valores `GeneralizedTime` DEBEN expresarse en tiempo medio de Greenwich (Zulu) y DEBEN incluir segundos (es decir, las horas son AAAAMMDDHHMMSSZ), incluso cuando el número de segundos sea cero.
`revokedCertificates`	RFC 5280 – 5.1.2.6	Cuando no hay certificados revocados, la lista de certificados revocados DEBE estar ausente. De otra forma, los certificados revocados se indicarán por sus números de serie.
`crlExtensions`	RFC 5280 – 5.2	SE EXIGE que los expedidores de CRL conformes incluyan las extensiones de identificador de clave de autoridad (sección 5.2.1) y de número de CRL (sección 5.2.3) en todas las CRL expedidas.
	X.690 – 11.5	La codificación de un valor `set` o de un valor `sequence` no incluirá la codificación de ningún valor componente que sea igual a su valor por defecto.
`authorityKeyIdentifier`	RFC 5280 – 5.2.1	Los expedidores de CRL conformes DEBEN utilizar el método de identificador de clave, y DEBEN incluir esta extensión en todas las CRL expedidas.
`issuerAlternativeName`	RFC 5280 – 5.2.2	
`cRLNumber`	RFC 5280 – 5.2.3	Los expedidores de CRL conformes a este perfil DEBEN incluir esta extensión en todas las CRL y DEBEN indicar que esta extensión no es crítica. `CRLNumber ::= INTEGER (0..MAX)` Considerando los requisitos anteriores, los números de CRL podrían contener enteros largos. Los verificadores de la CRL DEBEN poder tramitar valores `CRLNumber` de hasta 20 octetos. Los expedidores de CRL conformes NO DEBEN utilizar valores de `CRLNumber` con longitudes mayores de 20 octetos.

Componente/Extensión	***Referencia***	***Extractos pertinentes***
	X.690 – 8.3.2	Si el contenido en octetos de una codificación de valor entero es mayor que uno, entonces los bits del primer octeto y el bit 8 del segundo octeto: a) no serán todos unos; y b) no serán todos ceros. *Nota*.— Estas reglas aseguran que un valor entero siempre se codifica en el menor número posible de octetos.
	X.690 – 8.3.3	El contenido en octetos será un número binario con complemento a dos igual al valor entero y consistirá en 8 a 1 del primer octeto, seguido de los bits 8 a 1 del segundo octeto, seguido de los bits 8 a 1 de cada octeto siguiente hasta el último octeto del contenido inclusive.
`deltaCRLIndicator`	RFC 5280 – 5.2.4	
`issuingDistribution Point`	RFC 5280 – 5.2.5	
`freshestCRL`	RFC 5280 – 5.2.6	
`reasonCode`	RFC 5280 – 5.3.1	
`holdInstructionCode`	RFC 5280 – 5.3.2	
`invalidityDate`	RFC 5280 – 5.3.3	
`certificateIssuer`	RFC 5280 – 5.3.4	

— — — — — — — —

Apéndice C de la Parte 12

PERFILES DE CERTIFICADO ANTERIORES (INFORMATIVO)

Los perfiles de certificado de este apéndice se especificaron en la sexta edición del Doc 9303 de la OACI. Aunque las CSCA DEBEN expedir certificados que cumplan con los actuales perfiles según se especifica en la sección 7, los perfiles anteriores se incluyen aquí con fines de información solamente, dado que los certificados expedidos con arreglo a los perfiles anteriores estarán en circulación y serán procesados por los sistemas de inspección durante varios años más.

