UNIVERSIDAD DON BOSCO
VICERRECTORÍA ACADÉMICA
FACULTAD DE INGENIERÍA

TRABAJO DE GRADUACIÓN PARA OPTAR AL GRADO DE

Maestro en Seguridad y Gestión de Riesgos Informáticos

PROYECTO

Aplicación para gestionar certificados digitales como Autoridad Certificadora (AC) de acuerdo a la Ley de firma electrónica de El Salvador

PRESENTADO POR

Álvaro Hernán Zavala Ruballo

Leonel Antonio Maye Menjívar

ASESOR

Dr. Francisco Rodríguez-Henríquez

Antiguo Cuscatlán, La Libertad, El Salvador, Centro América

Enero 2020

ÍNDICE GENERAL

Índice de Tablas

Índice de Figuras

Introducción

Los certificados de llave pública o certificados digitales son estructuras de datos que unen valores de llave pública a los sujetos. El enlace se confirma al hacer que una Autoridad Certificadora (AC o CA[1]) de confianza firme digitalmente cada certificado (IETF, RFC 5280 2018).

Una AC es una entidad de confianza que gestiona los certificados digitales desde su emisión hasta su revocación, utilizando en ellos la firma electrónica, para lo cual se emplea la criptografía de llave pública, las tareas de emisión y revocación de certificados las puede instar el titular del certificado o cualquier tercero con interés legítimo ante la AC

El presente trabajo propone una aplicación para la gestión de certificados digitales basándose en La Ley de firma electrónica de El Salvador, software que debe ser implementado por una AC.

La investigación se desglosa de la siguiente forma:

Capítulo I. Planteamiento del problema

El primer capítulo contiene la delimitación y planteamiento del problema de investigación, los objetivos de la investigación y justificación.

Capítulo II. Marco de la Investigación

En este capítulo se describe el Estado del Arte de la base teórica que fundamenta el tema de investigación.

Capítulo III. Metodología de la Investigación

En dicho capítulo, se encuentra el tipo de investigación, unidades de análisis, variables y su medición.

Capítulo IV. Análisis y discusión de resultados

Se analizan los requerimientos técnicos funcionales que deben ser incorporados en la aplicación a desarrollar

[1] Certificate Authority

Capítulo IV. Propuesta técnica

Presenta el desarrollo por fases, mediante la metodología Scrum, de la aplicación para la gestión de certificados digitales.

Capítulo VI. Discusión de seguridad y aplicabilidad

Se hace una discusión y recomendaciones de aspectos importantes de seguridad que la aplicación implementa, así como algunas áreas de aplicación.

Capítulo IV. Conclusiones y recomendaciones

Capítulo que contiene las conclusiones y recomendaciones de la investigación.

CAPÍTULO I

PLANTEAMIENTO DEL PROBLEMA

Capitulo I. Planteamiento del Problema

1.1 Problema de investigación

En El Salvador actualmente se están realizando esfuerzos para implementar la firma electrónica y como parte de esta tecnología se requiere de una infraestructura de llave pública (PKI)[2], la cual deberá validar a los Prestadores de Servicios de Certificación (AC). Para una AC es necesario implementar el software que le permita gestionar los certificados digitales y realizar los procedimientos de seguridad para la ejecución con garantías de las operaciones criptográficas, como el cifrado, la firma digital, y el no repudio de transacciones electrónicas.

1.2 Antecedentes del problema

La Ley de Firma Electrónica de El Salvador fue aprobada en 2015 (Cerén, Ley de Firma Electrónica 2015). El Ministerio de Economía fue designado como la Autoridad Certificadora Raíz o Root CA en inglés y en abril de 2016 entró en vigencia.

En octubre de 2016 se crea el Reglamento de la Ley de Firma Electrónica (Cerén, Reglamento de la Ley de Firma Electrónica, 2016). A continuación, se forma el Comité Técnico Consultivo en julio 2017, con el objetivo de asesorar la implementación del proyecto, este está conformado por gobierno, empresa y universidades (MINEC 2017).

Para la supervisión y control se crea la Unidad de Firma Electrónica en 2016, (STPP 2017) que será la autoridad registradora y acreditadora raíz, y la competente para la acreditación, control y vigilancia de los proveedores de los servicios de certificación

[2] Public Key Infrastructure

electrónica y de almacenamiento de documentos electrónicos, de conformidad con la ley, su reglamento y las normas y reglamentos técnicos.

La factura electrónica en El Salvador es un proyecto que ha sido aprobado y uno de los bloques principales para su implementación es la firma electrónica.

Con este esfuerzo modernizador de América Latina, en el siglo XXI se ha implementado la factura electrónica, iniciando con Chile en 2003 y a mediados de 2017, se suman otras experiencias avanzadas en Argentina, Brasil, Ecuador, México, Perú y Uruguay. Actualmente, existen proyectos en proceso en varios países de la región latinoamericana, entre ellos: Costa Rica, Colombia, Guatemala, Panamá y Paraguay, y se ha manifestado la intención de desarrollar sistemas nacionales en Honduras, República Dominicana, Venezuela y por supuesto en El Salvador (BCR 2019).

De acuerdo al Plan Estratégico Institucional 2015-2019 publicado en abril 2019, por el Ministerio de Hacienda de El Salvador, uno de los proyectos que se está llevando a cabo en el país es el de la Factura Electrónica, que obedece al objetivo estratégico de implementar una política tributaria progresiva que genere el cumplimiento voluntario de las obligaciones tributarias, mediante el fortalecimiento de los controles de la Administración Tributaria. les para su implementación es la firma electrónica (BCR 2019).

Sobre las AC, la Ley de Firma Electrónica establece la equivalencia jurídica únicamente aquellas autoridades certificadoras que cumplan los requisitos técnicos emanados por la Unidad de Firma Electrónica y que se sometan a la evaluación para respaldar la competencia técnica y credibilidad de los entes acreditados.

A 2020, se puede afirmar que se están realizando esfuerzos importantes para la implementación de la firma y factura electrónica, pero, aún no se cuenta hasta la fecha con una Infraestructura de Llave Pública proporcionada por el Ministerio de Economía, como AC Raíz, y por la limitación mencionada, tampoco existen Autoridades

Certificadoras acreditadas para brindar los servicios de expedición y revocación de certificados digitales.

1.3 Objetivos del proyecto

Objetivo general

Desarrollar aplicación para gestionar certificados digitales como Autoridad Certificadora (AC) de acuerdo a la Ley de firma electrónica de El Salvador.

Objetivos específicos

- Establecer los requerimientos de la aplicación de gestión de certificados.
- Definir y aplicar los algoritmos criptográficos con las garantías de seguridad en la ejecución de las operaciones.
- Crear módulos para la expedición y revocación de certificados.

1.4 Justificación del proyecto

Uno de los principales desafíos a que se enfrentan los medios telemáticos es asegurar la identidad de las partes que intervienen en cualquier operación. La solución adoptada para garantizar la seguridad en el uso de medios electrónicos está basada en la criptografía.

Los certificados digitales son un bloque importante en los esquemas de firma electrónica para garantizar el no repudio de operaciones electrónicas y están basados en la utilización de técnicas de criptografía asimétrica, en la que existe una llave pública a disposición de todo el mundo y que sirve para identificar al titular del certificado y una

llave privada que solamente conoce el titular del certificado y le sirve para firmar electrónicamente.

Esta investigación busca proporcionar una aplicación para la gestión de certificados digitales para llevar a cabo los procedimientos de seguridad criptográficos de una AC y establecer una base fundamental de cómo debería funcionar en El Salvador, esto para ser tomada en cuenta por las entidades que deseen certificarse como prestadores de estos servicios.

1.5 Delimitación del Proyecto

- El proyecto está enmarcado en un plazo de 6 meses.
- Es aplicable únicamente considerando la Ley de Firma Electrónica de El Salvador.
- El proyecto no abarca el establecimiento de políticas y procedimientos para registrarse y acreditarse como una AC ante el MINEC.
- No se considera el manejo de pagos para la obtención de certificados, se deja para futuros proyectos.

CAPÍTULO II

ESTADO DEL ARTE

Capitulo II. Estado del arte

2.1 Criptografía

La criptografía utiliza técnicas matemáticas para codificar mensajes y hacerlos ininteligibles para receptores no autorizados. Es el estudio de técnicas matemáticas relacionadas con aspectos de seguridad de la información, tales como confidencialidad, integridad de datos, autenticación de entidades y autenticación de origen de datos (A. Menezes 1996).

La criptografía no es el único medio para proporcionar seguridad de la información, sino más bien un conjunto de técnicas.

2.1.1 Objetivos de la criptografía

1. La confidencialidad es un servicio utilizado para mantener protegido el contenido de la información de todos menos aquellos autorizados para tenerlo (A. Menezes 1996). El secreto es un término sinónimo de confidencialidad y privacidad. Existen numerosos enfoques para proporcionar confidencialidad, desde protección física hasta algoritmos matemáticos que hacen que los datos sean ininteligibles.

2. La integridad de los datos es un servicio que aborda la alteración no autorizada de los datos (ISACA 2017). Para asegurar la integridad de los datos, se debe tener la capacidad de detectar la manipulación de datos por parte de personas no autorizadas. La manipulación de datos incluye cosas tales como inserción, eliminación y sustitución.

3. La autenticación es un servicio relacionado con la identificación. Esta función se aplica tanto a las entidades como a la información misma. Dos partes que inician una comunicación deben identificarse entre sí. La información entregada a través de un canal debe autenticarse en cuanto a su origen, fecha de origen, contenido

de datos, hora de envío, etc. Por estas razones, este aspecto de la criptografía generalmente se subdivide en dos clases principales: autenticación de entidad y autenticación de origen de datos. La autenticación del origen de datos proporciona implícitamente la integridad de los datos (porque si se modifica un mensaje, la fuente ha cambiado) (A. Menezes 1996).

4. El no repudio es un servicio que evita que una entidad niegue compromisos o acciones anteriores (ISACA 2017). Cuando surgen disputas debido a que una entidad niega que se hayan tomado ciertas acciones, es necesario un medio para resolver la situación. Por ejemplo, una entidad puede autorizar la compra de bienes por otra entidad y luego negar que se haya otorgado dicha autorización. Se necesita un procedimiento que involucre a un tercero de confianza para resolver la disputa.

2.1.2 Definición de criptosistema

Un criptosistema es una quíntupla (P, C, K, E, D) donde las siguientes condiciones se cumplen (Stinson 1995):

- P es el espacio de textos en claro.
- C es el espacio de textos cifrados.
- K es el espacio de llaves.
- Para cada llave $k \in K$ existe una función de cifrado $e_k \in E$ y su respectiva función de descifrado $d_k \in D$. donde e_k: P → C y d_k: C → P y $d_k(e_k(x)) = x$ para toda $x \in P$.

La criptografía puede dividirse en un numero básico de herramientas criptográficas (primitivas) usadas para proveer seguridad entre ellas las involucradas en este proyecto se describen a continuación.

2.2 Criptografía simétrica

La criptografía simétrica (también conocida como de llave secreta) es un método criptográfico monollave, esto quiere decir que se usa la misma llave para cifrar y descifrar. Esto supone un grave problema a la hora de realizar el intercambio entre el emisor y el receptor, dado que si una tercera persona estuviese escuchando el canal podría hacerse con la llave, siendo inútil el cifrado (Stalling 2011).

Es importante que la llave sea difícil de adivinar y el método de cifrado empleado el adecuado. Hoy en día, con la capacidad computacional disponible, si se emplean los algoritmos adecuados, dependiendo del método de cifrado empleado se puede obtener una llave en cuestión de minutos-horas.

2.2.1 AES

El algoritmo AES (Advanced Encryption Standard) también conocido como Rijndael fue el ganador del concurso convocado en el año 1997 por el NIST (Instituto Nacional de Normas y Tecnología) con objetivo de escoger un nuevo algoritmo de cifrado. En 2001 fue tomado como FIPS y en 2002 se transformó en un estándar efectivo. Desde el año 2006 es el algoritmo más popular empleado en criptografía simétrica (Stalling 2011).

AES opera sobre una matriz de 4x4 bytes. Mediante un algoritmo se reordenan los distintos bytes de la matriz. El cifrado es de llave simétrica, por lo que la misma llave aplicada en el cifrado se aplica para el descifrado.

Basado en El algoritmo Rijndael, al contrario que su predecesor DES, Rijndael es una red de sustitución permutación, no una red de Feistel. AES es rápido tanto en software como en hardware, es relativamente fácil de implementar, y requiere poca memoria (NIST, Advanced Encryption Standard 2001).

El algoritmo AES funciona mediante una serie de bucles que se repiten. 10 ciclos para llaves de 128 bits, 12 para 192 y 14 para 256.

2.3 Criptografía asimétrica

La criptografía asimétrica (también conocida como de llave pública) es un sistema que emplea un par de llaves. Este par de llaves pertenecen a la misma persona. Una es de dominio público y cualquiera puede tenerla y la otra es privada (Stalling 2011). El funcionamiento de este sistema es el siguiente:

El remitente usa la llave pública del destinatario para cifrar un mensaje y sólo con la llave privada se podrá descifrar el mensaje. De esta forma se consigue que sólo el destinatario pueda acceder a la información. De la misma forma si el propietario usa su llave privada para cifrar un mensaje sólo se podrá descifrar con la llave pública. Pero, si todo el mundo puede tener acceso a la llave pública ¿Que utilidad tiene esto? Precisamente por esto es interesante el sistema. Usando la llave privada se demuestra la identidad, pues, en teoría, solo el poseedor de esa llave privada la puede usar. La mayor ventaja de este sistema es que la distribución de llaves es más fácil y segura que usando su contraparte que es la criptografía simétrica donde solo existe una llave secreta para cifrar y descifrar. Sin embargo, este sistema tiene varias desventajas (A. Menezes 1996):

- Mayor tiempo de proceso en mismas condiciones respecto a llave simétrica.
- Llaves más grandes que en sistemas simétricos.
- El mensaje cifrado es más grande que el original.

Actualmente para firma digital se utiliza criptografía asimétrica y se enfoca en algoritmos como: RSA y Curvas Elípticas.

2.3.1 Algoritmo de curvas elípticas (ECC)[3]

Algoritmo de curvas elípticas (ECC) es un término utilizado para describir un conjunto de herramientas criptográficas y protocolos cuya seguridad se basa en versiones especiales del problema del logaritmo discreto. No usa números módulo *p* en relación al algoritmo RSA (Stalling 2011).

ECC se basa en conjuntos de números que están asociados con objetos matemáticos llamados curvas elípticas.

ECC incluye una variedad de muchos esquemas criptográficos que fueron diseñados inicialmente para números modulares como el cifrado ElGamal y el algoritmo de firma digital.

Se cree que el problema del logaritmo discreto es mucho más difícil cuando se aplica a puntos en una curva elíptica.

Las llaves más cortas resultan en dos beneficios (Nakamoto 2010):

- Facilidad de gestión de llaves
- Computación eficiente

Estos beneficios hacen que las variantes de esquema de cifrado basadas en curvas elípticas sean muy atractivas para aplicaciones donde los recursos informáticos están restringidos.

A continuación, se muestra un ejemplo grafico de una curva elíptica de secp256k1 usada por la criptomoneda Bitcoin $y^2 = x^3 + 7$ sobre los números reales.

[3] Elliptic Curve Cryptography

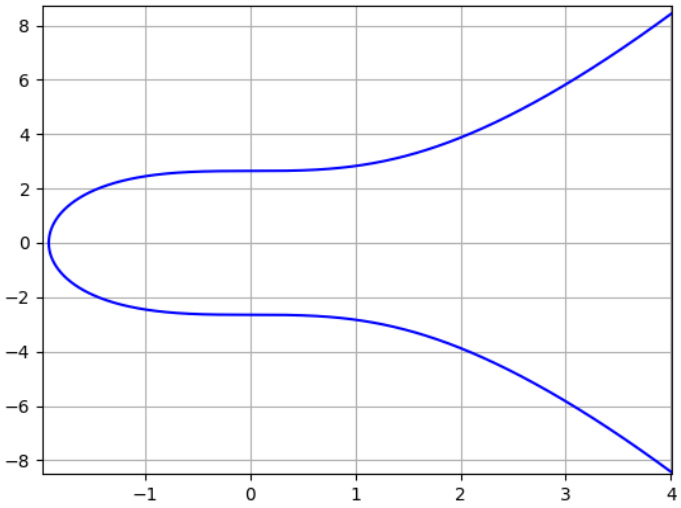

Fig 1. Gráfico de la curva elíptica secp256k1 *(Nakamoto 2010)*

2.4 Funciones resumen (hash)

Una función de resumen (hash) es un algoritmo matemático que, con una entrada A, nos da una salida B. B se conoce como digesto o huella digital (A. Menezes 1996).

Una función resumen puede ser cualquier algoritmo matemático, pero tiene que cumplir una serie de propiedades

1. Para una misma función hash, sea cual sea la longitud de la entrada A la salida B tiene que ser de un tamaño fijo.
2. Para cada A, B tiene que ser única.
3. Tiene que ser rápido y fácil de calcular.
4. No se puede volver a A desde B.
5. No puede presentar colisiones. Esto quiere decir que para dos A diferentes no se puede dar un mismo B.

Observando las condiciones 1 y 2 vemos que esto es imposible en la función MD5 que nos da como resultado un hash de 128 bits. Esto quiere decir que como máximo

hay solo 2^{128} textos diferentes, lo cual, es falso. Se hace por tanto muy importante que las colisiones sean mínimas y que encontrarlas sea muy difícil.

En el caso de funciones hash criptográficas se requiere de forma adicional que sean uniformes (para una A elegida aleatoriamente todos los valores hash son equiprobables) y con efecto avalancha (un cambio de un único bit en A supone una B completamente diferente).

Podemos distinguir dos grupos de funciones: las que tienen como objetivo mantener la integridad de los mensajes (detección de modificaciones) y las que tienen como objetivo verificar el origen del mensaje (autenticación).

2.4.1 Huella Digital

Un algoritmo de huella digital (función resumen) es un procedimiento que asigna un elemento de datos arbitrariamente grande (como un archivo informático) a una cadena de bits mucho más corta, su huella digital, que identifica de forma única los datos originales para todos los fines prácticos, como humanos las huellas dactilares identifican a las personas de manera única por ello de forma analógica podemos decir que la huella digital de un archivo lo identifica de manera única y cualquier variación, por pequeña que sea, en el documento original produce una huella digital distinta.

Este es uno de los elementos más importantes en el esquema de firma digital.

2.5 Firma Digital

La firma digital es el resultado de una transformación criptográfica de datos que, cuando se implementa correctamente, proporciona un mecanismo para verificar la autenticación de origen, la integridad de los datos y el no repudio del signatario (Cerén, Ley de Firma Electrónica 2015).

En el mundo físico, es común usar firmas autógrafas en mensajes escritos a mano o escritos a máquina. Se usan para unir el signatario al mensaje. Y es común confundir el término firma digital con el simple escaneo de la firma autógrafa y su inserción en un

documento como imagen, esto no es considerado firma digital, ya que para ello debe proveer los mecanismos mencionados en el primer párrafo.

Del mismo modo, una firma digital es una técnica que vincula a una persona/entidad con los datos digitales. Este enlace puede ser verificado independientemente por el receptor y por cualquier tercero.

La firma digital es un valor criptográfico que se calcula a partir de los datos y una llave secreta conocida solo por el firmante.

En el mundo real, el receptor del mensaje necesita la seguridad de que el mensaje pertenece al remitente y no debería poder rechazar el origen de ese mensaje. Este requisito es muy importante en las aplicaciones comerciales, ya que la probabilidad de una disputa sobre el intercambio de datos es muy alta.

A continuación, se muestra una imagen que describe el proceso de firma digital.