Tabla C-1. Cuerpo del certificado

Componente del certificado	*Sección en RFC 3280*	*Certificado de CA de firma de país*	*Certificado de firmante del documento*	*Comentarios*
`Certificate`	4.1.1	m	m	
`TBSCertificate`	4.1.1.1	m	m	Véase la tabla C-2.
`SignatureAlgorithm`	4.1.1.2	m	m	El valor insertado aquí depende del algoritmo seleccionado.
`SignatureValue`	4.1.1.3	m	m	El valor insertado aquí depende del algoritmo seleccionado.
`TBSCertificate`	4.1.2			
`version`	4.1.2.1	m	m	SERÁ `v3`.
`serialNumber`	4.1.2.2	m	m	
`signature`	4.1.2.3	m	m	El valor insertado aquí CORRESPONDERÁ al OID en `signatureAlgorithm`.
`issuer`	4.1.2.4	m	m	
`validity`	4.1.2.5	m	m	Las implementaciones ESPECIFICARÁN el uso de la hora UTC hasta 2049 y de ahí en adelante utilizarán `GeneralizedTime`.
`subject`	4.1.2.6	m	m	
`subjectPublicKeyInfo`	4.1.2.7	m	m	
`issuerUniqueID`	4.1.2.8	x	x	

Componente del certificado	*Sección en RFC 3280*	*Certificado de CA de firma de país*	*Certificado de firmante del documento*	*Comentarios*
subjectUniqueID	4.1.2.8	x	x	
extensions	4.1.2.9	m	m	Véase la tabla C-2 sobre las extensiones que DEBERÍAN estar presentes.

Tabla C-2. Extensiones

Nombre de la extensión	*Párrafo en RFC 3280*	*Certificado de CA de firma de país*	*Certificado de firmante del documento*	*Comentarios*
AuthorityKeyIdentifier	4.2.1.1	o	m	Obligatorio en todos los certificados salvo los certificados CSCA autofirmados.
SubjectKeyIdentifier	4.2.1.2	m	o	
KeyUsage	4.2.1.3	mc	mc	Esta extensión SE INDICARÁ como CRÍTICA.
PrivateKeyUsagePeriod	4.2.1.4	o	o	Este sería el período de expedición de la clave privada.
CertificatePolicies	4.2.1.5	o	o	
PolicyMappings	4.2.1.6	x	x	
SubjectAltName	4.2.1.7	x	x	
IssuerAltName	4.2.1.8	x	x	
SubjectDirectoryAttributes	4.2.1.9	x	x	
BasicConstraints	4.2.1.10	mc	x	Esta extensión SE INDICARÁ como CRÍTICA.
NameConstraints	4.2.1.11	x	x	
PolicyConstraints	4.2.1.12	x	x	
ExtKeyUsage	4.2.1.13	x	x	

Nombre de la extensión	***Párrafo en RFC 3280***	***Certificado de CA de firma de país***	***Certificado de firmante del documento***	***Comentarios***
CRLDistributionPoints	4.2.1.14	o	o	Si los Estados expedidores u organizaciones expedidoras optan por usar esta extensión INCLUIRÁN el PKD de la OACI como punto de distribución. Las implementaciones también pueden incluir DP de CRL relativos para fines locales; los otros Estados receptores pueden ignorar estos DP.
InhibitAnyPolicy	4.2.1.15	x	x	
FreshestCRL	4.2.1.16	x	x	
privateInternetExtensions	4.2.2	x	x	
other private extensions	N/A	o	o	Si se incluye una extensión privada para fines nacionales, esta NO SE INDICARÁ. Se pide a los Estados expedidores u organizaciones expedidoras que no incluyan extensiones privadas.
AuthorityKeyIdentifier	4.2.1.1			
keyIdentifier		m	m	Si se utiliza esta extensión, SE ADMITIRÁ como mínimo este campo.
authorityCertIssuer		o	o	
authorityCertSerialNumber		o	o	
SubjectKeyIdentifier	4.2.1.2			
subjectKeyIdentifier		m	m	
KeyUsage	4.2.1.3			
digitalSignature		x	m	
nonRepudiation		x	x	
keyEncipherment		x	x	