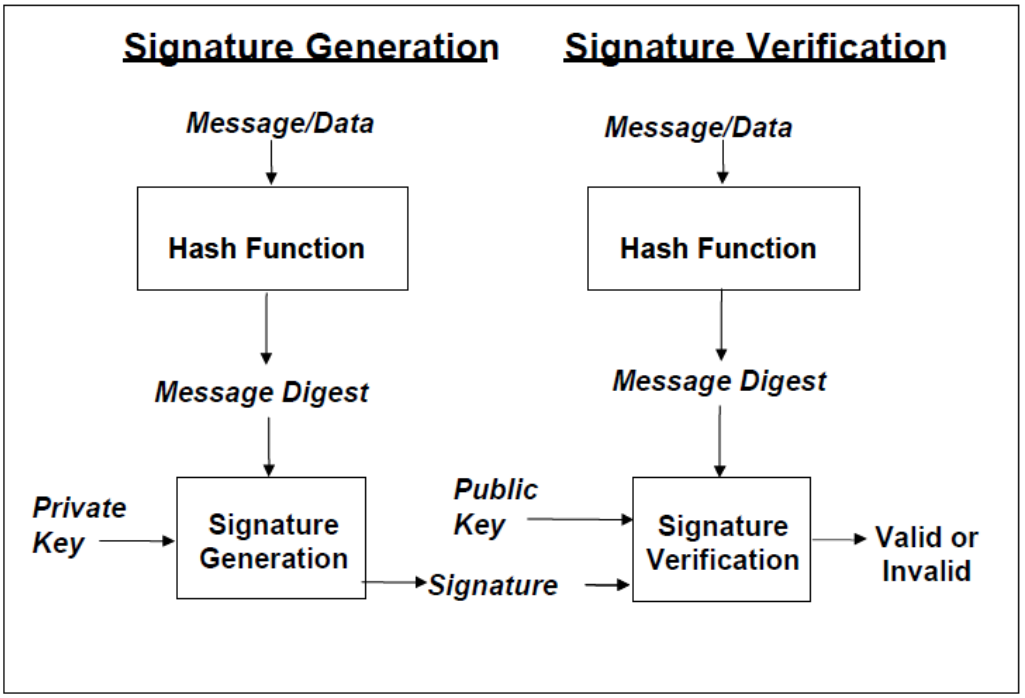

Fig 2. Proceso de firma digital *(NIST, Digital Signature Standard (DSS) 2013)*

Un algoritmo de firma digital incluye un proceso de generación de firma y un proceso de verificación de firma. Un signatario usa el proceso de generación para generar una firma digital en datos; un verificador utiliza el proceso de verificación para verificar la autenticidad de la firma (NIST, Digital Signature Standard (DSS) 2013). Cada firmante tiene una llave pública y privada y es el propietario de ese par de llaves. Como se muestra en la Figura 2, la llave privada se usa en el proceso de generación de firmas. El propietario del par de llaves es la única entidad autorizada para usar la llave privada para generar firmas digitales. Para evitar que otras entidades afirmen ser el propietario del par de llaves y usar la llave privada para generar firmas fraudulentas, la llave privada debe permanecer en secreto. Los algoritmos de firma digital aprobados por el NIST[4] (RSA y ECC) están diseñados para evitar que un adversario que no sabe la llave privada del firmante genere la misma firma que el firmante en un mensaje diferente. En otras palabras, las firmas están diseñadas para que no puedan falsificarse. Se pueden usar varios términos alternativos para referirse al propietario del par de llaves o signatario. Una entidad que tiene la intención de generar firmas digitales en el futuro se puede denominar el *signatario previsto*. Antes de la verificación de un mensaje firmado, el firmante se denomina el *signatario reclamante* hasta que se pueda obtener una garantía adecuada de la identidad real del firmante.

La llave pública se usa en el proceso de verificación de firma (vea la Figura 2). La llave pública no necesita mantenerse en secreto, pero se debe mantener su integridad. Cualquiera puede verificar un mensaje correctamente firmado utilizando la llave pública del signatario.

Tanto para la generación de firma como para los procesos de verificación, el mensaje (es decir, los datos firmados) se convierten en una representación de longitud fija del

[4] National Institute of Standard and Technology – Instituto Nacional de Estándares y Tecnología (NIST)

mensaje (huella digital) por medio de una función hash aprobada no “rota”. Tanto el mensaje original como la firma digital están disponibles para un verificador.

Un verificador requiere la seguridad de que la llave pública que se utilizará para verificar una firma pertenece a la entidad que afirma haber generado una firma digital (es decir, el signatario reclamante). Es decir, un verificador requiere la seguridad de que el firmante es el propietario real del par de llaves público/privado utilizado para generar y verificar una firma digital. Se debe realizar una unión de la llave pública y la identidad de su propietario para proporcionar esta garantía, esto se logra de manera segura con lo especificado en el estándar RFC 3820 de PKI, de acuerdo con dicho estándar una AC, de confianza ante terceros, debe emitir un certificado digital donde avala el vínculo de la llave pública con su propietario.

Un verificador también requiere la seguridad de que el propietario del par de llaves posee realmente la llave privada asociada con la llave pública, y que la llave pública es una llave matemáticamente correcta.

Al obtener estas garantías, el verificador tiene la seguridad de que, si la firma digital se puede verificar correctamente utilizando la llave pública, la firma digital es válida (es decir, el propietario del par de llaves realmente firmó el mensaje). La validación de la firma digital incluye la verificación (matemática) de la firma digital y la obtención de las garantías adecuadas.

2.6 Infraestructura de llave pública

Una infraestructura de llave pública (PKI) vincula las llaves públicas a las entidades, permite que otras entidades verifiquen los enlaces de las llaves públicas y proporciona los servicios necesarios para la gestión continua de las llaves en un sistema distribuido (NIST, Public Key Infrastructure 2001).

Es una combinación de hardware, software, y políticas y procedimientos de seguridad, que permiten la ejecución con garantías de operaciones criptográficas, como el cifrado, la firma digital, y el no repudio de transacciones electrónicas (IETF, RFC 3820 2018).

2.6.1 Componentes de una PKI

PKI proporciona seguridad de llave pública. Proporciona la identificación de llaves públicas y su distribución. Una anatomía de PKI comprende los siguientes componentes: Autoridad Certificadora, Autoridad de Registro, Repositorio, Archivo y Usuarios (NIST, Public Key Infrastructure 2001).

Autoridad Certificadora (AC)

La autoridad de certificación, o AC, es el componente básico de la PKI. La AC es una colección de hardware, software y las personas que lo operan. La AC se conoce por dos atributos: su nombre y su llave pública. La AC realiza cuatro funciones básicas de PKI:

1. Emite certificados (es decir, los crea y los firma);
2. Mantiene información sobre el estado del certificado y emite CRL[5];
3. Publica sus certificados y CRL actuales (por ejemplo, no vencidos), para que los usuarios puedan obtener la información que necesitan para implementar servicios de seguridad;
4. Y mantiene archivos de información de estado sobre los certificados vencidos que emitió. Estos requisitos pueden ser difíciles de satisfacer simultáneamente. Para cumplir con estos requisitos, la AC puede delegar ciertas funciones a los otros componentes de la infraestructura.

[5] Certificate Revocation List

Una AC puede emitir certificados a los usuarios, a otras AC o ambas. Cuando una AC emite un certificado, afirma que el sujeto (la entidad nombrada en el certificado) tiene la clave privada que corresponde a la clave pública contenida en el certificado. Si la AC incluye información adicional en el certificado, la AC está afirmando que la información corresponde al tema también. Esta información adicional puede ser información de contacto (por ejemplo, una dirección de correo electrónico) o información de política (por ejemplo, los tipos de aplicaciones que se pueden realizar con esta clave pública). Cuando el sujeto del certificado es otra AC, el emisor está afirmando que los certificados emitidos por la otra AC son confiables.

La AC inserta su nombre en cada certificado (y CRL) que genera, y los firma con su clave privada. Una vez que los usuarios establecen que confían en una AC (directamente o mediante una ruta de certificación) pueden confiar en los certificados emitidos por esa AC. Los usuarios pueden identificar fácilmente los certificados emitidos por esa AC comparando su nombre. Para asegurarse de que el certificado sea genuino, verifican la firma utilizando la clave pública de la AC. Como resultado, es importante que la AC brinde protección adecuada para su propia clave privada

Autoridad de Registro (AR)

Un AR (en inglés RA[6]) está diseñado para verificar el contenido del certificado para la AC. El contenido del certificado puede reflejar la información presentada por la entidad que solicita el certificado, como una licencia de conducir o un recibo de pago reciente. También pueden reflejar información proporcionada por un tercero. Por ejemplo, el límite de crédito asignado a una tarjeta de crédito refleja la información obtenida de las agencias de crédito. Un certificado puede reflejar datos del departamento de Recursos Humanos de la compañía, o una carta de un funcionario designado de la compañía. Por ejemplo, el certificado de Bob podría indicar que tiene autoridad de firma

[6] Registration Authority

para contratos pequeños. La RARA agrega estas entradas y proporciona esta información a la AC.

Al igual que la AC, la AR es una colección de hardware, software y la persona o personas que lo operan. A diferencia de una AC, una AR a menudo será operada por una sola persona. Cada AC mantendrá una lista de AR acreditadas; esa es una lista de ARs que se consideran confiables. La AC conoce un RA por un nombre y una clave pública. Al verificar la firma de la AR en un mensaje, la AC puede estar segura de que una AR acreditada proporcionó la información, y se puede confiar en ella. Como resultado, es importante que el AR proporcione protección adecuada para su propia clave privada

Repositorio PKI

Un repositorio es una base de datos de certificados digitales activos para un sistema de AC. El negocio principal del repositorio es proporcionar datos que permitan a los usuarios confirmar el estado de los certificados digitales para individuos y empresas que reciben mensajes firmados digitalmente, y además gestionar la actualización de certificados. Estos destinatarios de mensajes se denominan partes confiables. Las AC publican certificados y CRL en repositorios.

Archivo (historial)

Un archivo acepta la responsabilidad del almacenamiento a largo plazo de la información histórica en nombre de la AC. Un archivo afirma que la información era buena en el momento en que se recibió y que no se ha modificado mientras estaba en el archivo. La información proporcionada por la AC al archivo debe ser suficiente para determinar si la AC emitió un certificado tal como se especifica en el certificado y si es válido en ese momento. El archivo protege esa información a través de mecanismos técnicos y procedimientos apropiados mientras está bajo su cuidado. Si surge una disputa en una fecha posterior, la información se puede usar para verificar que la llave

privada asociada con el certificado se usó para firmar un documento. Esto permite la verificación de firmas en documentos antiguos (como testamentos) en una fecha posterior.

Usuarios de la PKI

Los usuarios de la PKI son organizaciones o individuos que usan la PKI, pero no emiten certificados. Confían en los otros componentes de la PKI para obtener certificados y para verificar los certificados de otras entidades con las que hacen negocios. Las entidades finales incluyen la parte que confía, que se basa en el certificado para conocer, con certeza, la llave pública de otra entidad; y el titular del certificado, que recibe un certificado y puede firmar documentos digitales. Se debe tener en cuenta que un individuo u organización puede ser una parte confiable y un titular de certificado para diversas aplicaciones.

2.6.2 Estructuras de datos de una PKI

2.6.2.1 Certificado digital

Un certificado digital o como se le conoce en criptografía: certificado de llave pública es un documento electrónico usado para probar la propiedad de una llave pública, este incluye de acuerdo con el estándar X.509 del RFC 3280 (IETF, RFC 3280 2018) ciertas partes, entre ellas las más importantes: llave publica e identidad del propietario, firma digital de la AC que emite el certificado y periodo de validez. La AC es la entidad de confianza responsable de emitir y revocar certificados firmándolos con su llave privada, normalmente estas tienen un certificado raíz auto firmado o firmado por la autoridad raíz, en caso que haya una jerarquía de confianza, que la identifica como autoridad de certificación.

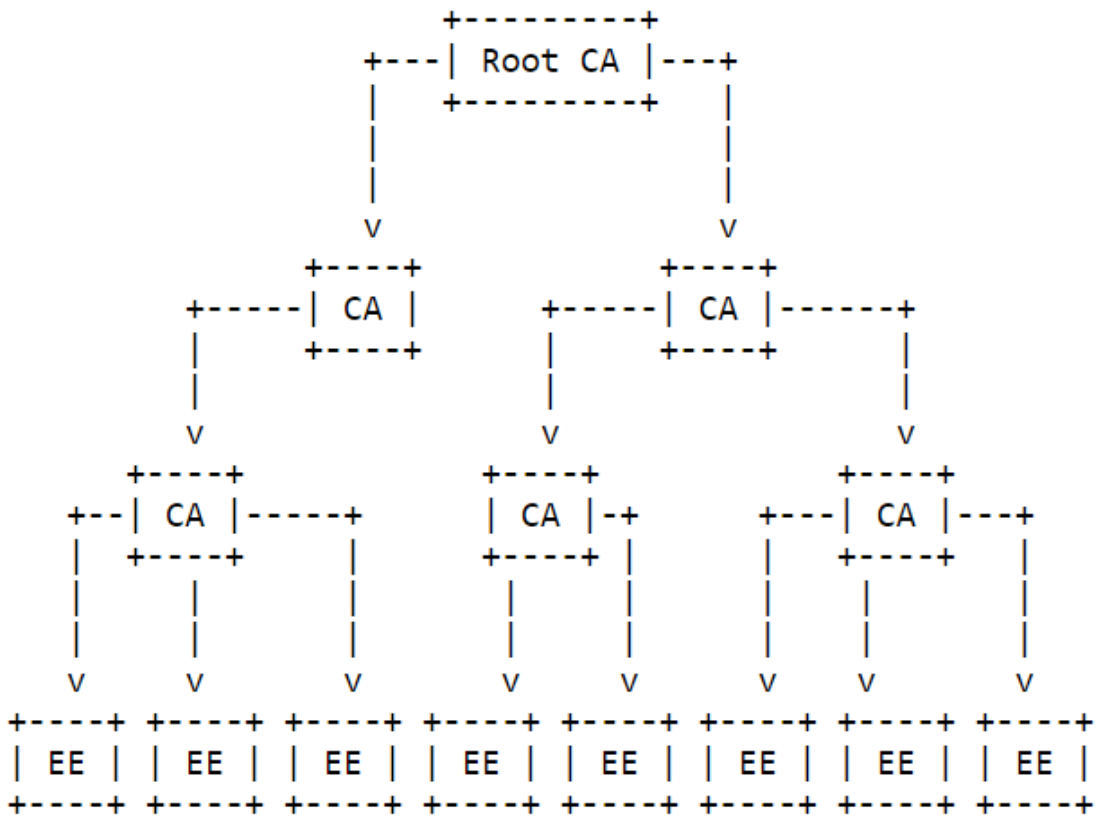

Fig 3. Ejemplo de jerarquía de PKI, RFC 4158 *(IETF, RFC 4158 2018)*

La construcción de la ruta de certificación en una PKI jerárquica es un proceso directo que simplemente requiere que la parte que confía recupere sucesivamente certificados del emisor hasta un certificado que fue emitido por el ancla de confianza (la "AC raíz" en la Figura 3). El estándar RFC 4158 explica otras variaciones de la estructura (IETF, RFC 4158 2018).

El certificado X.509 V3 (Figura 3) emitido a usuarios finales debe contar con los siguientes elementos de acuerdo con el estándar RFC 3280 (IETF, RFC 3820 2018)

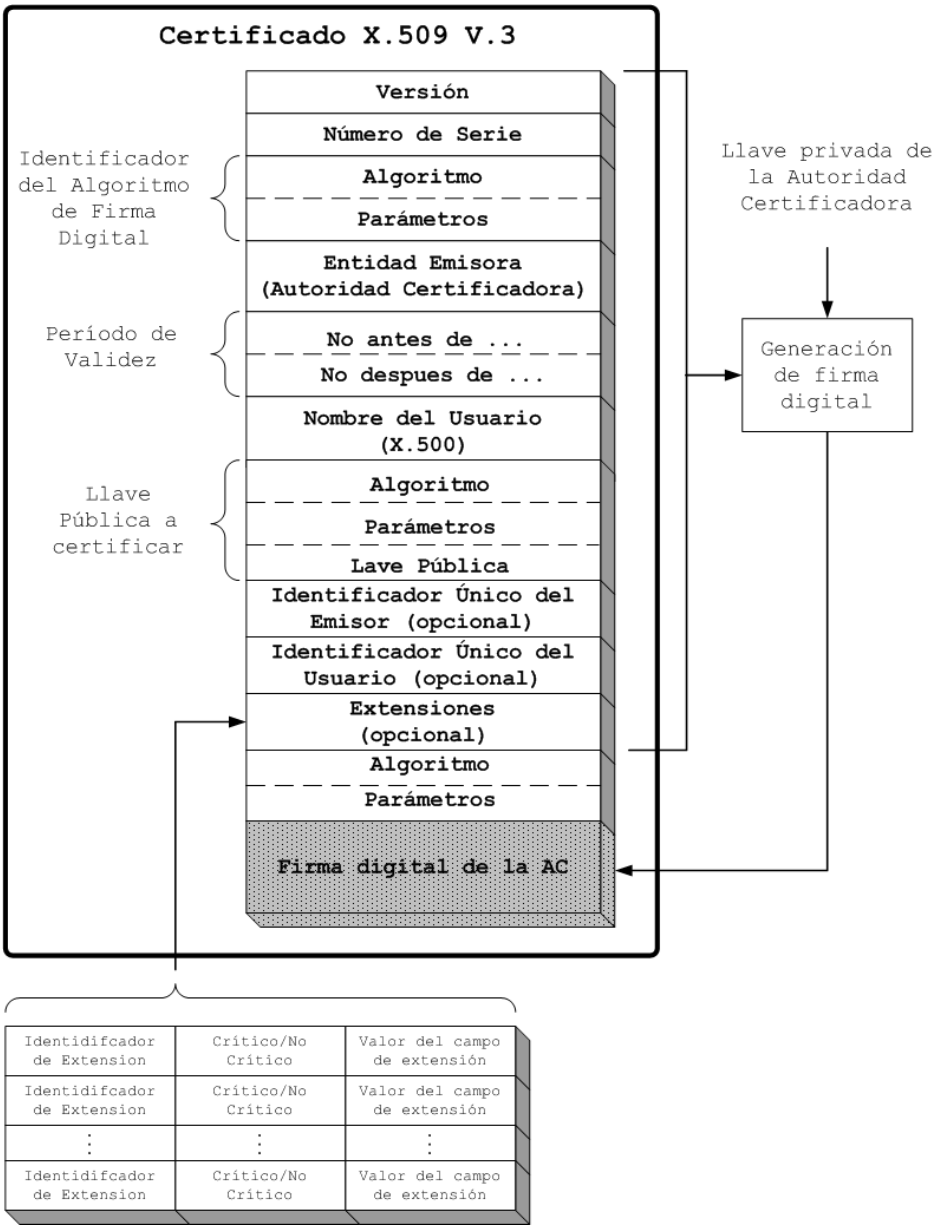

Fig 4. Formato X.509 V3

2.6.2.2 Lista de certificados revocados (CRL)

Los certificados contienen una fecha de vencimiento. Desafortunadamente, los datos en un certificado pueden volverse poco confiables antes de que llegue la fecha de

vencimiento. Los emisores de certificados necesitan un mecanismo para proporcionar una actualización de estado para los certificados que han emitido. Un mecanismo es la lista de revocación de certificación X.509 (CRL).

Las CRL son el análogo PKI de la lista activa de tarjetas de crédito que los empleados de la tienda revisan antes de aceptar grandes transacciones con tarjeta de crédito. La CRL está protegida por una firma digital del emisor de la CRL. Si se puede verificar la firma, los usuarios de CRL saben que el contenido no ha sido alterado desde que se generó la firma. Las CRL contienen un conjunto de campos comunes y también pueden incluir un conjunto opcional de extensiones.

2.7 Usurpación de identidad

La usurpación de la identidad, en firma digital, debe ser entendida como la suplantación del firmante por un impostor para obtener un beneficio injusto, y recibe cada vez más atención en materia de fraudes cometidos con la ayuda de las tecnologías de la información, la premisa para usurpar la identidad de un firmante es tener acceso a su llave privada.

Las malas decisiones técnicas del criptosistema, así como el mal manejo de parte de los usuarios de su llave privada, pues normalmente el eslabón más débil en la seguridad de la información es la llamada “capa 8”, darán como resultado que la llave privada del firmante se vea comprometida y dar lugar a ataques de usurpación de identidad. Según el principio de Kerckhoffs el oponente conoce el algoritmo criptográfico utilizado, Por lo tanto, la seguridad del sistema debe estar basada en:

- La calidad (fortaleza) del algoritmo
- El tamaño del espacio de la llave (tamaño en bits de la llave)

Mitigando lo más posible los errores de la “capa 8” y seleccionando algoritmos que provean las garantías de seguridad mínimas con respecto a la calidad y al tamaño de las llaves, y que las funciones resumen para el cálculo de las huellas digitales no entes “rotas” son decisiones claves. Debe mencionarse que la vulnerabilidad de la criptografía

empleada dependerá en gran medida de estas decisiones de seguridad donde los algoritmos como SHA-1 para cálculo de digestos o el RSA de 1024 bits son altamente frágiles a ataques de fuerza bruta, siendo esta la principal arma de un hacker.

Ataque de fuerza bruta. Consiste en intentar todas las posibles llaves hasta dar con la correcta. Este tipo de ataque es factible según el número de posibles llaves lo cual suele venir dado por la longitud (tamaño) de la llave.

Otras formas de vulnerar el criptosistema pueden venir de lo explicado a continuación

Búsqueda de alguna debilidad o fallo. Esta debilidad suele provenir de:

- Obsolescencia del algoritmo. Para ello es importante tener en cuenta los nuevos descubrimientos que pueden dejar un algoritmo obsoleto. Por ejemplo, un avance en factorización de números primos podría dejar obsoleto el algoritmo RSA o un avance en el cálculo de logaritmos discretos lo haría con ECC. Otro ejemplo sería el uso de ordenadores cuánticos que acelerarían mucho lo velocidad de computación
- Un fallo en la implementación: Un ejemplo típico es el uso de generador de números aleatorios con debilidades como el ataque que realizaron Ian Goldberg y David Wagner al SSL de Netscape
- Aplicación de técnicas de criptoanálisis.

2.8 Ley de Firma Electrónica de El Salvador

A continuación, y de acuerdo con (Cerén, Ley de Firma Electrónica 2015) se establecen aquellas partes de la ley referentes o relacionados a la gestión de certificados por parte de una AC.

2.8.1 Objeto de la ley

La ley tiene por objeto:

a) Equiparar la firma electrónica simple y firma electrónica certificada con la firma autógrafa;
b) Otorgar y reconocer eficacia y valor jurídico a la firma electrónica certificada, a los mensajes de datos y a toda información en formato electrónico que se encuentren suscritos con una firma electrónica certificada, independientemente de su soporte material;
c) Regular y fiscalizar lo relativo a los proveedores de servicios de certificación electrónica, certificados electrónicos y proveedores de servicios de almacenamiento de documentos electrónicos.

2.8.2 Autoridad Competente

La Autoridad de Control y Vigilancia

Art. 35.- Créase la Unidad de Firma Electrónica, como parte del Ministerio de Economía, el que en el texto de esta Ley podrá abreviarse MINEC. El ministro nombrará al funcionario que estará a cargo de esta Unidad, quien deberá reunir los requisitos que para tal efecto se establezcan en el reglamento de esta Ley.

De la Unidad de Firma Electrónica

Art. 36.- La Unidad de Firma Electrónica será la autoridad registradora y acreditadora raíz, y la competente para la acreditación, control y vigilancia de los proveedores de los servicios de certificación electrónica y de almacenamiento de documentos electrónicos, de conformidad con esta Ley, su reglamento y las normas y reglamentos técnicos.

Competencias de la Unidad de Firma Electrónica

Art. 37.- La Unidad de Firma Electrónica tendrá las siguientes competencias:

a) Elaborar las normas y los reglamentos técnicos que sean necesarios para la implementación de la presente Ley, en coordinación con el Organismo Salvadoreño de Reglamentación Técnica (OSARTEC) y el Organismo Salvadoreño de Normalización (OSN);

b) Otorgar, registrar o revocar la acreditación a los proveedores de servicios de certificación y a los prestadores de servicios de almacenamiento de documentos electrónicos, una vez cumplidas las formalidades y requisitos de esta Ley, su reglamento y demás normas y reglamentos técnicos aplicables;

c) Validar los certificados electrónicos emitidos a favor de los proveedores de servicios de certificación y de almacenamiento de documentos electrónicos;

2.8.3 Acreditación y prestación de los servicios de certificación

El art 43 inciso a) de los requisitos generales para una AC menciona lo siguiente

a) Contar con suficiente capacidad técnica para garantizar la seguridad, la calidad y la fiabilidad de los certificados emitidos, de conformidad a los requerimientos contenidos en las normas técnicas;

Obligaciones de los Proveedores

Art. 48.- Los proveedores de servicios de certificación tendrán las siguientes obligaciones:

a) Adoptar las medidas necesarias para determinar la exactitud de los certificados electrónicos que proporcionen, la identidad y la calidad del signatario;

b) Garantizar la validez, vigencia, legalidad y seguridad del certificado

electrónico que proporcione;

c) Garantizar la adopción de las medidas necesarias para evitar la falsificación de certificados electrónicos y de las firmas electrónicas certificadas que proporcionen;

d) Verificar la información suministrada por el signatario;

e) Crear y mantener un archivo actualizado de los certificados emitidos en medios electrónicos, para su consulta por plazo indefinido;

f) Garantizar a los usuarios los mecanismos necesarios para el ejercicio de los derechos de acceso, rectificación, cancelación y oposición;

g) Sin perjuicio de otras obligaciones establecidas en la Ley de Protección al Consumidor, deberá informar a los interesados de sus servicios de certificación, utilizando un lenguaje comprensible, a través de su sitio de internet y a través de cualquier otra forma de acceso público, los términos precisos y condiciones para el uso del certificado electrónico y, en particular, de cualquier limitación sobre su responsabilidad, así como de los procedimientos especiales existentes para resolver cualquier controversia;

h) Garantizar la autenticidad, integridad y confidencialidad de la información, y documentos relacionados con los servicios que proporcione. A tales efectos, deberán mantener un sistema de seguridad informática y respaldos confiables y seguros de dicha información, de conformidad a lo establecido en la presente Ley, su reglamento, y normas y reglamentos técnicos;

i) Efectuar las notificaciones para informar a los signatarios y personas interesadas y las publicaciones necesarias, acerca del vencimiento, revocación, suspensión o cancelación de los certificados electrónicos que proporcione, así como de cualquier otro aspecto de relevancia para el público en general, en relación con los mismos;

j) Dar aviso a la Fiscalía General de la República, cuando en el desarrollo de sus actividades tenga indicios de la comisión de un delito;

k) Renovar anualmente la fianza establecida en el Art 43, literal d) de esta Ley, previo a su vencimiento; y,

l) Cumplir con las demás obligaciones establecidas en esta Ley, su reglamento, y demás normas y reglamentos técnicos.

El incumplimiento de cualquiera de los requisitos anteriores dará lugar a las sanciones establecidas en la presente Ley.

2.8.4 De los certificados electrónicos

Garantía de la Autoría de la Firma Electrónica Certificada

Art. 57.- El certificado electrónico garantiza la autoría de la firma electrónica certificada, así como la autenticidad, integridad, confidencialidad y no repudiación del documento electrónico.

Contenido del Certificado Electrónico

Art. 58.- El certificado electrónico deberá contener al menos, la siguiente información:

a) Identificación del titular del certificado electrónico, indicando su domicilio y dirección electrónica;

b) Identificación del proveedor de servicios de certificación que proporciona el certificado electrónico, indicando su domicilio y dirección electrónica;

c) Fecha de la acreditación y caducidad asignada al proveedor de servicios de certificación por la Unidad de Firma Electrónica;

d) Fecha de emisión y expiración del certificado;

e) Número de serie o de identificación del certificado;

f) La firma electrónica certificada del prestador de servicios de certificación

que emitió el certificado;

g) Datos de verificación de la firma, los cuales deben corresponder a la información de su creación y que están bajo el control del firmante;

h) Cualquier información relativa a las limitaciones de uso, vigencia y responsabilidad a las que esté sometido el certificado electrónico;

i) Indicación de la ruta de certificación; y,

Si el certificado ha sido emitido por una persona que ha actuado en representación de una persona natural o jurídica; en tal caso, el certificado deberá incluir una indicación del documento legal, público, o privado autenticado, que acredite de forma fehaciente las facultades del firmante para actuar en nombre de la persona física o jurídica a la que representa.

La falta de alguno de estos requisitos invalidará el certificado.

Vigencia del Certificado Electrónico

Art. 59.- El proveedor de servicios de certificación y el signatario, de mutuo acuerdo, determinarán el plazo de vigencia del certificado electrónico.

Cancelación del Certificado Electrónico

Art. 60.- El certificado electrónico de la firma electrónica certificada puede ser cancelado por resolución judicial, de conformidad con el ordenamiento legal. Asimismo, puede ser cancelado por resolución razonada emitida por el Ministerio de Economía a través de la Unidad de Firma Electrónica, en cualquiera de los supuestos siguientes:

a) Que se compruebe que alguno de los datos del certificado electrónico proporcionado por el proveedor de servicios de certificación, es falso;

b) Que sea violentado el sistema de seguridad del proveedor de servicios de certificación, y que afecte la integridad y confiabilidad del certificado;

c) Que el signatario dé aviso al proveedor, de la destrucción o extravío del

certificado electrónico. En tal caso, el proveedor de servicios de certificación procederá inmediatamente a la cancelación del certificado; y,

d) Por fallecimiento, o muerte presunta, previa resolución judicial. Para el caso de persona jurídica en el cese de sus actividades, por disolución

Procedimiento para la Cancelación de un Certificado Electrónico

Art. 61.- El Ministerio de Economía por medio de la Unidad de Firma Electrónica, previa denuncia del interesado o de oficio, ordenará audiencia por tres días hábiles al proveedor de servicios de certificación, y con lo que conteste o no, se abrirá a pruebas por ocho días hábiles, a fin de demostrar cualquiera de las situaciones consideradas en el artículo anterior; finalizado el término probatorio, la Unidad de Firma electrónica emitirá resolución razonada, en un plazo no mayor de diez días hábiles, para que determine si es procedente la cancelación del certificado que ampara la firma electrónica. Esta resolución admitirá recurso de revisión y será resuelto en el plazo de quince días hábiles, con la vista de autos.

2.8.5 Derechos y obligaciones de los Usuarios

Art. 62.- Además de los derechos reconocidos por la Ley de Protección al Consumidor y cualquier otra normativa aplicable, los usuarios o titulares de los servicios regulados en esta Ley, tendrán los siguientes derechos, según sea el caso:

a) A ser informados por los proveedores de servicios de certificación, de las características generales de los procedimientos de creación y de verificación de firma electrónica certificada, así como de las reglas sobre prácticas de certificación, y los demás que éstos se comprometan a seguir en la prestación de los servicios, lo que deberá realizarse de forma previa a la adquisición del servicio;

b) A la confidencialidad en la información, en los supuestos en que los

proveedores de servicios de certificación, y de almacenamiento de documentos electrónicos decidan cesaren sus actividades;

c) A ser informados, antes de la emisión de un certificado, de los precios de los servicios, incluyendo cargos adicionales y formas de pago, en su caso; de las condiciones precisas para la utilización de los servicios y de sus limitaciones de uso, y de los procedimientos de reclamación y de resolución de litigios;

d) A que el prestador de servicios le proporcione la información sobre su domicilio en el país;

e) A ser informado, al menos con noventa días de anticipación, por los prestadores de servicios de certificación, y almacenamiento de documentos electrónicos, para los efectos del cierre de actividades;

f) A traspasar sus datos a otro prestador de servicios de certificación y de almacenamiento de documentos electrónicos, si así lo solicitan;

g) A que el proveedor no proporcione u otorgue servicios no solicitados; deteriorar la calidad de los servicios contratados en calidad de inferioridad; o servicios adicionales cobrados no pactados; a no recibir publicidad comercial de ningún tipo por intermedio del proveedor, salvo autorización expresa del usuario en todos los casos señalados; y,

h) La cancelación del certificado por petición del usuario o su representante legal.

La violación a los derechos previstos en este artículo constituye infracción grave en los términos señalados en la Ley de Protección al Consumidor, y será sancionada como tal.

La determinación de la infracción y la imposición de la sanción correspondiente será competencia del Tribunal Sancionador de la Defensoría del Consumidor, y de acuerdo con el procedimiento previsto en la Ley de Protección al Consumidor, en lo que fuere aplicable.

Obligaciones de los Usuarios

Art. 63.- Los usuarios o titulares de firmas electrónicas certificadas, y de almacenamiento de documentos electrónicos, quedarán obligados en el momento de proporcionar los datos de su identidad personal u otras circunstancias objeto de certificación, a:

a) Brindar declaraciones veraces y completas;

b) Custodiar adecuadamente los mecanismos de seguridad del funcionamiento del sistema de certificación que le proporcione el prestador y actualizar sus datos en la medida que éstos vayan cambiando, so pena de responder por la indemnización de daños y perjuicios derivada del incumplimiento de estas obligaciones; y,

c) Solicitar oportunamente la suspensión o revocación del certificado, ante cualquier circunstancia que pueda haber comprometido la privacidad de los datos de creación de firma electrónica certificada.

CAPÍTULO III

METODOLOGÍA DE LA INVESTIGACIÓN

Capitulo III. Metodología de la investigación

3.1 Tipo de investigación

La investigación a realizar para el presente trabajo es de tipo descriptiva. Este tipo de estudio busca descubrir el estado de las variables que intervienen dentro de esta, es decir se debe medir, recolectar o evaluar datos sobre diversos conceptos (variables), aspectos, dimensiones o componentes del proyecto a desarrollar.

3.2 Unidades de Análisis

- Certificado digital X.509 v3.

  El formato a estudiar es el más común para los certificados de clave pública y está definido por X.509. Debido a que X.509 es muy general, el formato está aún más restringido por los perfiles definidos para ciertos casos de uso, como la Infraestructura de clave pública (X.509) como se define en RFC 5280. En sistemas de firma electrónica, el sujeto de un certificado suele ser una persona u organización. La versión X.509 v3 permite utilizar campos opcionales (nombres alternativos, usos permitidos para la clave, ubicación de la CRL y de la CA.

- Algoritmo de firma digital de curva elíptica (ECDSA)[7].

  El algoritmo estudiado es una modificación del algoritmo DSA que emplea operaciones sobre puntos de curvas elípticas en lugar de las exponenciaciones que usa DSA. La principal ventaja de este esquema es que requiere números de tamaños menores para brindar la misma seguridad que DSA o RSA.

[7] Elliptic Curve Digital Signature Algorithm

- Ley de Firma Electrónica de El Salvador.

  Los elementos a estudiar en La Ley son todos aquellos relacionados con los requerimientos de los certificados y los procesos que debe realizar una AC en cuanto al ciclo de vida de estos certificados que son emitidos a usuarios finales.

- Estándar NIST PKI.

  Las secciones a considerar son las relacionadas al funcionamiento de los componentes y estructuras de datos involucradas en una PKI. Se busca describir el flujo de procesos seguido.

3.3 Variables y su medición

3.3.1 Definición de variables

Variable	Definición conceptual	Definición operacional
Requerimientos	Son declaraciones, en lenguaje natural y/o en diagramas, de los servicios que se espera que el software proporcione y de las restricciones bajo las cuales debe de funcionar.	Son las especificaciones acerca de la delimitación y funcionalidad que ofrecerá el software de gestión de certificados para una AC considerando la Ley de Firma Electrónica de El Salvador.
Especificaciones técnicas	Documento o descripción de las normas, exigencias y procedimientos técnicos aplicados que debe reunir	Son los requisitos técnicos que requiere el algoritmo ECDSA para garantizar la seguridad de las operaciones criptográficas.

	un producto, proceso, servicio o sistema.	
Tamaños de llave	El tamaño o longitud de la clave es el número de bits de una clave que utiliza un algoritmo criptográfico (por ejemplo, un cifrador).	Tamaños en bits de las llaves seguras empleadas en el algoritmo ECDSA.
Procesos de una AC	Los procesos involucrados en la gestión de los certificados firmados. Esto incluye las tareas de creación, revocación y renovación de certificados por parte de la AC.	La gestión del ciclo de vida de los certificados desde su solicitud hasta su revocación o renovación por caducidad.

Tabla 1. Cuadro de definición de variables

3.3.1 Indicadores y su medición

Este cuadro presenta nuevamente las variables en estudio con sus respectivos indicadores, los instrumentos de medición, la técnica a aplicar y a que unidad de análisis pertenecen.

Unidad de análisis	**Variables**	**Indicadores**	**Técnicas a aplicar**	**Instrumento de medición u observación**
Certificado Digital X.509 V3	Requerimientos	Listado de requerimientos	Observación	Hoja de anotaciones (Anexo 2)
ECDSA	Especificaciones técnicas	Listado de especificaciones técnicas	Análisis documental	Hoja de anotaciones (Anexo 2)

	Tamaño de llaves	Tamaños de llave de seguridad mínima en bits	Análisis documental	Cuadro comparativo (Anexo 3)
Ley de firma electrónica de El Salvador + Estándar NIST PKI	Requerimientos	Listado de requerimientos	Observación	Hoja de anotaciones (Anexo 4)
	Procesos de una AC	Diagrama de procesos	Observación	Hoja de anotaciones (Anexo 5)

Tabla 2. Cuadro de variables y su medición

3.3.2 Instrumentos de medición

- Hoja de anotaciones

  Documento donde se detallan las observaciones, notas, explicaciones u otros tipos de comentarios que deben tomarse en cuenta para describir las variables de la investigación.

- Cuadro Comparativo

  Estrategia que permite identificar las semejanzas y diferencias de dos o más objetos o hechos. Una cuestión importante es que, luego de hacer el cuadro comparativo se enuncia la conclusión a la que se llegó.

3.4 Procesamiento y análisis de la información

El proceso a seguir se describe a continuación.

Recolección de información

Primero se seleccionarán los documentos a analizar según las unidades de análisis definidas, luego se procederá a estudiar los apartados relacionas con las

variables y su medición llenando los instrumentos de medición respectivos hasta obtener los resultados por cada variable.
Las técnicas empleadas serán análisis documental y observación

Procesamiento de la información

La información recopilada será analizada de acuerdo a su relevancia para el proyecto. De esta manera incluir únicamente aquellos resultados que son pertinentes y generan un aporte para la realización del proyecto.

Análisis de la información

De acuerdo a cada variable y su indicador se generan las conclusiones o resultados finales a ser tomados en cuenta para la realización del proyecto. Los resultados se muestran por unidad de análisis y sus variables en el capítulo IV de este documento.

3.5 Cronograma de actividades

A continuación, se describe el cronograma de actividades para la realización del proyecto.

Id.	Nombre de tarea	Comienzo	Fin	Duración
1	Elaboración de perfil del proyecto	2/2/2020	10/2/2020	9d
2	Revisión de bibliografía	18/2/2020	25/2/2020	8d
3	Capítulo I. Planteamiento del problema	26/2/2020	27/2/2020	2d
4	Capítulo II. Estado del arte	28/2/2020	6/3/2020	8d
5	Capítulo III. Metodología de la investigación	7/3/2020	8/3/2020	2d
6	Enviar anteproyecto (primeros 3 capítulos)	9/3/2018	9/3/2018	1d
7	Capítulo IV. Análisis y discusión de resultados	1/4/2020	15/5/2020	45d
8	Capitulo V. Propuesta Técnica	1/6/2020	4/11/2020	157d
9	Capítulo VI. Conclusiones y recomendaciones	5/11/2020	6/11/2020	2d
10	Elaboración de artículo	7/11/2020	20/11/2020	14d
11	Entrega de proyecto final	22/11/2020	22/11/2020	1d
12	Entrega de artículo	22/11/2020	22/11/2020	1d

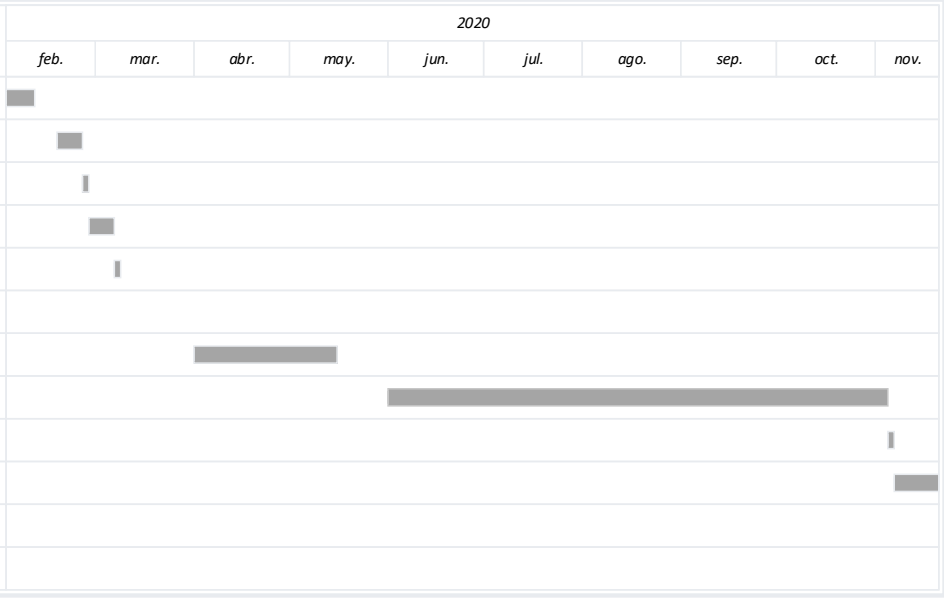

Tabla 3. Cronograma de actividades del proyecto.

CAPÍTULO IV

ANÁLISIS Y DISCUSIÓN DE RESULTADOS

Capitulo IV. Análisis y discusión de resultados

A continuación, se presenta el desglose de la información recopilada y analizada para medir las variables planteadas en la investigación

4.1 Variable: Requerimientos certificados digitales

Para establecer los requerimientos de un certificado digital se tomaron en cuenta dos fuentes: El Estándar RFC 5280 (Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile) y La ley de firma electrónica del El Salvador.