Nombre de la extensión	*Párrafo en RFC 3280*	*Certificado de CA de firma de país*	*Certificado de firmante del documento*	*Comentarios*
dataEncipherment		x	x	
keyAgreement		x	x	
keyCertSign		m	x	
cRLSign		m	x	
encipherOnly		x	x	
decipherOnly		x	x	
BasicConstraints	4.2.1.10			
cA		m	x	TRUE (VERDADERA) para certificados CA.
PathLenConstraint		m	x	0 para nuevos certificados CSCA, 1 para certificado de enlace CSCA.
CRLDistributionPoints	4.2.1.14			
distributionPoint		m	x	
reasons		m	x	
cRLIssuer		m	x	
CertificatePolicies	4.2.1.5			
PolicyInformation				
policyIdentifier		m	m	
policyQualifiers		o	o	

— — — — — — — —

Apéndice D de la Parte 12

COMPATIBILIDAD DE VALIDACIÓN EN RFC 5280 (INFORMATIVO)

En este apéndice se proporciona orientación a los Estados receptores que deseen emplear sistemas que implementen la ruta de certificación de [RFC 5280] y sus algoritmos de validación de CRL.

El modelo de confianza de la PKI del eMRTD es un subconjunto del abarcado por los procedimientos de validación definidos en [RFC 5280]. En la sección D.1 se identifica el subconjunto de etapas de la definición de [RFC 5280] requerido para la aplicación del eMRTD y se proporcionan las entradas necesarias y los valores y procesos de inicialización para la validación de la ruta de certificación, la validación de CRL y la verificación de revocación.

La sección D.2 abarca las etapas restantes de la definición de [RFC 5280] y no son pertinentes a la aplicación del eMRTD. Se proporcionan las entradas y valores de inicialización para la validación de ruta de certificación y de CRL. La orientación en esta sección se aplica en situaciones en que los mecanismos implementan los algoritmos [RFC 5280] completos, en vez de solamente el subconjunto que se describe en D.1.

En la sección D.3 se proporciona orientación para apoyar la extensión del procesamiento CRL basado en [RFC 5280] para abarcar la verificación de la revocación después de que una CSCA ha experimentado un cambio de nombre.

D.1 ETAPAS PERTINENTES A LOS eMRTD

El procedimiento de validación de ruta de certificación de eMRTD definido aquí se basa en el procedimiento que se describe en [RFC 5280]. Los perfiles de certificado eMRTD restringen las rutas de certificación a un único certificado y prohíben el uso de muchas características opcionales que se emplean en otras aplicaciones, como la PKI de Internet definida en [RFC 5280]. Las etapas de validación de ruta relacionadas con estas características se omiten del procedimiento de validación de ruta de certificación para eMRTD.

D.1.1 Procedimiento de validación de la ruta de certificación

D.1.1.1 Entradas

En la [RFC 5280] se define un conjunto de nueve entradas para el algoritmo de validación de ruta. Solo las tres siguientes son pertinentes a la aplicación del eMRTD:

- ruta de certificación: un certificado único (p. ej., el certificado de firmante del documento);
- fecha/hora actuales; e
- información sobre punto de confianza, incluido:
    - nombre del expedidor de confianza: si el punto de confianza está en forma de certificado CSCA, el nombre del expedidor de confianza es el valor del campo `subject` de dicho certificado;

- algoritmo de clave pública de confianza: si el punto de confianza está en forma de certificado CSCA, el algoritmo de clave pública de confianza se toma del campo `SubjectPublicKeyInfo` de dicho certificado;

- clave pública de confianza: si el punto de confianza está en forma de certificado CSCA, la clave pública de confianza se toma del campo `SubjectPublicKeyInfo` de dicho certificado; y

- parámetros de clave pública de confianza: esta es una entrada opcional que se incluye solamente si el algoritmo de clave pública de confianza requiere parámetros. Si el punto de confianza está en forma de certificado CSCA, estos parámetros se toman del campo `SubjectPublicKeyInfo` de dicho certificado.

Si una implementación requiere que se suministren las seis entradas adicionales, en D.2 figuran recomendaciones al respecto.

Puede haber varios puntos de confianza para la CSCA que expidió el certificado que se está validando. De estos puntos de confianza, el que DEBE utilizarse es el que contiene la clave pública que corresponde al valor de la extensión del identificador de clave de autoridad en el certificado que se está validando.