Ley de firma electrónica	**RFC 5280**
1. Identificación del titular del certificado electrónico, indicando su domicilio y dirección electrónica	Sujeto, (sujeto titular) expresado en notación DN (Distinguished Name), compuesto por CN (Common Name), OU (Organizational Unit), O (Organization), C (Country), STREET (Domicilio) e E (Email). Además, la ciudad y el estado. El sujeto puede ser una persona, un servidor o un servicio.
2. Identificación del proveedor de servicios de certificación que proporciona el certificado electrónico, indicando su domicilio y dirección electrónica;	Emisor
3. Fecha de la acreditación y caducidad asignada al proveedor de servicios de certificación por la Unidad de Firma Electrónica	No Aplica
4. Fecha de emisión y expiración del certificado	Validez

5. Número de serie o de identificación del certificado	Número de serie del certificado (debe ser único para cada certificado emitido por una misma CA)
6. La firma electrónica certificada del prestador de servicios de certificación que emitió el certificado	• Firma digital del certificado

7. Datos de verificación de la firma, los cuales deben corresponder a la información de su creación y que están bajo el control del firmante	• Información de clave pública del sujeto
	  ○ Algoritmo de clave pública
	  ○ Clave pública del sujeto

8. Cualquier información relativa a las limitaciones de uso, vigencia y responsabilidad a las que esté sometido el certificado electrónico	• Versión • ID del algoritmo utilizado por el CA para firmar • Algoritmo usado para firmar el certificado
9. Indicación de la ruta de certificación; y	Certificado de CA raíz
10. Si el certificado ha sido emitido por una persona que ha actuado en representación de una persona natural o jurídica; en tal caso, el certificado deberá incluir una indicación del documento legal, público, o privado autenticado, que acredite de forma fehaciente las facultades del firmante para actuar en nombre de la persona física o jurídica a la que representa	No Aplica

Tabla 4. Requerimientos certificado digital

La versión X.509.v3 también permite utilizar campos opcionales (nombres alternativos, usos permitidos para la clave, ubicación de la CRL y de la CA, etc.). Lo

anterior con el fin de incluir los campos que la ley menciona pero que el estándar básico no tiene.

Por tabla anterior podemos concluir los campos que debe incluir el certificado y que son indispensables según la ley y sus equivalentes en el estándar X.509 V3 así como los campos que deben ser agregados.

4.2 Variable: Tamaño de llaves para la firma digital

La fortaleza de un sistema criptográfico descansa en la calidad del algoritmo y el tamaño de las llaves. Para la elección del tamaño de llaves se considera el contenido en (NIST, Recommendation for key management 2016) del cual se obtiene el siguiente cuadro comparativo:

Security Strength	Symmetric key Algorithms	FFC (e.g., DSA, D-H)	IFC (e.g., RSA)	ECC (e.g., ECDSA)
<=80	2TDEA	L=1024 N= 160	K= 1024	f=160-223
112	3TDEA	L=2048 N= 224	K=2048	f=224-255
128	AES-128	L=3072 N= 256	K=3072	f=256-383
192	AES-192	L=7680 N= 384	K=7680	f=384-511
256	AES-256	L=15360 N= 512	K=15360	f=512+

Tabla 5. Comparación de fortaleza de llaves

Tener en cuenta que las fortalezas de las claves de 192 y 256 bits identificadas para los algoritmos FFC e IFC (sombreadas en amarillo) no se incluyen actualmente en los estándares del NIST por razones de interoperabilidad y eficiencia.

Además, tener en cuenta que las combinaciones de algoritmo / tamaño de clave que se han estimado en una potencia de seguridad máxima de menos de 112

bits (sombreada en naranja arriba) ya no están aprobadas para aplicar la protección criptográfica (por ejemplo, cifrar datos o generar una firma digital).

Con el análisis de la tabla 2 se concluye cuáles son tamaños de llave seguros para la aplicación los cuales están en la zona sombreada verde. En la tabla 3 se plantean algoritmos y tamaños de llaves con base en la tabla 3.

4.3 Variable: Especificaciones técnicas firma digital.

Posterior al análisis del algoritmo en (NIST, Digital Signature Standard (DSS) 2013) se obtiene el siguiente resultado:

Descripción	**Especificación técnica**
Algoritmo de firma	Elliptic Curve Digital Signature Algorithm (ECDSA)
Curva utilizada	Secp256r1
Función Hash para Huella Digital	SHA3-256withECDSA
Cifrador por bloques	AES
Tamaño de llave para AES	256 bits (32 bytes)
Función Hash para cifrado de llave privada	SHA3-256
Formato de exportación de llaves y certificados	Privacy Enhaced Mail (PEM)
Contraseñas	>=12 caracteres

Tabla 6. Especificaciones técnicas para ECDSA

De acuerdo a los tamaños de llaves y algoritmos sugeridos por el NIST (National Institute of Standards and Technology) (NIST, Recommendation for key management 2016), (NIST, SHA3 standard 2015), (NIST, Digital Signature Standard (DSS) 2013) las especificaciones anteriores superan la seguridad mínima requerida en cada uno de ellos siendo como objetivo mínimo una seguridad de 128 bits.

4.4 Requerimientos de la aplicación según La Ley de Firma Electrónica

A partir del análisis de la ley y los artículos referentes a los certificados digitales se sintetizan los siguientes requerimientos a ser implementados en la aplicación

- Para la validación de ruta de certificación, la unidad de firma electrónica ejercerá como tal, este será el certificado auto firmado incluido para todas las CAs
- Los certificados de otras CAs estarán firmados por la autoridad raíz.
- Emitir certificados de acuerdo a los requerimientos de Tabla 2.
- Las llaves deberán ser generadas por el solicitante para garantizar la confidencialidad de la llave privada,
- El solicitante deberá presentar una CSR (Certificate Signing Request) para poder generarle el certificado correspondiente.
- Los certificados con base a su fecha de vencimiento deberán entrar la categoría de certificados caducados, en caso contrario formarán parte de la categoría de certificados vigentes.
- Para garantizar la privacidad de las llaves de firma de certificados de la CA, se debe implementar protocolo de almacenamiento seguro de llaves.
- Implementar un formulario para suministrar y verificar la información del signatario, incluyendo la CSR presentada.
- Crear y mantener un archivo actualizado de los certificados emitidos en medios electrónicos, para su consulta y verificación por plazo indefinido. Un usuario podría querer verificar la validez de un certificado en particular.
- La aplicación debe permitir mecanismos de actualización y revocación (cancelación de certificados) por parte de la CA previa solicitud del usuario.
- La revocación también puede darse de manera unilateral por parte de la CA si se comprueba alguna ilegalidad o vulneración del certificado, así como otros casos emitidos en la ley y deberá quedar registrada su justificación (datos falsos, vulneración los datos del proveedor, extravío o destrucción, fallecimiento, cese de actividades en una persona jurídica).

- Informar al usuario de un certificado a través de correo electrónico
    - La creación y modificación de certificados
    - La revocación de certificados
    - La suspensión de certificados
    - Vencimiento y renovación de certificados

La revocación de un certificado consiste en anular su validez completamente antes de la fecha de caducidad, lo cual deja inservible para cualquier uso legal

La suspensión de un certificado consiste en invalidar temporalmente un certificado, mientras se realizan investigaciones para determinar si se revoca o vuelve a tener efectos legales válidos.

- Contar con un mecanismo de consulta eficiente de una Lista de Revocación de certificados (CRL[8]) para los usuarios de los certificados electrónicos y evitar cualquier tipo de suplantación y/o fraude.
- El plazo de vigencia del certificado es de mutuo acuerdo entre el signatario y la CA, pero para efectos de esta aplicación, este será un parámetro configurable.

Requerimientos de roles y perfiles

Aunque la ley no explica exactamente qué medidas de seguridad implementar se consideran las siguientes:

- Manejo de sesiones de usuario con caducidad por inactividad
- Contraseñas de 12 caracteres (4 palabras aleatorias) almacenadas usando SHA3
- Manejo de roles y perfiles (Usuario, administrador, super administrador)
- Envío de token por correo para activación de cuentas

[8] Certificate Revocation List

4.5 Procesos de una CA considerando el estándar NIST PKI 800-32

Infraestructura de Llave Pública (PKI)

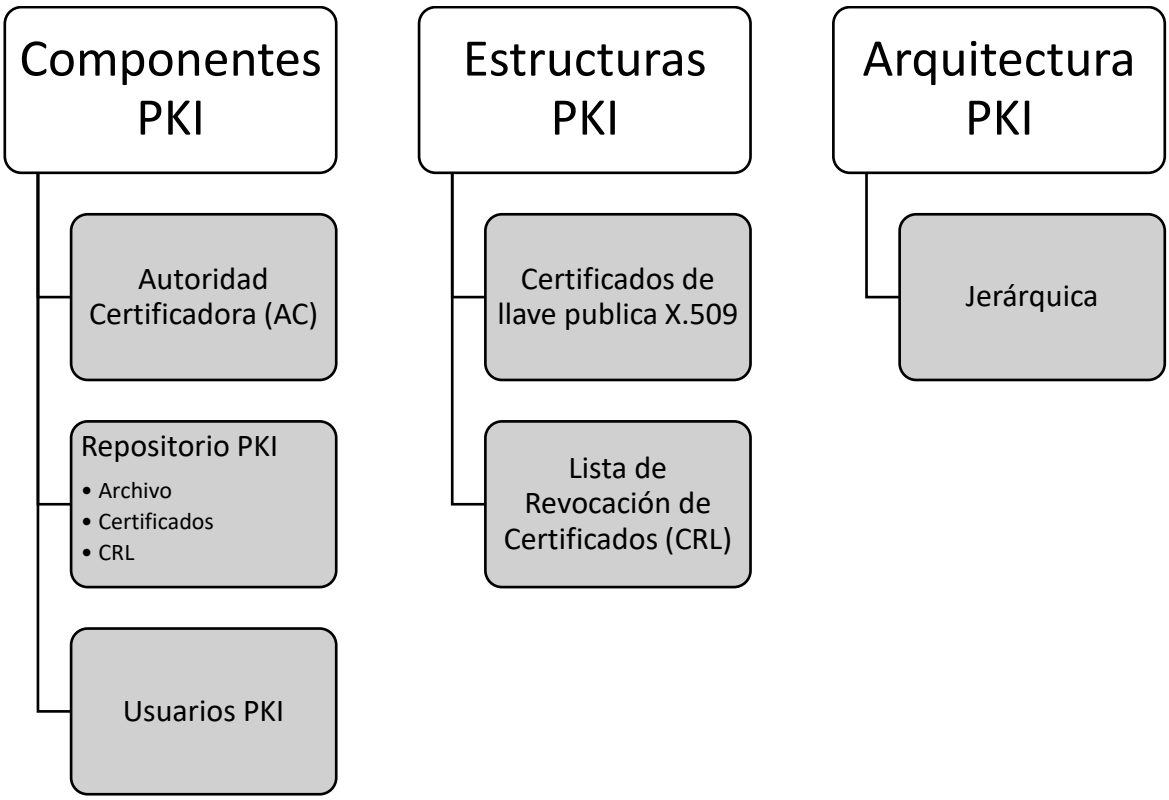

Fig 5. Infraestructura de llave pública

Arquitectura PKI: Jerárquica

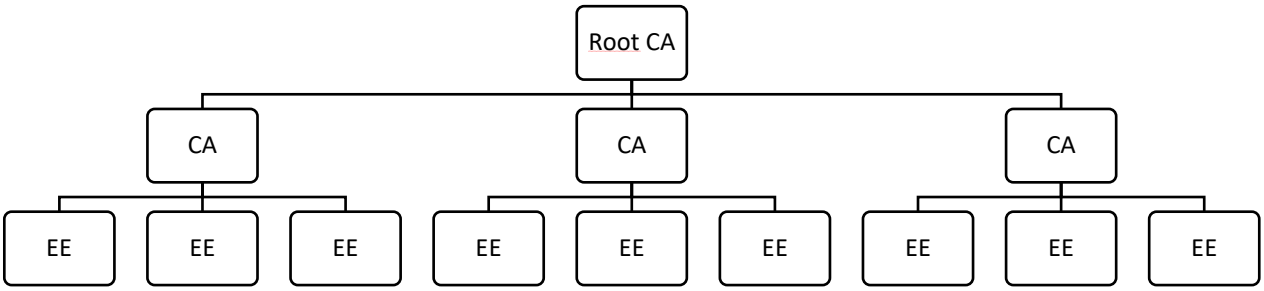

Fig 6. Arquitectura PKI Jerárquica

Referencias

Root CA: Autoridad Certificadora Raíz

CA: Autoridad Certificadora

EE: Entidad Final (End Entity)

Proceso, componentes y estructuras de una PKI

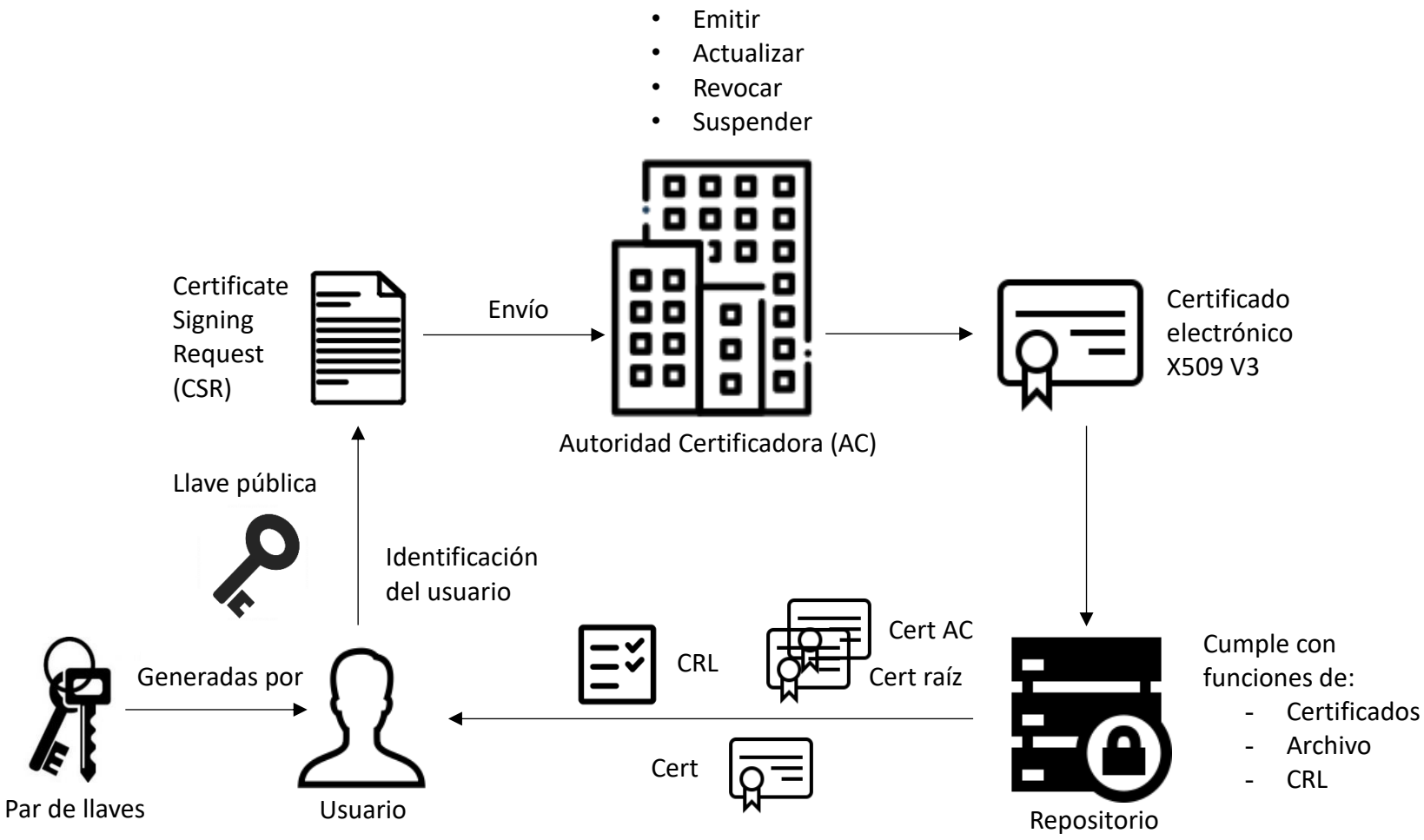

Fig 7. Proceso entre componentes y estructuras de datos de una PKI

CAPÍTULO V

PROPUESTA TÉCNICA

Capitulo V. Propuesta técnica

Para el desarrollo del software, se utiliza como referencia la metodología de desarrollo de software ágil Scrum, y basándonos en algunos de sus artefactos como: Pila del producto, Pila de Iteraciones e Iteración (Sprint) (scrum.org 2020). Para ello se establecieron tres fases: Inicio, Desarrollo y Cierre. Además, se creado un nombre ficticio para la Autoridad Certificadora: eSignAC.

5.1 Roles Equipo de Trabajo

En Scrum, el equipo se focaliza en construir software de calidad, la gestión se centra en definir cuáles son las características que debe tener el producto. El equipo entrega productos de forma iterativa e incremental, maximizando las oportunidades de obtener retroalimentación. Las entregas incrementales de producto “Terminado” aseguran que siempre estará disponible una versión potencialmente útil y funcional del producto.

Equipo Scrum	
Rol	**Responsable**
Dueño del producto	Universidad Don Bosco
Dueño del proceso	Francisco Rodríguez-Henríquez
Equipo de desarrollo	Leonel Maye Álvaro Zavala

Tabla 7. Equipo de trabajo para el desarrollo de las iteraciones

5.2 Fase de Inicio

5.2.1 Visión del producto

El producto final es una aplicación para gestionar el ciclo de vida de los certificados digitales, desde la solicitud (CSR) hasta su revocación o vencimiento, para una Autoridad Certificadora encargada de emitirlos a entidades finales,

siguiendo los estándares de seguridad mínimos en las operaciones criptográficas y de acuerdo con la Ley de Firma Electrónica de El Salvador.

5.2.2 Historias de Usuario

Las historias de usuario son una representación simple de un requisito para el proyecto de software en este caso se definen con base a los requerimientos definidos en el apartado 4.4 (Requerimientos de la aplicación según La Ley de Firma Electrónica) de este documento.

Las historias que se utilizarán son las siguientes:

1. Diseño de base de datos y administración de usuarios y roles.
2. Control de solicitudes de creación de certificados.
3. Generación de certificados para usuarios finales.
4. Consulta y verificación de certificados.
5. Revocación de certificados.

Y se desglosan en las siguientes descripciones detalladas:

<table>
<tr><th colspan="3">Historia de usuario</th></tr>
<tr><td>Número: 1</td><td colspan="2">Usuario: Equipo de trabajo</td></tr>
<tr><td colspan="3">Nombre de la historia: Diseño de base de datos y administración de usuarios y roles.</td></tr>
<tr><td>Prioridad en negocio:
Alta</td><td>Riesgo en desarrollo:
Bajo</td><td>Iteración asignada:
1</td></tr>
<tr><td colspan="3">Programador responsable: Álvaro Zavala.</td></tr>
<tr><td colspan="3">Descripción:
Se diseña la base de datos a emplear en la aplicación.
Permite la creación, modificación, bloqueo, desbloqueo, asignación de roles y activación e inactivación de usuarios, así como su búsqueda y consulta.
Incluye un auto registro de usuarios y manejo de la información de su perfil y restablecimiento de contraseñas.</td></tr>
</table>

Validación: Los usuarios serán capaces de registrarse o ser registrados y acceder a sus datos en la aplicación a través de un usuario y contraseña mostrando las opciones de acuerdo a su rol asignado.

Tabla 8. Historia de usuario Diseño de base de datos y administración de usuarios y roles

Historia de usuario		
Número: 2	**Usuario**: Equipo de trabajo	
Nombre de la historia: Control de solicitudes de creación de certificados.		
Prioridad en negocio: Alta	**Riesgo en desarrollo**: Medio	**Iteración asignada**: 1
Programador responsable: Álvaro Zavala.		
Descripción: Función para que los usuarios finales puedan: • Generar la solicitud de firma de certificados (CSR) • A partir de La CSR puedan solicitar el certificado La aplicación debe validar las CSR enviada para poder ser aceptada. Tanto los usuarios finales como los administradores deben poder consultar las CSR's con base en estados y códigos, los usuarios finales solo tienen acceso a solicitudes propias		
Validación: • Los usuarios generan y envían solicitudes válidas. • Las solicitudes están disponibles para consulta		

Tabla 9. Historia de usuario control de solicitudes de creación de certificados

Historia de usuario		
Número: 3	**Usuario**: Equipo de trabajo	
Nombre de la historia: Generación de certificados para usuarios finales.		
Prioridad en negocio: Alta	**Riesgo en desarrollo**: Medio	**Iteración asignada**: 2
Programador responsable: Leonel Maye.		
Descripción:		

Función para que los usuarios administradores puedan: • Aprobar solicitudes de creación de certificados. • Denegar solicitudes de creación de certificados. La aplicación debe validar las extensiones a incluir en cada certificado y solicitar una contraseña para descifrar la llave privada de la AC y proceder a firmar el certificado. El certificado se genera, notifica al usuario y lo almacena en un repositorio de archivos y la base de datos.
Validación: • Los administradores aprueban o deniegan las solicitudes para generar el certificado • Los certificados se generan y cumplen con el estándar X509 V3 y están firmados por la AC.

Tabla 10. Historia de usuario generación de certificados para usuarios finales.

Historia de usuario		
Número: 4	**Usuario**: Equipo de trabajo	
1. **Nombre de la historia**: Consulta y verificación de certificados.		
Prioridad en negocio: Alta	**Riesgo en desarrollo**: Medio	**Iteración asignada**: 2
Programador responsable: Leonel Maye.		
Descripción: Función para que los usuarios puedan: • Consultar los certificados propios y de terceros en el repositorio y descargarlos • Verificar la validez de un certificado, con respecto a fecha de vencimiento, firma y revocación		
Validación: • Los usuarios consultan y descargan los certificados • Los usuarios verifican el estado de un certificado devolviendo si este certificado es válido o no.		

Tabla 11. Consulta y verificación de certificados

Historia de usuario		
Número: 5	**Usuario**: Equipo de trabajo	
Nombre de la historia: Revocación de certificados		
Prioridad en negocio: Alta	**Riesgo en desarrollo**: Medio	**Iteración asignada**: 3
Programador responsable: Leonel Maye y Álvaro Zavala.		
Descripción: Función para que Los usuarios finales puedan: • Solicitar la revocación de un certificado • Descargar el repositorio de certificados revocados (CRL) • Consultar los certificados revocados Los usuarios administradores puedan: • Revocar certificados con base en razones estandarizadas • Generar el repositorio de certificados revocados (CRL) • Consultar los certificados revocados		
Validación: • Los usuarios consultan y descargan los certificados • Los usuarios verifican el estado de un certificado devolviendo si este certificado es válido o no.		

Tabla 12. Revocación de certificados

5.2.3 Pila del producto

Con base en las historias de usuario antes descritas, la Pila del producto es el conjunto de requerimientos funcionales y no funcionales, que debe cumplir el producto una vez entregado.

No	Descripción	Responsable	Estimación (Días)
1	Diseño de base de datos y administración de usuarios y roles	Álvaro Zavala	15

2	Control de solicitudes de creación de certificados	Álvaro Zavala	20
3	Generación de certificados para usuarios finales	Leonel Maye	20
4	Consulta y verificación de certificados	Leonel Maye	15
5	Revocación de certificados	Leonel Maye y Álvaro Zavala	20

Tabla 13. Pila del producto para la aplicación

5.2.4 Prioridades de la Pila del producto.

La estimación de prioridades de los requerimientos especificados en las historias de usuario se realizó utilizando la técnica de “Planning Poker” (para disponer de la estimación de tiempo requerido). Y en consenso con el equipo de trabajo para establecer la importancia de cada requerimiento. A continuación, se muestra en la siguiente tabla las prioridades enumeradas del 1 al 5, donde 1 es poco importante, 2 importante, 3 muy importante, 4 extrema importancia y 5 imprescindible.

El orden de asignación se hace en función del requerimiento, es decir ha sido asignado en razón de la experiencia del equipo de trabajo respecto a los requerimientos presentados. Se han calificado con 5 (imprescindible), los requerimientos clave. Mientras que con 4 (extrema importancia) y 3(muy importante) a los requerimientos que tienen una escala menor de importancia. No se calificó ninguno con la calificación 1, 2 y 3, ya que ningún requerimiento está clasificado en esa escala.

No	**Descripción**	**Prioridad**
1	Diseño de base de datos y administración de usuarios y roles	5
2	Control de solicitudes de creación de certificados	4
3	Generación de certificados para usuarios finales	5
4	Consulta y verificación de certificados	4

5	Revocación de certificados	5

Tabla 14. Descripción de prioridades de la Pila del producto

5.3 Fase de Desarrollo

5.3.1 Definición de las Iteraciones.

A continuación, se definen las Iteraciones de cada elemento de la Pila del producto.

ID	Requerimiento Pila del producto	Tarea de la Iteración
1	Diseño de base de datos y administración de usuarios y roles.	Diseñar base de datos para la aplicación
		Creación y modificación de usuarios y asignación de roles.
		Registro de usuarios.
		Autenticación y autorización de usuarios
		Consulta de usuarios
2	Control de solicitudes de creación de certificados.	Generar CSR
		Crear solicitud de creación de certificado
		Aprobar y denegar solicitudes
		Consultar solicitudes
3	Generación de certificados para usuarios finales.	Generar y almacenar de manera segura llave privada de la AC.
		Descifrar llave privada para firmar certificados
		Generar el archivo del certificado, almacenarlo en un repositorio y la base de datos
4		Consultar y descargar certificados

	Consulta y verificación de certificados.	Verificar certificados
		Renovar certificados
5	Revocación de certificados.	Solicitar revocar un certificado
		Generar CRL
		Descargar CRL (Punto de distribución)
		Revocar, suspender y modificar certificados

Tabla 15. División de la Pila del producto en Iteraciones

5.3.2 Desarrollo de las Iteraciones planificadas

5.3.2.1 Iteración 1

A continuación, se presenta la Pila de la Iteración 1 que servirá para identificar las tareas en cada requerimiento que lo conforma.

ID	Requerimiento Pila del producto	Tarea de la Iteración
1	Diseño de base de datos y administración de usuarios y roles.	Diseñar base de datos para la aplicación
		Creación y modificación de usuarios y asignación de roles.
		Registro de usuarios.
		Autenticación y autorización de usuarios
		Consulta de usuarios
2	Control de solicitudes de creación de certificados.	Generar CSR
		Crear solicitud de creación de certificado
		Consultar solicitudes
		Aprobar y denegar solicitudes

Tabla 16. Pila de la Iteración 1

Tarea 1.1. Diseñar la base de datos para la aplicación

La base de datos diseñada basada en los requerimientos de las secciones 4.4 y 4.5 de este documento es la siguiente:

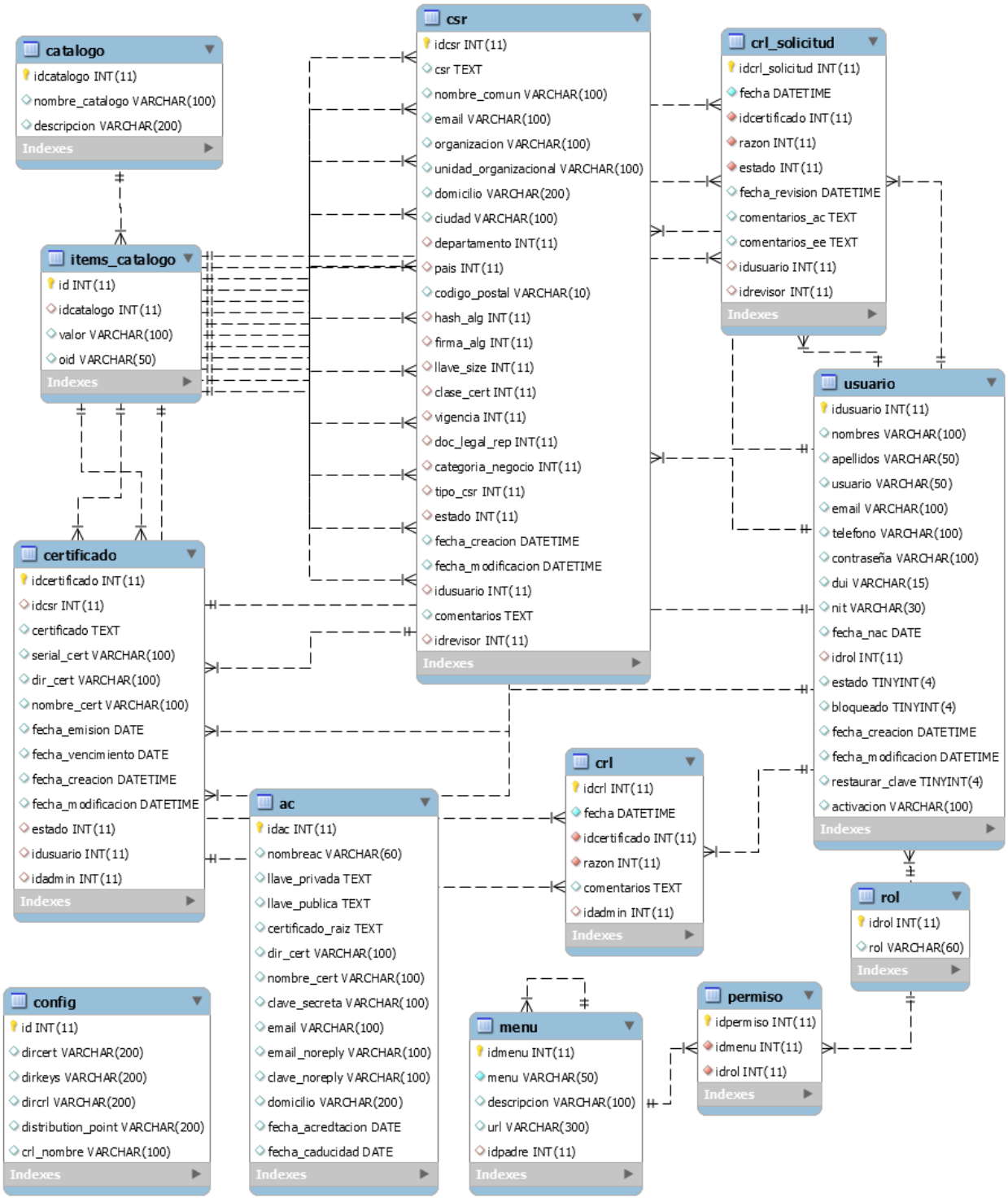

Fig 8. Diagrama Entidad Relación usado por la aplicación.

Tarea 1.2. Creación y modificación de usuarios y asignación de roles.

Con base en las tablas menú, permiso, rol y usuario se creó una configuración de permisos sobre las diferentes opciones de la aplicación, para este caso se definieron tres roles: END_USER, ADMIN y SA (Super Admin). Los permisos para cada rol se describen a continuación:

No.	Permiso	Id padre	Roles
1	Perfil		END_USER ADMIN SA
2	CSR		END_USER ADMIN SA
3	Certificados		END_USER ADMIN SA
4	CRL		END_USER ADMIN SA
5	Usuarios		SA
6	Mi Perfil	1	END_USER ADMIN SA
7	Cambiar Contraseña	1	END_USER ADMIN SA
8	Mis Certificados	1	END_USER
9	Crear CSR	2	END_USER
10	Solicitar certificado (CSR)	2	END_USER
11	Solicitudes pendientes (CSR)	2	ADMIN SA
12	Solicitudes enviadas (CSR)	2	END_USER
13	Historial de solicitudes (CSR)	2	ADMIN SA
14	Eliminar envío no procesado (CSR)	2	END_USER
15	Ver CSR	2	END_USER ADMIN SA
16	Aprobar o Denegar CSR	2	ADMIN SA
17	Renovar Certificado	3	END_USER
18	Descargar Certificado	3	END_USER ADMIN SA
19	Abrir Certificado	3	END_USER ADMIN SA
20	Solicitar Revocación	3	END_USER
21	Verificar Certificado	3	END_USER ADMIN SA
22	Repositorio de Certificados	3	END_USER ADMIN SA
23	Repositorio CRL	4	END_USER ADMIN SA
24	Solicitudes de revocación enviadas	4	END_USER
25	Generar CRL	4	ADMIN SA
26	Descargar CRL		END_USER ADMIN SA
27	Solicitudes CRL Pendientes	4	ADMIN SA
28	Aprobar o Denegar Solicitudes CRL	4	ADMIN SA
29	Historial de solicitudes CRL Pendientes	4	ADMIN SA
30	Lista de usuarios (Modificar, Eliminar, Bloquear y Desactivar, Asignar rol)	5	SA
31	Registrar usuario	5	SA

Tabla 17. Permisos asignados a cada rol.

Descripción de los roles

- END_USER: Rol asignado a los usuarios finales de un certificado.
- ADMIN: rol asignado para gestionar las operaciones de las CSR's, Certificados y CRL.
- SA: rol definido para realizar las operaciones de una ADMIN más las operaciones de gestión de usuarios.

Para la creación, modificación, eliminación, bloqueo, desactivación y asignación de roles se creó el siguiente formulario:

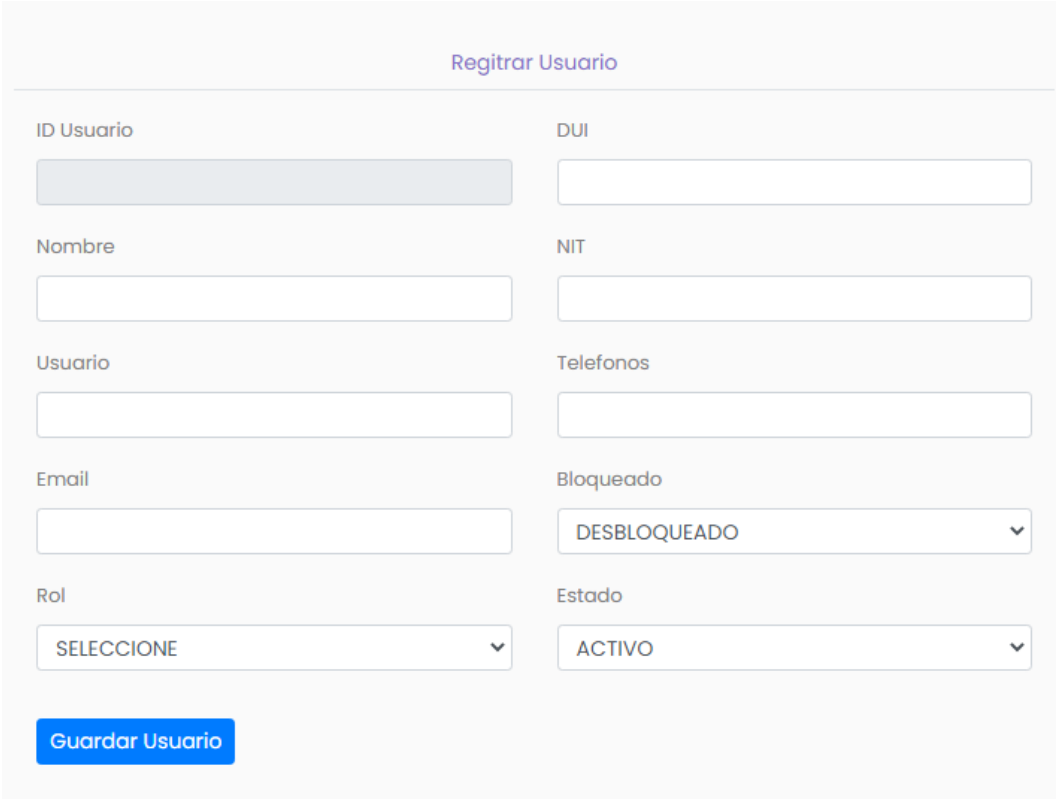

Fig 9. Formulario de registro de usuario por un SA.

Tarea 1.3. Registro de usuarios.

Para los usuarios finales (END_USER) se desarrolló el siguiente formulario de registro, el cual almacena las contraseñas con SHA-256 y una salt. Los usuarios y correos deben ser únicos.

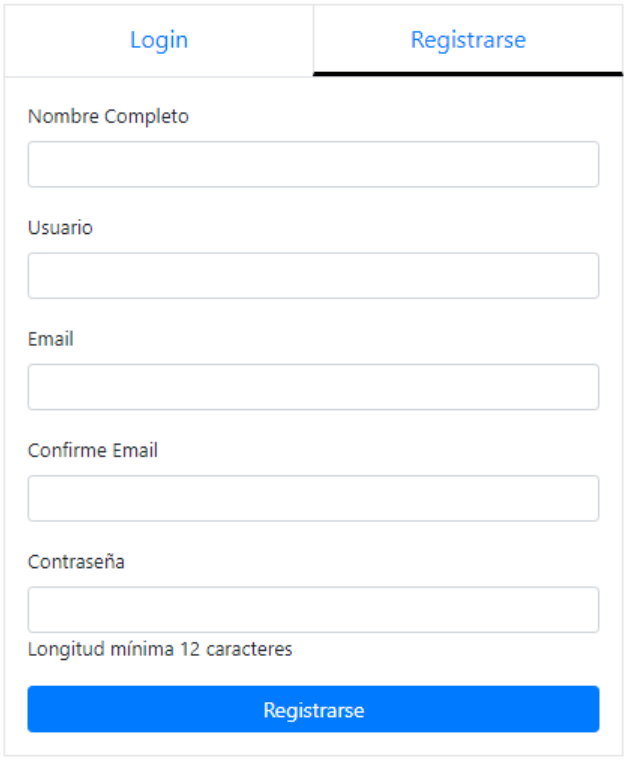

Fig 10. Formulario de auto registro de usuarios finales.

Tarea 1.4. Autenticación y autorización de usuarios

Para la autenticación de usuarios se programó la siguiente interfaz, que válida también como nombre de usuario el correo registrado, adicionalmente cuenta con 5 intentos, posteriores a los cuales se pedirá un captcha, y posterior a 20 intentos el usuario se bloqueará.

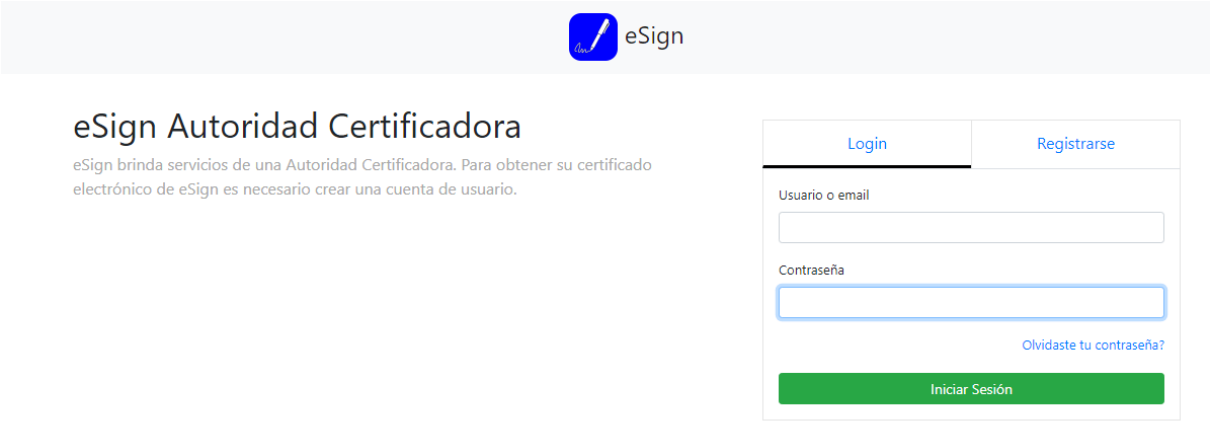

Fig 11. Interfaz de inicio de sesión.

Al iniciar sesión el usuario aterriza en la siguiente página, donde se muestra en el panel de la derecha, un menú desplegable con los permisos asignados al rol del usuario y en la esquina superior derecha se visualiza el nombre del usuario, usuario y opciones de perfil y cerrar sesión

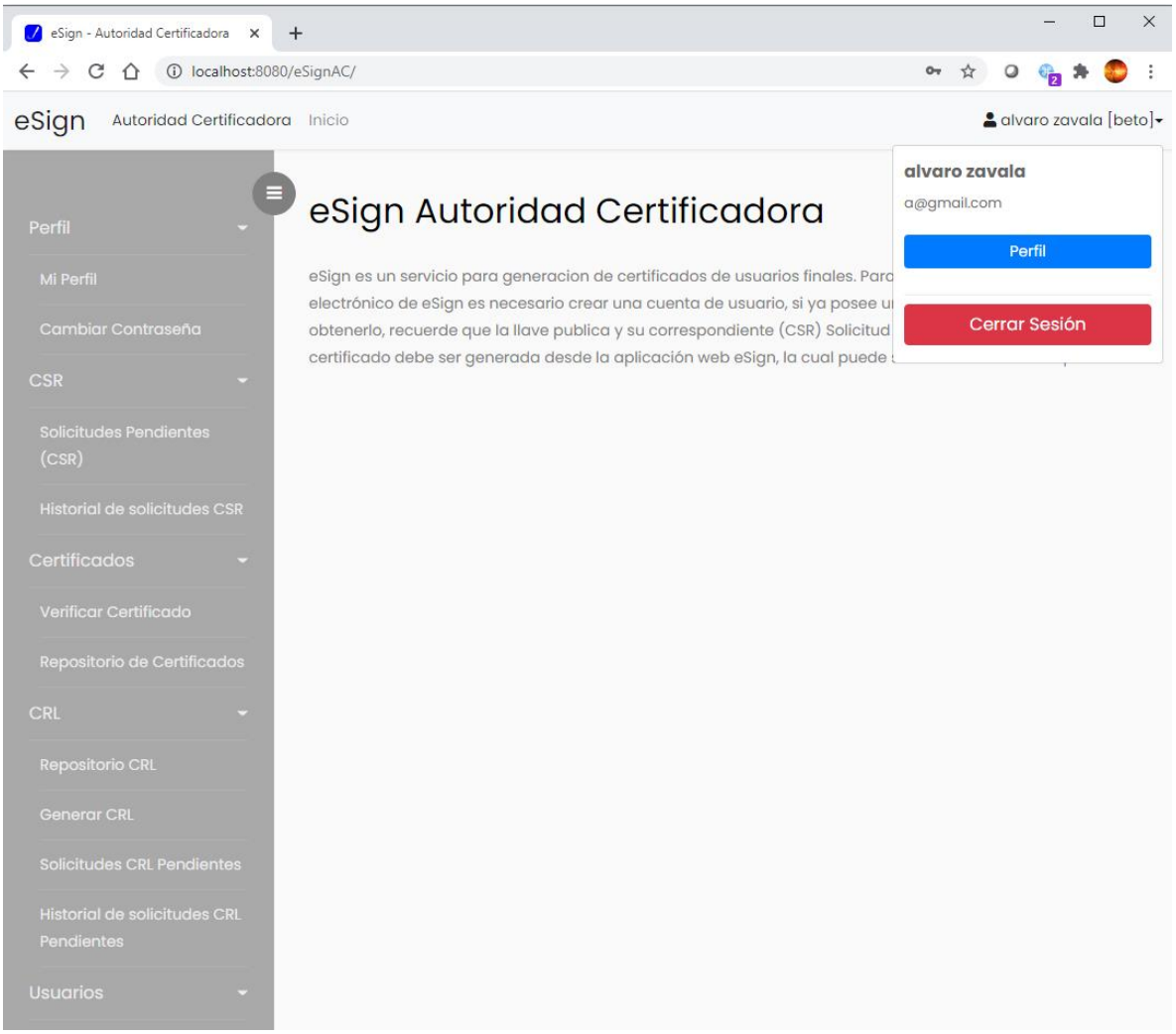

Fig 12. Página principal de la aplicación.