D.1.1.2 Inicialización

Hay once variables de Estado definidas en [RFC 5280]. Solo las cinco siguientes son pertinentes a la aplicación del eMRTD:

- aplicación: max_path_length: inicializar a “0”;

- working_issuer_name: inicializar al valor del nombre del expedidor de confianza;

- working_public_key_algorithm: inicializar al valor del algoritmo de clave pública de confianza;

- working_public_key: inicializar al valor de la clave pública de confianza; y

- working_public_key_parameters: inicializar al valor de los parámetros de clave pública de confianza.

Si una implementación requiere que se inicialicen las seis variables adicionales, en D.2 figuran recomendaciones al respecto.

D.1.1.3 Procesamiento de certificados

Las etapas de procesamiento de certificados eMRTD son un subconjunto de las definidas en [RFC 5280]. El resultado de procesamiento de un certificado eMRTD utilizando este procedimiento simplificado será coherente con el resultado de aplicar el algoritmo RFC 5280 completo. Si las entradas adicionales y variables de Estado se configuran como se describe en D.2:

a) Verificar la información de certificado básica. El certificado DEBE satisfacer cada uno de los siguientes aspectos:

   - la firma del certificado puede verificarse utilizando working_public_key_algorithm, working_public_key y working_public_key_parameters;

- el período de validez de certificado incluye la hora actual;
- en el momento actual, el certificado no está revocado (en 6.3 figura información detallada al respecto); y
- el nombre del expedidor de certificado es el working_issuer_name.

b) Asignar el `subjectPublicKey` del certificado al working_public_key.

c) Si el campo `subjectPublicKeyInfo` del certificado contiene un campo de algoritmo con parámetros que no son nulos, asignar los parámetros a la variable working_public_key_parameters. Si el campo `subjectPublicKeyInfo` del certificado contiene un campo de algoritmo con parámetros nulos o se han omitido parámetros, comparar el algoritmo `subjectPublicKey` del certificado con el algoritmo working_public_key_algorithm. Si el algoritmo `subjectPublicKey` del certificado y el algoritmo working_public_key_algorithm son diferentes, poner nulo a los working_public_key_parameters.

d) Asignar el algoritmo `subjectPublicKey` de certificado a la variable working_public_key_algorithm.

e) Reconocer y procesar cualquier otra extensión crítica presente en el certificado.

f) Procesar cualquier otra extensión no crítica reconocida presente en el certificado.

Si alguna de las verificaciones en la etapa a) falla o si hay extensiones críticas no reconocidas en el certificado que no puedan procesarse, el procedimiento de validación de ruta también falla. De otra forma, el procedimiento tiene éxito.

D.1.1.4 Salidas

Si la validación dela ruta tiene éxito, el procedimiento termina, devolviendo una indicación de éxito junto con working_public_key, working_public_key_algorithm y la working_public_key_parameters.

Si la validación de la ruta falla, el procedimiento termina, devolviendo una indicación de falla y un motivo apropiado.

D.1.2 Validación y verificación de revocación de las CRL

El algoritmo de validación de CRL que figura en [REC 5280] abarca varios tipos de CRL incluyendo las delta CRL, las CRL particionadas, las CRL indirectas, etc. El perfil de CRL para la aplicación del eMRTD es muy restrictivo y prohíbe el uso de cualquiera de estas características. El uso de la extensión `issuingDistributionPoint` así como todas las extensiones normalizadas con entradas de CRL también se prohíbe. Como resultado, la validación y verificación de revocación de las CRL para la aplicación del eMRTD es relativamente sencilla.

D.1.2.1 Entradas

En [RFC 5280] se definen dos entradas para el algoritmo de validación de CRL. Solo la siguiente, entre ellas, es pertinente a la aplicación del eMRTD. Si una implementación requiere que se suministre la entrada adicional, en D.2 figura una recomendación al respecto.

- certificate: número de serie y nombre del expedidor del certificado.

D.1.2.2 Inicialización

Hay tres variables del Estado que se definen en [RFC 5280]. Solo la siguiente, entre ellas, es pertinente a la aplicación del eMRTD. Si una implementación requiere que las dos variables adicionales sean inicializadas, en D.2 figuran recomendaciones para ello.

- cert_status: inicializar al valor UNREVOKED.

D.1.2.3 Procesamiento de la CRL

Todas las CRL en la aplicación del eMRTD son CRL completas que abarcan todos los certificados actuales expedidos por la CSCA que expidió la CRL. No hay CRL particionadas, delta o indirectas. Las etapas del algoritmo de procesamiento de las CRL para la aplicación del eMRTD son:

a) Obtener la CRL actual de la CSCA que expidió el certificado. Si no puede obtenerse dicha CRL, la variable cert_status variable se establece en UNDETERMINED y se detiene el procesamiento.

b) Verificar que el expedidor de la CRL es la misma CSCA que expidió el certificado en cuestión. Debido a que hay una única CSCA en cada país, y la aplicación del eMRTD es una aplicación cerrada donde los sistemas de inspección conservan una caché de CRL que es única para esta aplicación, alcanza con verificar que el nombre del país es el mismo en el campo de expedidor de la CRL y en el campo del expedidor del certificado.

   - Si la CSCA no ha experimentado un cambio de nombre desde que se expidió el certificado, el campo de expedidor en la CRL y el campo de expedidor en el certificado serán idénticos.

   - Si la CSCA ha experimentado un cambio de nombre desde que el certificado fue expedido, el atributo país del nombre en el campo de expedidor de certificado y en el campo de expedidor de las CRL serán los mismos, pero algunos otros atributos pueden ser diferentes.

   - Si la parte que confía desea verificar que no ha ocurrido sustitución de algunas CRL que no son de eMRTD, tiene la opción de verificar que tiene puntos de confianza para ambos nombres de CSCA y que dichos puntos de confianza son para la misma CSCA. Si la CSCA ha experimentado un cambio de nombre y ha incluido en la CRL la extensión opcional `issuerAltName`, la parte que confía PUEDE opcionalmente verificar que el campo de expedidor en el certificado es idéntico a uno de los valores de esta extensión.

Si el expedidor de la CRL no es la CSCA que expide el certificado, la variable cert_status se establece en UNDETERMINED, y el procesamiento se detiene.

c) Validar la ruta de certificación para el expedidor de la CRL. Obsérvese que en la aplicación del eMRTD todas las CRL son expedidas por CSCA que son los puntos de confianza para las respectivas rutas. A diferencia del algoritmo de [RFC 5280], la aplicación del eMRTD NO requiere que el punto de confianza utilizado para validar la ruta de certificación de la CRL sea el mismo punto de confianza que se usó para validar el certificado de que se trata. No obstante, si los puntos de confianza son diferentes, ambos DEBEN ser puntos de confianza para la misma CSCA. A diferencia de la [RFC 5280], la aplicación del eMRTD tiene múltiples puntos de confianza para una determinada CSCA que son válidos al mismo tiempo. Si la ruta de certificación no puede validarse satisfactoriamente, la variable cert_status se establece en UNDETERMINED y el procesamiento se detiene.

d) Verificar la firma en la CRL. Si la firma no puede verificarse satisfactoriamente, la variable cert_status se establece en UNDETERMINED y el procesamiento se detiene.

e) Buscar el certificado en la CRL. Si se encuentra una entrada que corresponde al expedidor de certificado y al número de serie, entonces la variable cert_status se establece en UNSPECIFIED.

D.1.2.4 Salida

Devolver el cert_status. Si las etapas a), b), c) o d) fallaron, el estado será UNDETERMINED. Si el certificado fue incluido en la CRL como revocado, el estado será UNSPECIFIED. Si la validación de la CRL tuvo éxito, pero el certificado no estaba incluido en la CRL, el estado será UNREVOKED.