En el caso de la autorización, esta se basa en la Tabla 15 de permisos de este documento, y para la validar tales permisos se hace uso de un filtro web de Java, una variable de sesión que almacena los permisos para el usuario logueado

y una variable con los permisos que no necesitan inicio de sesión, denegando aquellos permisos no asignados. El siguiente código es un extracto del filtro usado:

```
public void doFilter(ServletRequest request, ServletResponse response,
            FilterChain chain)
            throws IOException, ServletException {
        httpRequest = (HttpServletRequest) request;
        httpResponse = (HttpServletResponse) response;

…
        HttpSession s = httpRequest.getSession();

        String urlDenegacion =
httpRequest.getContextPath()+"/no_autorizado/denegado";
        String[] urlPermitidas = {
            "/users/sign_in",
            "/logout.jsp",
            "/users/Login",
            "/",
            "/users/activar",
            "/certs/download_cert",
            "/csr/download_privkey",
            "/users/reset_pass",
            "/users/set_newpass",
            "/crl/download_eSignCRL",
            "/css/style.css",
            "/js/main.js",
            "/certs/descargar.jsp",
            "/botdetectcaptcha"
        };
        boolean isLoggedIn = (s.getAttribute("usuario") != null);
        try {
            System.out.println("URI:"+httpRequest.getRequestURI());
            if (isLoggedIn){
                if (httpRequest.getRequestURI().equals(urlDenegacion))
                    chain.doFilter(request, response);
                boolean go = false;
                List<Menu> permisos =
(List<Menu>)s.getAttribute("permisos");
                if (permisos!=null)
                for (Menu m: permisos){
                    if
(httpRequest.getRequestURI().equals(m.getUrl()==null?"":httpRequest.getCo
ntextPath()+m.getUrl())){
                        go=true;
                    }else{

                    }
                }
                    for (String url: urlPermitidas){
                        if
(httpRequest.getRequestURI().equals(httpRequest.getContextPath()+url)){
                            go=true;
                        }
                    }
```

```
                ...
                if (go){
                    chain.doFilter(request, response);
                }else{
                    httpResponse.sendRedirect(urlDenegacion);
                }
            }
            else{
                chain.doFilter(request, response);
            }
        } catch (Throwable t) {

            problem = t;
        }

    }
```

Fig 13. Fragmento de código de filtro de autorización.

La página mostrada al denegar un servicio se muestra a continuación.

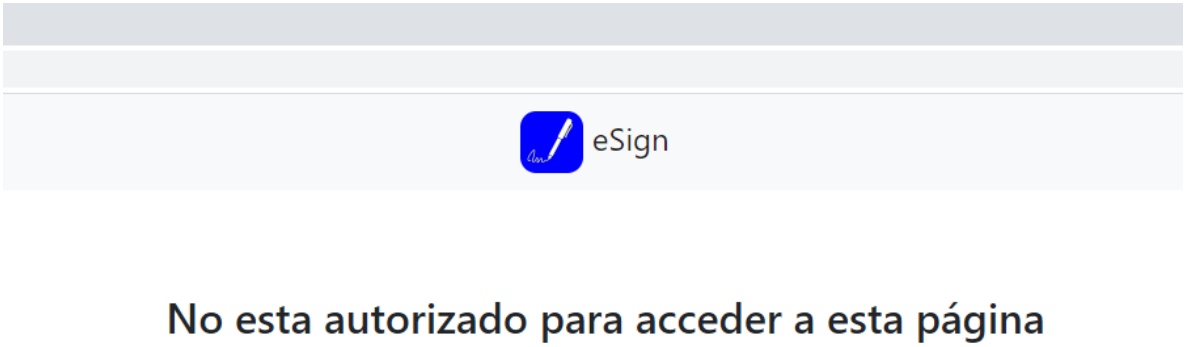

Fig 14. Página de acceso no autorizado.

Tarea 1.5. Consulta de usuarios

Para la consulta y búsqueda de usuario se desarrolló la siguiente opción:

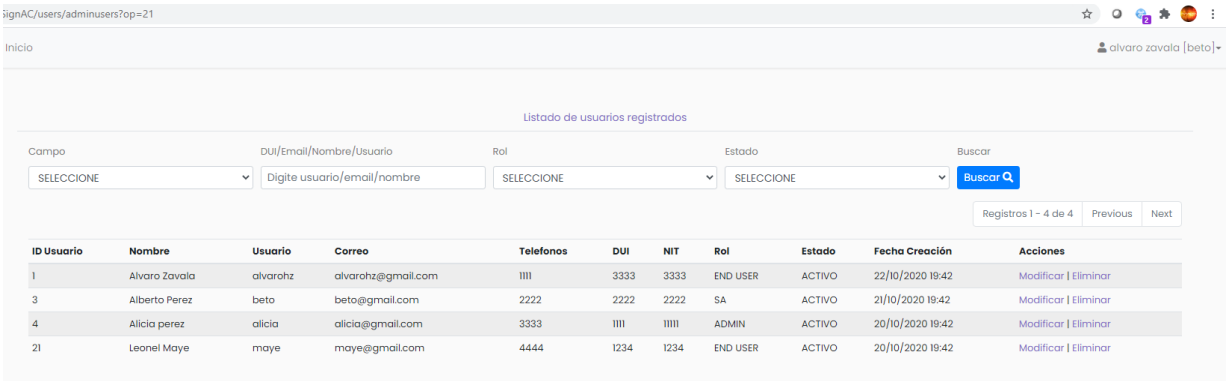

Fig 15. Interfaz para consulta de usuarios.

Tarea 2.1. Generar CSR

Para proseguir con los componentes creados para la aplicación a continuación se explican las clases que dan soporte a las acciones del usuario, las operaciones con la base de datos, y las operaciones criptográficas, se debe mencionar que estas clases se usan a lo largo de todo el proyecto, pero solo se presentan en este apartado particular.

Para el presente proyecto se hace uso del patrón de desarrollo MVC y la API de Persistencia de JAVA (JPA), cada módulo de la aplicación utiliza la siguiente arquitectura:

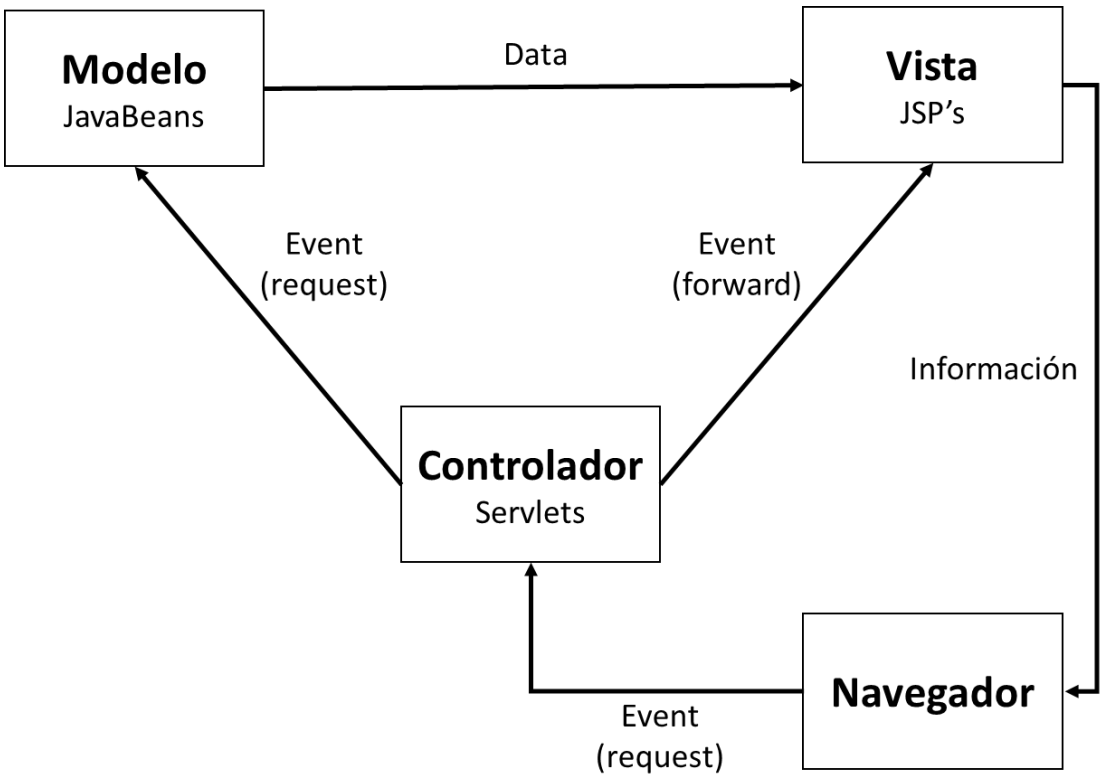

Fig 16. Patrón de desarrollo MVC empleado.

Operaciones criptográficas Implementadas con Bouncy Castle API

La API Bouncy Castle cuenta con versiones en C# y Java, la versión en Java permite desarrollo multiplataforma, y entre las primitivas criptográficas ofrecidas están las siguientes: criptografía de llave pública, criptografía de llave privada, funciones hash, códigos de autenticación de mensajes (MAC), firma digital y estampa de tiempo.

Algunos puntos importantes de la librería son: énfasis en el cumplimiento de estándares y normas, amplia documentación disponible en internet e integración transparente con la librería nativa security de java.

Con respecto a las funciones hash y firmas digitales, primitivas usadas en el proyecto, se destaca lo siguiente:

- Entre las funciones hash esta SHA3, pero requirió hacer adecuaciones debido a que la entrada es de cierto tipo y tamaño, así mismo la salida debe ser codificada de acuerdo a los requerimientos de la aplicación.
- La clase que permite implementar firmas digitales cuenta con distintos algoritmos estándares como DSA y ECDSA para el proyecto se utilizó el segundo. Entre las curvas disponibles están las señaladas por los estándares NIST y SEC, pero fue necesario escribir una implementación para procesar las entradas y salidas del algoritmo al formato necesario y de la misma forma para el algoritmo de verificación.
- La generación del par de llaves privada-pública fue acondicionada para integrarse con el algoritmo de firma y se utilizó el generador de números aleatorios que viene por defecto, además se codificó su salida.

En general para las implementaciones disponibles se requiere escribir código para acondicionarlo al caso específico de aplicación debido a que las entradas y salidas por defecto esperan cierto tipo y tamaño.

Diagrama de clases para operaciones criptográficas

Para las operaciones criptográficas se utilizan las clases del siguiente diagrama:

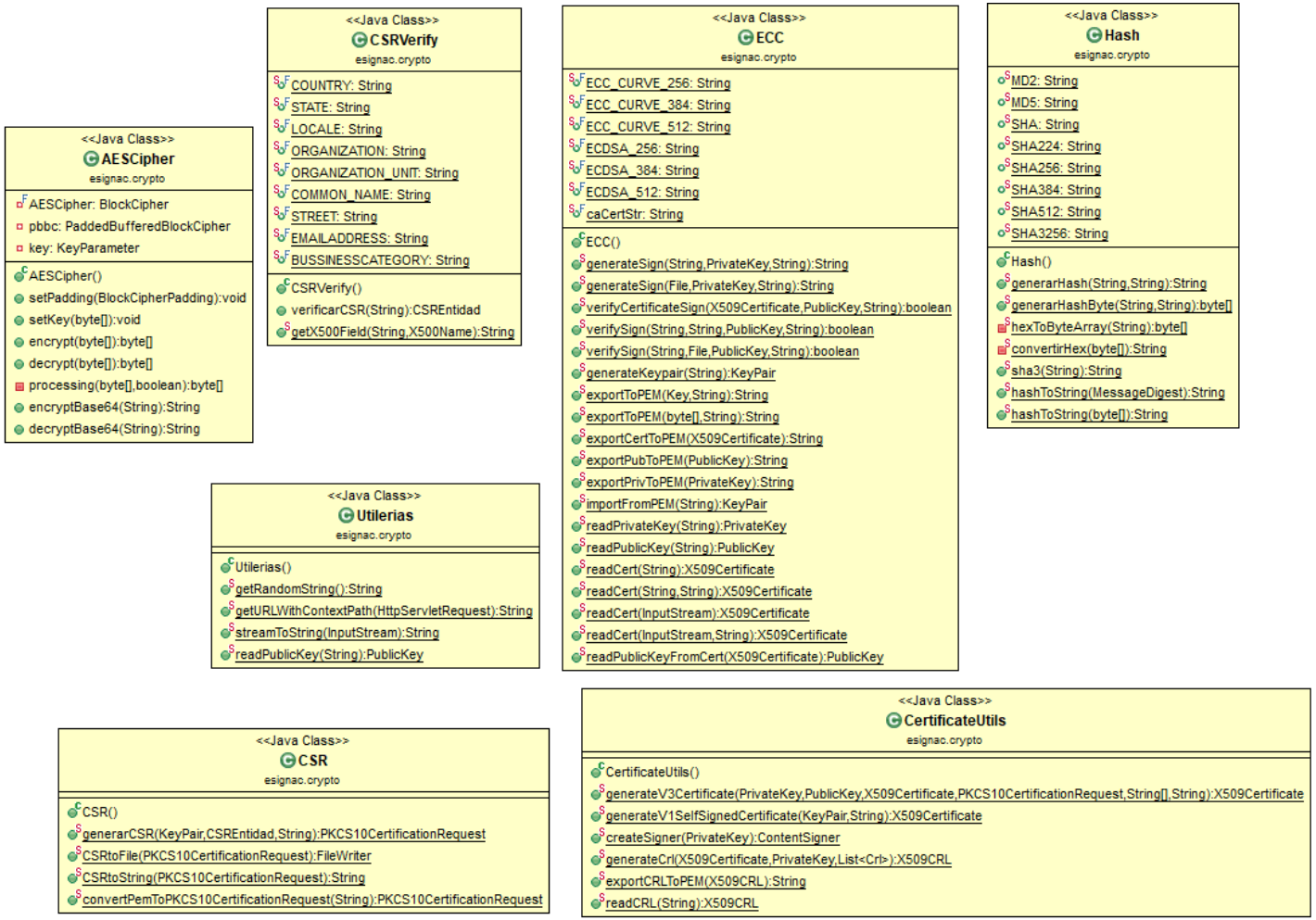

Fig 17. Diagrama de clases para operaciones criptográficas

Interfaz para generación de CSR

Los usuarios finales pueden hacer uso de la siguiente pantalla para generar una CSR (solicitud de firma de certificado), las opciones de Algoritmo, Tamaño de llave y Hash disponibles son las siguientes de acuerdo a los requerimientos de las secciones 4.2 y 4.3 de este documento:

Algoritmo de firma: *ECDSA*

Tamaño de llave: *Secp256r1, Secp384r1 y Secp521r1*

Algoritmo hash: *SHA3-256, SHA3-384 y SHA3-512*

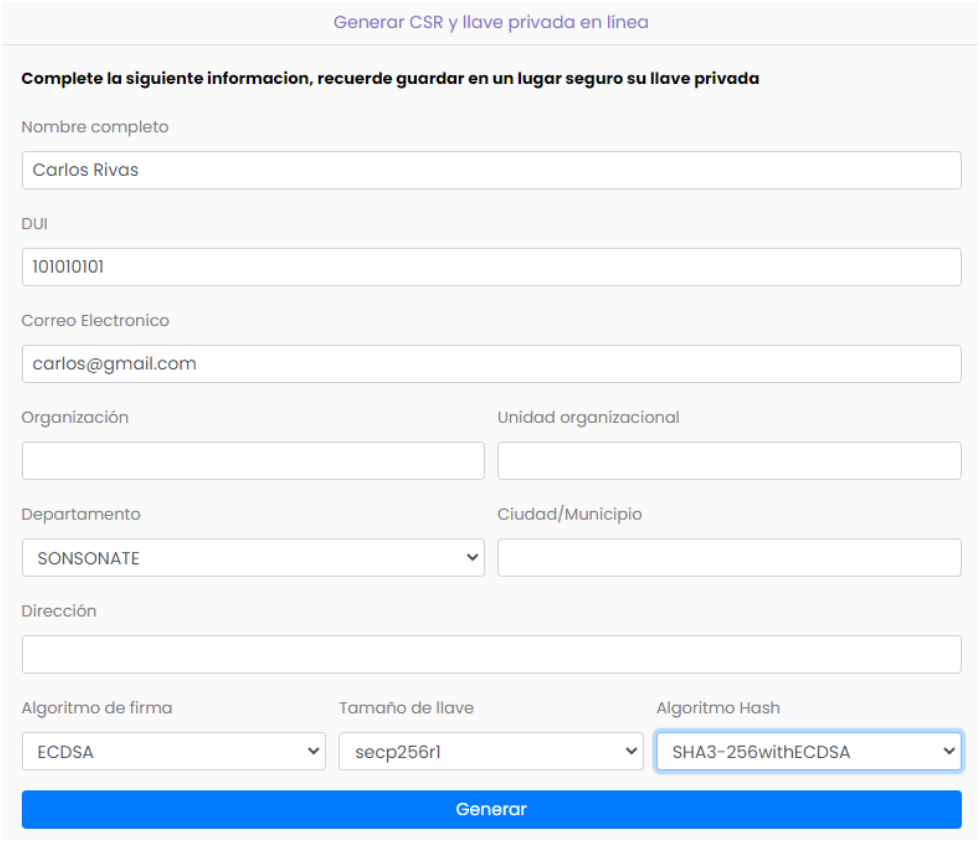

Fig 18. Interfaz para generación de CSR

El usuario genera una CSR la cual puede:

1. Copiar y pegar para solicitar un certificado a través de la misma aplicación.

2. Descargar la llave privada vinculada a la CSR (esta solo se muestra en esta única ocasión así que es importante que el usuario la almacene de manera segura).

La CSR contiene los datos solicitados según Fig. 18 y está firmada con la llave privada generada que lo vincula a la llave publica incluida en la misma CSR.

La Imagen siguiente muestra un ejemplo de la CSR y llave privada generada en formato PEM.

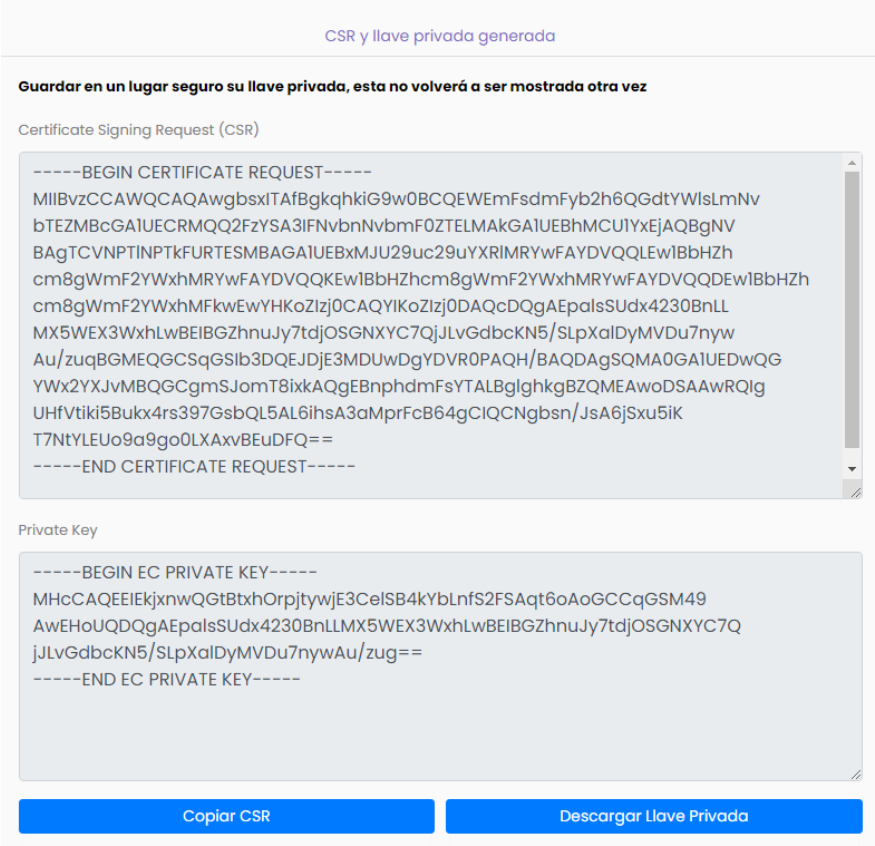

Fig 19. Ejemplo de llave privada y CSR generada.