D.2 ETAPAS NO REQUERIDAS POR eMRTD

D.2.1 Validación de la ruta de certificación

Los parámetros para entradas adicionales que no son pertinentes a la validación de eMRTD comprenden:

- inhibición de la correspondencia de política inicial: establecer en inhibir correspondencia de política;
- inhibición de cualquier política inicial: establecer en inhibir procesamiento del valor cualquier política;
- subárboles iniciales permitidos: establecer en permitir todos los subárboles;
- subárboles iniciales excluidos: establecer en no excluir subárboles;
- política inicial explícita: NO debería establecerse; y
- conjunto de política inicial de usuario: establecer en el valor especial “cualquier política”.

La inicialización de las variables del Estado que no son pertinentes a la aplicación del eMRTD comprenden:

- permitted_subtrees (subárboles permitidos): inicializar a todos los subárboles permitidos;
- excluded_subtrees (subárboles excluidos): inicializar a no excluir subárboles;
- inhibit_any_policy (inhibición de cualquier política): si se establece la inhibición inicial de cualquier política, inicializar a “0”. De otra manera, establecer el valor 1 o cualquier valor mayor que ese;
- policy_mapping (correspondencia de política): inicializar a “0”;
- explicit_policy (política explícita): inicializar a “2”; y
- valid_policy_tree (árbol de política válido): inicializar el elemento valid_policy en “anyPolicy”, el elemento qualifier_set en vacío y el elemento expected_policy_set en “anyPolicy”.

D.2.2 Validación de la CRL

Los parámetros para entradas adicionales que no son pertinentes a la validación de eMRTD comprenden:

- use-deltas: se establece para prohibir el uso de deltas.

La inicialización de las variables del Estado que no son pertinentes a la aplicación del eMRTD comprenden:

- reasons_mask (enmascarar razones): inicializar en un conjunto vacío; y
- Interim_reasons_mask (enmascarar razones provisionales): inicializar al valor especial “all-reasons”.

D.3 MODIFICACIONES REQUERIDAS PARA PROCESAR LAS CRL

Los sistemas de validación de CRL que cumplen con el procedimiento de validación de CRL en [RFC 5280] no tienen por objeto apoyar entornos donde una CA ha experimentado un cambio de nombre, como el entorno de aplicación del eMRTD. Por consiguiente, estos sistemas requieren alguna modificación para tramitar este caso especial, como se describe a continuación:

a) En la cláusula 6.3.3, etapa a) del procedimiento de validación de CRL en [RFC 5280], el nombre en el campo de punto de distribución de la extensión de punto de distribución de la CRL del certificado en cuestión se utiliza para actualizar la caché local con las CRL pertinentes. Para la aplicación del eMRTD, esta etapa tendría que modificarse y solo el atributo `countryName` del campo de punto de distribución debería utilizarse para identificar y obtener la CRL apropiada.

b) En la cláusula 6.3.3, etapa f) del procedimiento de validación de CRL en [RFC 5280], se establece el requisito de que el mismo punto de confianza se utilice para validar la ruta de certificación para el expedidor de CRL que se utilizó para validar el certificado de que se trata. Esto NO es un requisito para la aplicación del eMRTD porque se establecen puntos de confianza independientes para cada clave pública de la CSCA.

El punto de confianza empleado para la validación del expedidor de la CRL será el punto de confianza para la clave pública CSCA que corresponda a la clave privada utilizada para firmar la CRL. El punto de confianza utilizado para validar la ruta de certificación para el certificado en cuestión puede corresponder a un par de claves CSCA anteriores.

— — — — — — — —

Apéndice E de la Parte 12

EJEMPLO DE LDS2 (INFORMATIVO)

En el ejemplo siguiente se ilustran las interacciones entre los diferentes componentes de la PKI de firma LDS2 y la PKI de autorización LDS2.