Tarea 2.2. Crear solicitud de creación de certificado

Posterior a la generación de una CSR valida el usuario debe emplear el siguiente formulario que recopila información adicional requerida según la sección 4.1 de este documento en relación al contenido de un certificado.

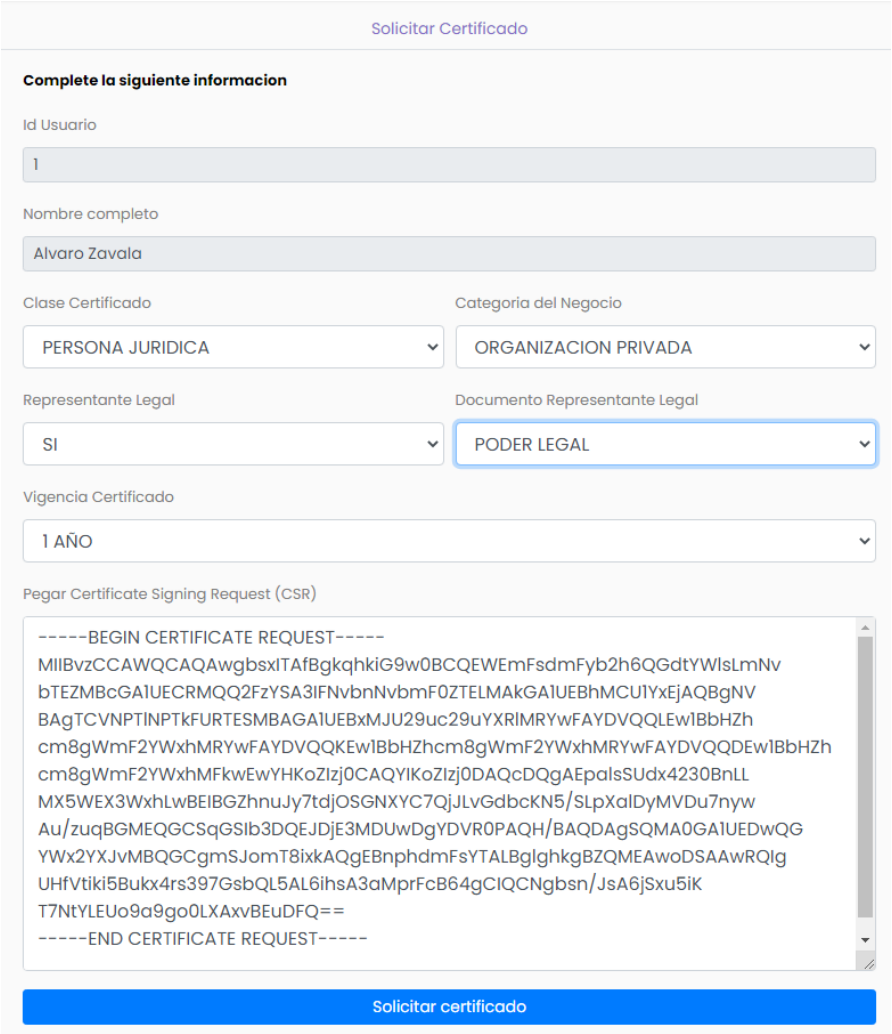

Fig 20. Formulario de solicitud de certificado

La información adicional para solicitar un certificado se detalla como sigue:

- Clase del certificado: Persona natural o jurídica.
- Categoría del negocio: Organización pública, privada o no comercial.
- Representante legal: si la persona representa a otra o a una organización.
- Documento representante legal: documento legal presentado que lo acredite como representante de otra entidad.
- Vigencia del certificado: 1, 2 o 3 año(s).

Validación de la solicitud CSR y envío de correo electrónico

Para poder aceptar la solicitud se corre un proceso de validación que revisa los siguientes puntos:

1. Validación de la firma en la CSR que vincula al usuario con su llave privada y la llave publica consignada en ella.
2. La validación de la firma proporciona garantía de que la información consignada en la CSR no ha podido ser alterada, por lo tanto, se considera integra.
3. Se verifica que los algoritmos y tamaños de llave utilizados obedezcan a los definidos para la Autoridad Certificadora, los cuales se consideran los autorizados para generar certificados.
4. Se verifica cada uno de los campos de la CSR garantizando que contiene los mínimos campos definidos en Fig.19.

Si la validación de la solicitud es exitosa se envía un correo electrónico al usuario para notificarle que su solicitud ha sido aceptada para ser candidata a generación del certificado correspondiente, y posterior a validaciones de identidad que debería correr la AC citando al usuario de manera presencial con la documentación pertinente.

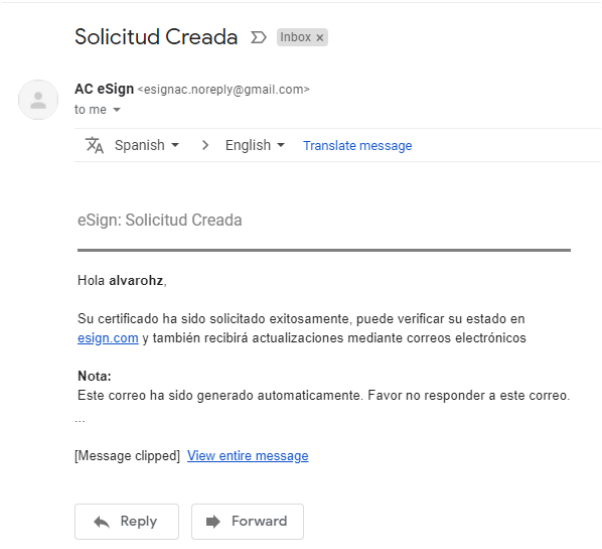

Fig 21. Ejemplo de correo enviado por solicitud aceptada.

Tarea 2.3. Consultar solicitudes CSR

La consulta de solicitudes es posible realizarla con diferentes filtros de: ID Solicitud, Clase certificado, Estado de la solicitud, Fecha, DUI y Nombre.

Los estados de la solicitud son: Solicitado, Validación, Denegada y Aceptada.

A continuación, se muestra la interfaz que permite consultar certificados

Fig 22. Interfaz de consulta de solicitudes de certificados.

Los usuarios finales solo podrán consultar solicitudes propias, mientras que los administradores pueden ver solicitudes de todos los usuarios, aprobarlas o denegarlas. La última columna de la derecha (revisar) está disponible solo para administradores (ADMIN, SA).

Tarea 2.4. Aprobar y denegar solicitudes

Haciendo click en revisar (última columna Fig. 21) de cada solicitud un administrador puede aprobar o denegar una solicitud y un correo es enviado al usuario cuando se aprueba e incluye el archivo generado del certificado mismo.

La funcionalidad que permite este proceso es la siguiente:

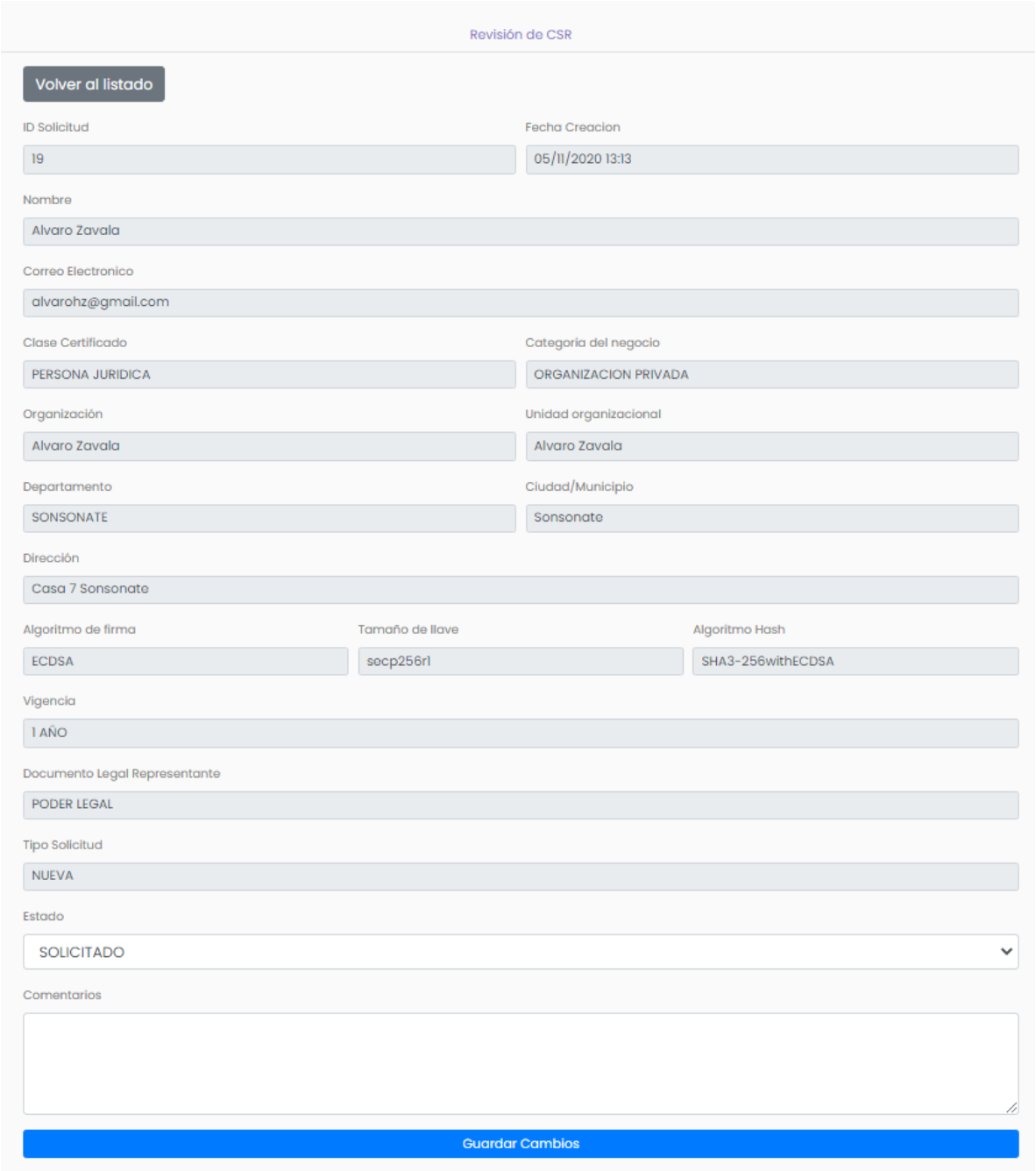

Fig 23. Revisar solicitud CSR para aprobación o denegación

En Fig. 22 se modifica el estado de la solicitud, si es aprobado se solicita una contraseña para descifrar la llave privada de la AC, la cual permanece cifrada en la base de datos, y luego proceder a generar el certificado firmado, el cual se envía al usuario por correo y también se muestra en la opción de *mis certificados* de cada usuario.

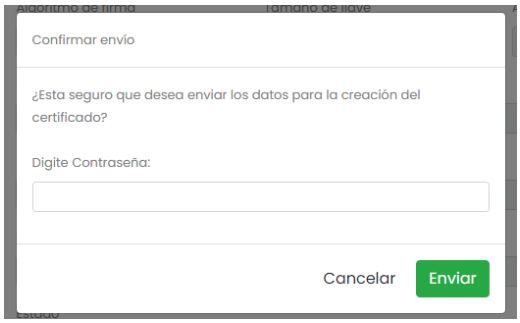

Fig 24. Solicitud de contraseña para usar llave privada de la AC.

Fig 25. Ejemplo de correo enviado incluyendo el archivo .crt del certificado

5.3.2.2 Iteración 2

A continuación, se presenta el Pila de la Iteración *2* que servirá para identificar las tareas en cada requerimiento que lo conforma.

ID	Requerimiento Pila del producto	Tarea de la Iteración
3	Generación de certificados para usuarios finales.	Generar y almacenar de manera segura llave privada de la AC.
		Descifrar llave privada para firmar certificados
		Generar el archivo del certificado, almacenarlo en un repositorio y la base de datos

4	Consulta y verificación de certificados.	Consultar y descargar certificados
		Verificar certificados
		Renovar certificados

Tabla 18. Pila de la Iteración 2

Tarea 3.1. Generar y almacenar de manera segura llave privada de la AC.

Para crear la llave privada y el certificado de la AC se hizo uso un programa en consola que utiliza las clases criptográficas creadas en el primer sprint:

```
public class GenCerts {
    public static void main(String[] args) throws
NoSuchAlgorithmException, NoSuchProviderException,
InvalidAlgorithmParameterException, OperatorCreationException,
IOException, CertificateException, Exception {
        Security.addProvider(new
org.bouncycastle.jce.provider.BouncyCastleProvider());

        KeyPair keyPair = ECC.generateKeypair();
        X509Certificate caCert =
CertificateUtils.generateV1SelfSignedCertificate(keyPair, "eSign");

        System.out.println("=");
        System.out.println("CERTIFICATE PEM (to store in a cert-
johndoe.pem file)");
        System.out.println("=");
        System.out.println();
        org.bouncycastle.openssl.jcajce.JcaPEMWriter pemWriter = new
org.bouncycastle.openssl.jcajce.JcaPEMWriter(new
PrintWriter(System.out));
        System.out.println(ECC.exportCertToPEM(caCert));

        System.out.println("=");
        System.out.println("PRIVATE KEY PEM (to store in a priv-
johndoe.pem file)");
        System.out.println("=");
        System.out.println();
        String kpriv = ECC.exportToPEM(keyPair.getPrivate(), "PRIVATE
KEY");
        System.out.println(kpriv);

        System.out.println("=");
        System.out.println("Public key from pem file");
        System.out.println("=");
        System.out.println();
        String kpub = ECC.exportToPEM(keyPair.getPublic(), "PUBLIC KEY");
        System.out.println(kpub);
    }
}
```

Fig 26. Generación de llaves privada-pública y certificado de la AC.

Tanto la llave privada, pública y el certificado se almacenan en la base de datos con la única diferencia que la llave privada se almacena cifrada y cada vez que se va a utilizar para generar un certificado se solicita una contraseña (llave secreta) para obtener la llave privada en claro y poder firmar los certificados emitidos como debe ser. El siguiente diagrama de bloques explica este proceso de almacenamiento de la llave privada.

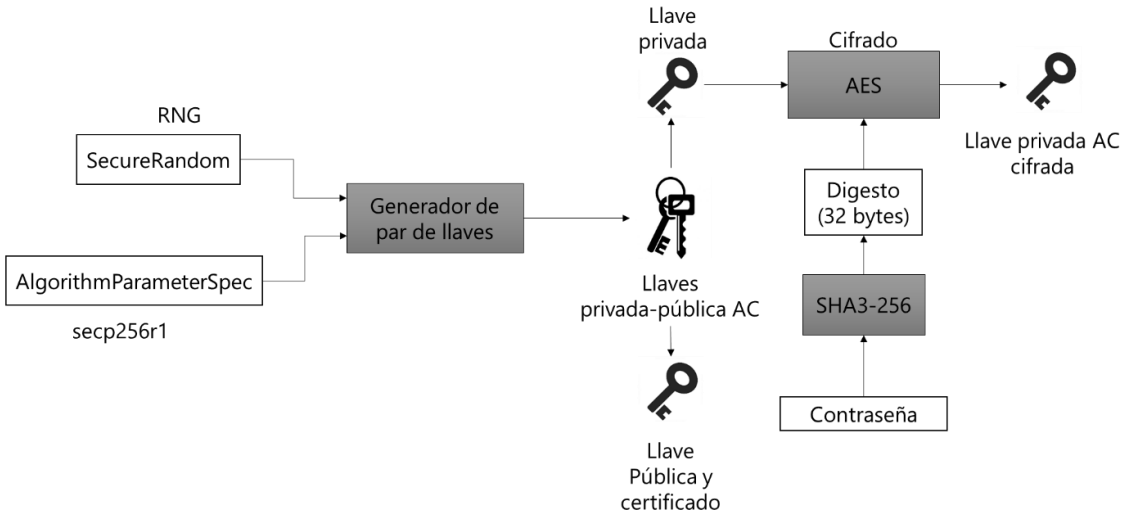

Fig 27. Proceso de generación y almacenamiento seguro de llave privada AC.

Aplicación de cifrado para llave privada

Para el cifrado de la llave privada se creó el siguiente programa con la tecnología de Swing de Java para generar la cifra a guardar en la base de datos.

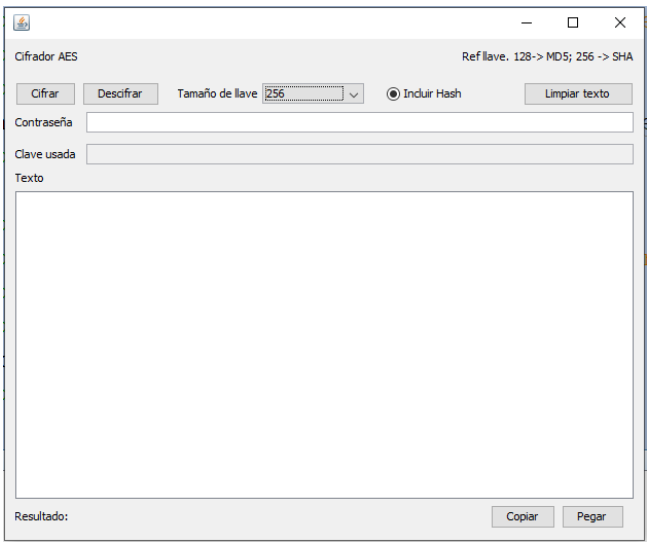

Fig 28. Aplicación para cifrador AES usado en llave privada AC.

Tarea 3.2. Descifrar llave privada para firmar certificados.

El siguiente diagrama explica el proceso para descifrar la llave privada de la AC para poder firmar los certificados emitidos a usuarios finales

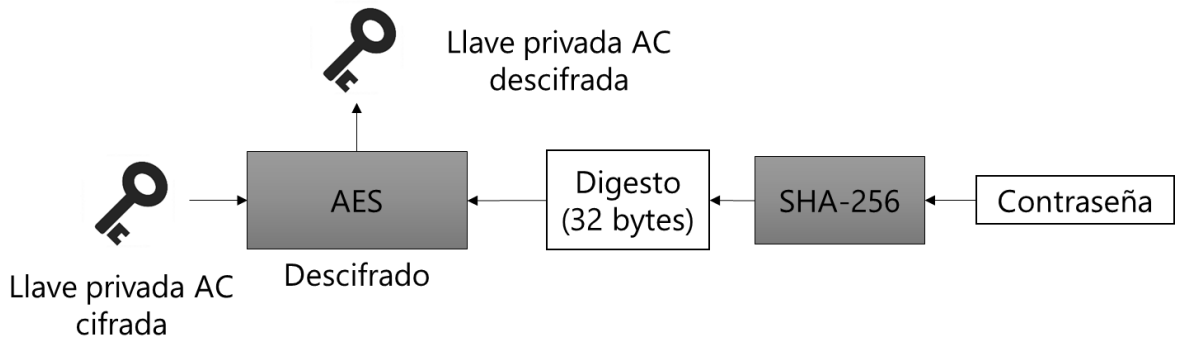

Fig 29. Descifrado de la llave privada de la AC para firmar certificados emitidos.

El protocolo anterior garantiza el resguardo y secreto de la llave privada de la AC únicamente para aquellas personas autorizadas a emitir los certificados.

Tarea 3.3. Generar el archivo del certificado, almacenarlo en un repositorio y la base de datos.

En Fig. 8 se muestra el diagrama de la base de datos donde se identifica una tabla llamada config, en cuyo campo *dircert* como parte de las configuraciones globales de la aplicación. Este campo guarda el directorio de archivos donde los certificados una vez son generados se guardan con un nombre único y extensión .crt, adicionalmente la tabla *certificado* almacena el certificado en formato PEM. Esto sirve de **repositorio de certificados** para consulta.

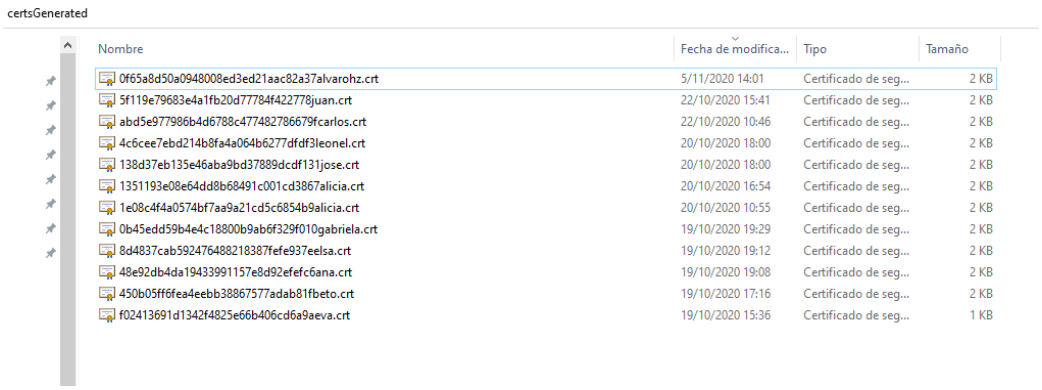

Fig 30. Ejemplo del repositorio de certificados

Los certificados generados son X.509 V3 y se considera los requerimientos de la sección 4.1, 4.2 y 4.3 de este documento. A continuación, se muestra un ejemplo de un certificado generado.