Para ilustrar las interacciones y los elementos preliminares que se necesitan para un caso hipotético típico, consideraremos una situación en que el país de Distopía desea escribir sellos de viaje en el pasaporte de las ciudadanas y ciudadanos del país de Utopía. Posteriormente, el país de Atlántida desea leer los sellos de viaje escritos por Distopía en los pasaportes de Utopía.

Los elementos preliminares son los siguientes:

- Utopía ha instalado una aplicación de sello de viaje LDS2 en sus pasaportes.
- Tanto Distopía como Utopía han establecido su PKI de autorización LDS2.
- Distopía ha establecido su PKI de firma LDS1 de forma que expida certificados de firmante LDS2.
- Los certificados CVCA y los certificados de cliente y servidor SPOC se intercambiaron de forma fiable entre Utopía y Distopía en algún momento (posteriormente, se pueden intercambiar nuevos certificados CVCA y SPOC directamente a través del SPOC).
- Los certificados CVCA y los certificados de cliente y servidor SPOC se intercambiaron de forma fiable entre Utopía y Atlántida en algún momento (posteriormente, se pueden intercambiar nuevos certificados CVCA y SPOC directamente a través del SPOC). Si la aplicación de los sellos de viaje LDS2 es de lectura abierta, es decir, que cualquier país puede leer los sellos de viaje LDS2 (solo se necesita permiso para escribir), este paso puede omitirse.
- Los certificados CSCA se han intercambiado de forma fiable entre Distopía y Atlántida en algún momento.

El proceso recurrente para que Distopía pueda sellar electrónicamente los eMRTD de Utopía es el siguiente:

- Distopía solicita un certificado DV a Utopía.
- El SPOC de Distopía utiliza su certificado de cliente SPOC y el certificado de servidor SPOC de Utopía para iniciar una conexión SPOC. A continuación, un DV de Distopía genera una solicitud, que se envía de SPOC a SPOC. Al recibir la solicitud, Utopía genera un certificado DV extranjero con acceso de lectura/escritura para Distopía, y el certificado se vuelve a enviar de SPOC a SPOC.
- Al recibir el certificado DV de su SPOC, el DV de Distopía genera certificados de terminal para los terminales de sus fronteras. Cuando se conecta con el pasaporte, el CI de los pasaportes de Utopía verifica el certificado de terminal de Distopía con el certificado DV de Distopía, y el certificado DV de Distopía con el certificado CVCA de Utopía. A continuación, el CI concede acceso de lectura/escritura del terminal de Distopía a la aplicación de sellos de viaje LDS2.

El proceso para sellar electrónicamente un eMRTD es el siguiente:

- Distopía crea un sello de viaje electrónico y lo firma con la clave privada correspondiente a la clave pública almacenada en un certificado de firmante LDS2 (sello de viaje) de la PKI de firma LDS2 de Distopía. El certificado de firmante LDS2 se almacena en el IC sin contacto del pasaporte de Utopía.

Cuando el pasaporte de Utopía llega a la frontera de Atlántida:

- Si la lectura de los sellos de viaje de los pasaportes de Utopía requiere un certificado de terminal con acceso de lectura, se envía una solicitud de certificado de Atlántida a Utopía de SPOC a SPOC. Al recibir la solicitud, Utopía genera un certificado DV extranjero con acceso de lectura para Atlántida y envía este certificado a Atlántida de SPOC a SPOC. A partir de ese certificado DV, Atlántida genera certificados de terminal con acceso de lectura para los pasaportes de Utopía en los terminales de Atlántida. Si los sellos de viaje de los pasaportes de Utopía pueden ser leídos por cualquier terminal, este paso puede omitirse.

- Para verificar un sello de viaje del pasaporte escrito por Distopía, Atlántida utiliza la PKI de firma LDS1 de Distopía: El certificado de firmante LDS2 de Distopía almacenado en el pasaporte se utiliza para verificar el sello de viaje. A continuación, se crea la cadena, es decir, el certificado de firmante LDS2 de Distopía se verifica con el certificado CSCA de Distopía recibido previamente.

— FIN —

ISBN 978-92-9265-540-2