Version: 3
Número Serial: 1395066111
Algoritmo de firma: 2.16.840.1.101.3.4.3.10(SHA3-256withECDSA)
Autoridad Emisora: CN=eSing, O=eSing Inc., OU=eSing Certification, L=San Salvador, ST=San Salvador, C=SV, STREET=Calle 1 Edificio 2, EMAILADDRESS=esign@esign.com

Validez

- Fecha Emision: 05/11/2020
- Fecha Expiracion: 05/11/2021

Sujeto: CN=Alvaro Zavala, O=Alvaro Zavala, OU=Alvaro Zavala, L=Sonsonate, ST=SONSONATE, C=SV, STREET=Casa 7 Sonsonate, EMAILADDRESS=alvarohz@gmail.com
UserID: DUI=3333

Info llave pública del sujeto

- Algoritmo: EC
- Tamaño llave: 256
- Llave pública: 3059301306072a8648ce3d020106082a8648ce3d03010703420004a5a96c49477 1e36df40672cb317e56117dd6c612f0044201199867b89cbbb5d8ce486357602ed08c92ef19d6dc28de7f48ba576a50f23150eeee7cb002eff3ba
- Curva: secp256r1

Extensiones

- Uso Firma: SI
- Uso Cifrado: SI
- Categoria negocio: ORGANIZACION PRIVADA
- Clase: PERSONA JURIDICA
- Documento Representante Legal: PODER LEGAL
- Punto de distribucion CRL: http://localhost:8080/eSignAC/crl/download_eSignCRL

Firma emisor

- Algoritmo de firma: SHA3-256withECDSA
- Firma: 3044022059a8db863de00198a9cdb2b72bb621cf46b20793d2e077a9fc897bef02c6553a0220717a074cd695ed886b6bae7edeec6763706 1f3da475af413182cfc6d782abd80

Fig 31. Ejemplo de certificado generado por la aplicación.

Las extensiones personalizadas mostradas en Fig. 31 son las siguientes:

- KeyUsage (Uso de la llave): Firma y Cifrado.
- Categoría del negocio: privado, público o no comercial.
- Clase: Persona jurídica o natural.
- Documento representante legal: definidos por la legislación del país.
- UserID: almacena el DUI del usuario final.
- Punto de distribución CRL: URL de descarga de la lista de certificados revocados.

Cada extensión de acuerdo con el estándar X.509 v3 obedece a un OID (Object Identifier) que consiste en un nodo en un espacio de nombres asignados.

Entre los campos importantes a resaltar del certificado están el serial y la dirección electrónica que garantizan la unicidad de cada certificado.

Tarea 4.1. Consultar y descargar certificados.

Para esta función se hace uso del repositorio de certificados y se ponen a disposición los certificados de los usuarios para ser consultados (buscados) y descargados.

Consulta de certificados

La consulta se puede realizar por el serial, clase, estado y fecha emisión. El estado de un certificado puede ser: Vigente, Expirado, Suspendido o Revocado y se explican a continuación (esto también cumple con los requerimientos de la sección 4.1 de este documento):

- **Vigente**: Certificado con fechas de validez verificadas y de uso seguro.
- **Expirado**: Certificado válido cuya fecha de vencimiento se ha cumplido y ya no es posible usarlo para operaciones posteriores a esa fecha, pero que avala todas las operaciones hechas antes de esa fecha y debe permanecer en el repositorio.
- **Suspendido**: Una revocación temporal que indica que una AC no responderá por ese certificado en un momento específico (no será válido). Una vez que

se revoca un certificado con un código de motivo de CertificateHold (Suspendido), el certificado se puede revocar con otro código, o anular su revocación y volver a utilizarse.

- **Revocado**: Un certificado revocado significa que ha dejado de ser confiable y la AC no responderá por ese certificado, pero las operaciones anteriores a la fecha de revocación siguen siendo válidas, por lo que pasa al archivo de certificados revocados conocido como CRL.

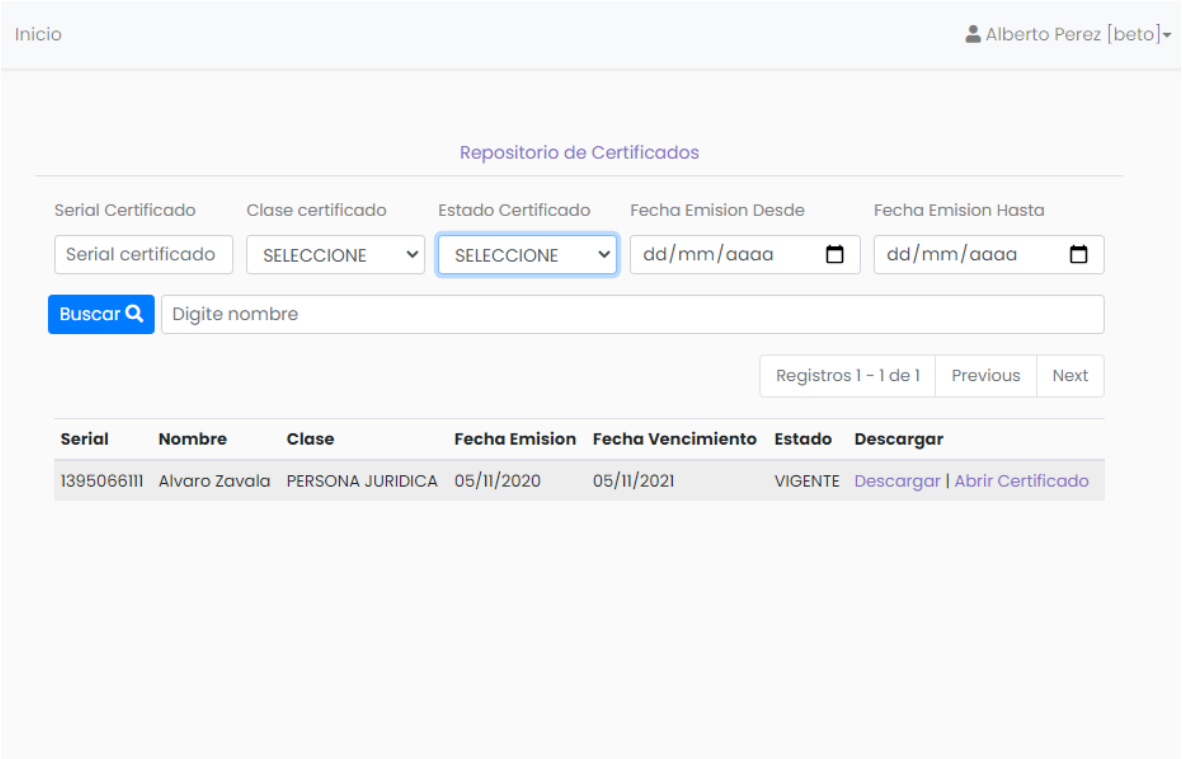

Fig 32. Interfaz para consulta y descarga de certificados.

Descarga de Certificados

La función de descargar certificados está disponible para todos los usuarios a través del mismo repositorio y está protegida por un Captcha para verificar que la obtención del mismo sea realizada en efecto por usuario final.

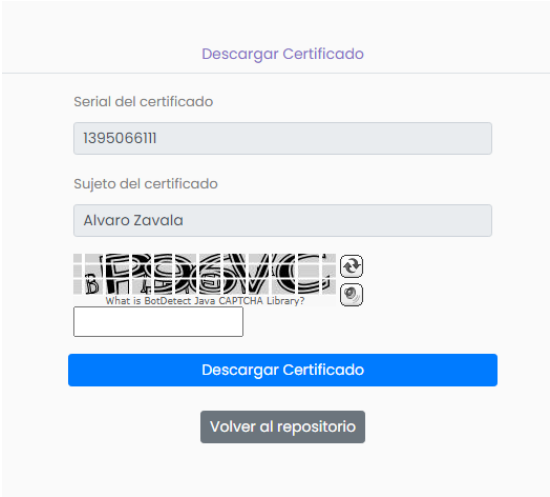

Fig 33. Descarga de certificado con validación de Catpcha

Tarea 4.2. Verificar certificados.

La función para verificar certificados se realiza en tres partes que se describen a continuación:

1. Verificación de fecha de validez: se verifica que el certificado no esté expirado.
2. Verificación de la integridad del certificado: se verifica que la firma de la AC consignada en el certificado, cuya validez garantiza que el certificado no ha sido alterado.
3. Verificación del estado del certificado: se verifica que el certificado este vigente.

A continuación, un ejemplo de su uso:

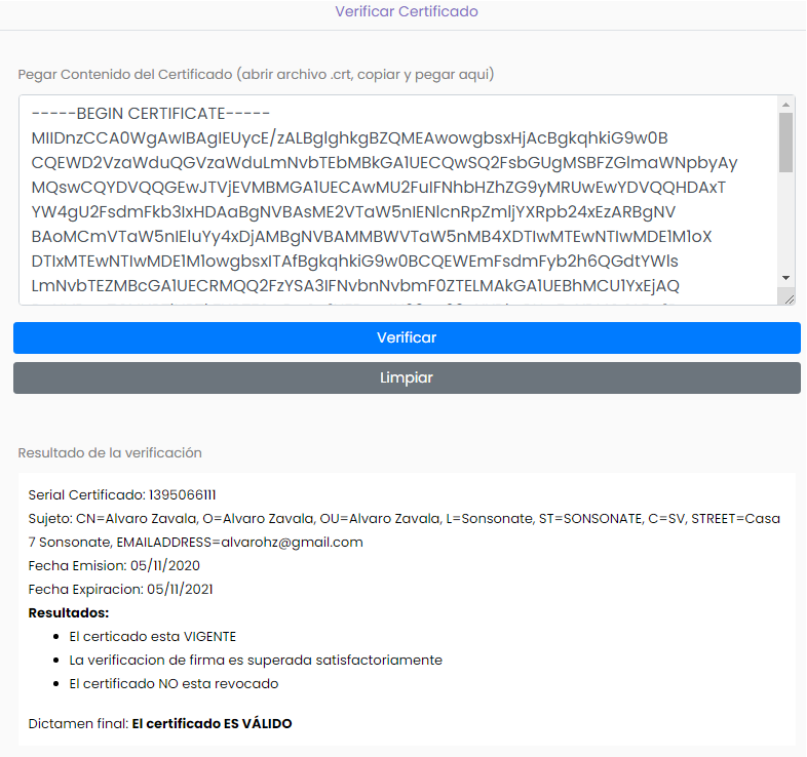

Fig 34. Verificación de la validez del certificado.

Tarea 4.3. Renovar certificados.

Un usuario puede acceder a la opción de *Renovar certificado* una vez el certificado este expirado desde la opción de menú *mis certificados*.

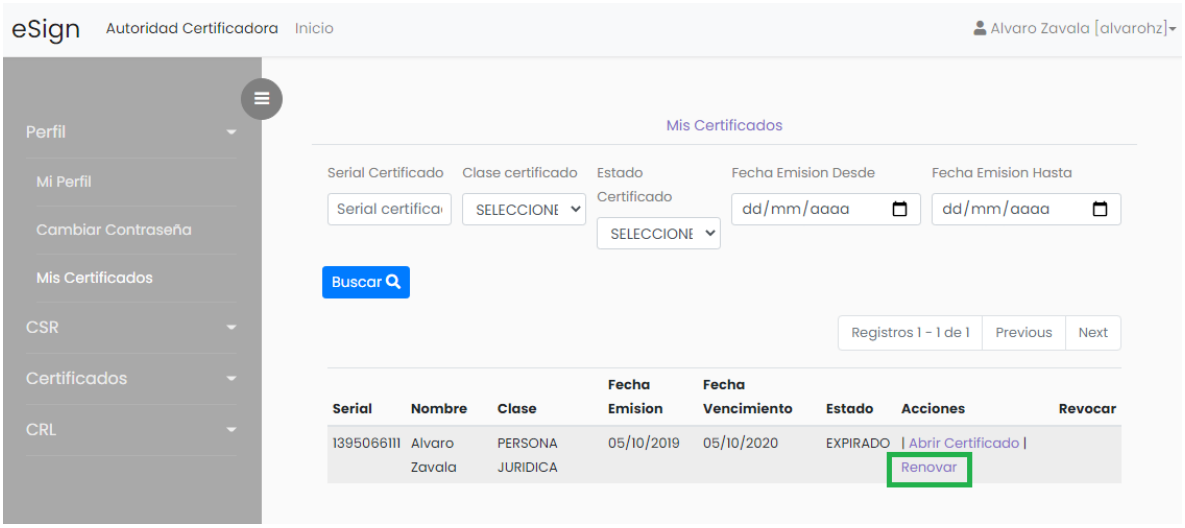

Fig 35. renovación de un certificado.

Luego el proceso pasa a ser similar al de una solicitud nueva, con la diferencia que ya existe información capturada previa del usuario pero que también puede ser actualizada en el proceso.

5.3.2.3 Iteración 3

A continuación, se presenta el Pila de la Iteración *3* que servirá para identificar las tareas en cada requerimiento que lo conforma.

ID	Requerimiento Pila del producto	Tarea de la Iteración
5	Revocación de certificados.	Solicitar revocar un certificado
		Generar CRL
		Descargar CRL (Punto de distribución)
		Revocar, suspender y modificar certificados

Tabla 19. Backlog del Sprint 3.

Tarea 3.1. Solicitar revocar un certificado.

Para revocar un certificado (de acuerdo a requerimientos de la sección 4.4 de este documento) se creó la opción en el menú “Mis Certificados”, hacer click en la fila del certificado en la opción “Solicitar Revocación” y se procede a seleccionar la razón de revocación, que puede ser una de las siguientes:

Razón de revocación (estándar RFC5280)

Razón	Descripción
KeyCompromise (Llave comprometida)	El token o la ubicación del disco donde la llave privada asociada con el certificado se ha visto comprometida y está en posesión de una persona no autorizada. Esto puede incluir el caso en el que se roba una computadora portátil o se pierde una tarjeta inteligente.

CessationOfOperation (Cese de operaciones)	Si una CA se da de baja y ya no se va a utilizar, el certificado de la CA debe revocarse con este código de motivo. No revoque el certificado de la CA si la CA ya no emite nuevos certificados, pero aún publica las CRL para los certificados emitidos actualmente.
CACompromise (CA Comprometida)	La ubicación del token o del disco donde se almacena la llave privada de la CA se ha visto comprometida y está en posesión de una persona no autorizada. Cuando se revoca la llave privada de una CA, todos los certificados emitidos por la CA que están firmados con la clave privada asociada con el certificado revocado se consideran revocados.
AffiliationChanged (Cambio de afiliación)	El usuario ha terminado su relación con la organización indicada en el atributo Nombre Distinguido del certificado. Este código de revocación se usa normalmente cuando un individuo es despedido o ha renunciado a una organización. No tiene que revocar un certificado cuando un usuario cambia de departamento, a menos que su política de seguridad requiera que una CA departamental emita un certificado diferente.
Superseded (Reemplazo o modificación)	Se ha emitido un certificado de reemplazo a un usuario y el motivo no se incluye en los motivos anteriores. Este motivo de revocación se utiliza normalmente cuando falla una tarjeta inteligente, un usuario olvida la contraseña de un token o el usuario ha cambiado su nombre legal.
CertificateHold (Suspendido)	Una revocación temporal que indica que una CA no responderá por un certificado en un momento específico.

	Una vez que se revoca un certificado con un código de motivo CertificateHold, el certificado se puede revocar con otro código de motivo, o anular su revocación y volver a utilizarse. **Nota**: Aunque CertificateHold permite que un certificado "no se revoque", no se recomienda retener un certificado, ya que resulta difícil determinar si un certificado fue válido durante un tiempo específico.
RemoveFromCRL (Remover de la CRL)	Si se revoca un certificado con el código de motivo CertificateHold, es posible "anular" un certificado. El proceso de revocación aún incluye el certificado en la CRL, pero con el código de motivo establecido en RemoveFromCRL. **Nota**: Esto es específico del motivo CertificateHold
Unspecified (Sin especificar)	Es posible revocar un certificado sin proporcionar un código de motivo específico. Si bien es posible revocar un certificado con el código de motivo no especificado, no se recomienda, ya que no proporciona una pista de auditoría de por qué se revoca un certificado.

Tabla 20. Razones de revocación de un certificado.

Para solicitar la revocación en la aplicación se desarrolló la siguiente pantalla:

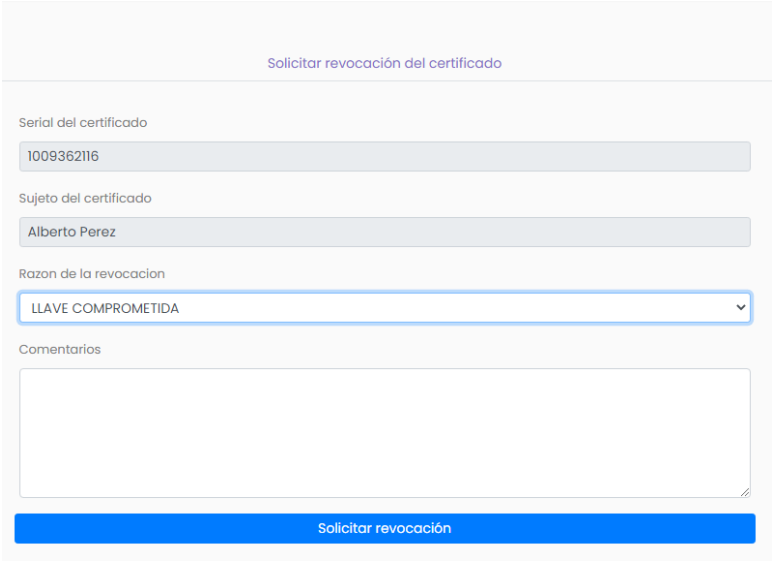

Fig 36. Pantalla para revocación de certificados.

Para usuarios finales las razones disponibles son: Llave comprometida, Modificación, Sin especificar y Cambio de afiliación. Las otras razones se consideran exclusiva decisión de la AC.

Tarea 3.2. Generar CRL.

La CRL (Certificate Revocation List) es un registro utilizado en la infraestructura de clave pública (PKI), para mantener un listado de aquellos certificados (más concretamente sus números de serie) que han sido revocados y, por tanto, ya no son válidos y en los que no se debería confiar. Esta CRL debe estar firmada por la AC y debe existir un punto de distribución para que los usuarios la puedan descargar. A continuación, se muestra la opción que permite generarla.

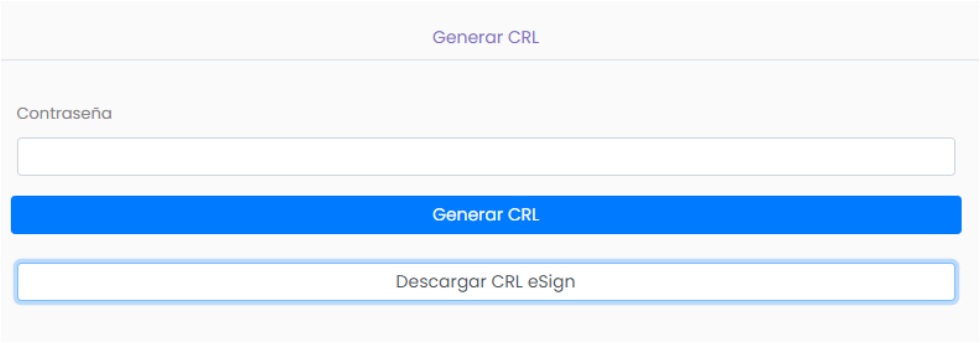

Fig 37. Generación de CRL.

Actualización de la CRL

- Se actualiza cada vez que se revoca un certificado.
- Con la opción de la Fig 37 de manera manual

En ambos casos se solicita la contraseña para firmar la CRL con la llave privada de la AC y se valida la firma en la CRL para verificar que no ha sido alterada y es confiable.

Tarea 3.3. Descargar CRL (Punto de distribución).

Para descargar la CRL cada certificado cuenta con la extensión Distribution Point como se aprecia en Fig 31, este archivo funciona como repositorio de certificados revocados y está a disposición del público para poder generar sus propias verificaciones, además de la que ofrece la aplicación según Fig 34. La opción también está disponible dentro de la aplicación en el apartado de Repositorio CRL como se muestra a continuación.

Fig 38. Consulta y descarga de CRL.

Para el punto de distribución (CDP – CRL Distribution Point) se crea un registro en la tabla *config* en los campos *dircrl, distribution_point y crl_nombre* con lo siguiente.

- dircrl: directorio de archivos donde se guarda la CRL en el servidor
- distribution_point: URL de descarga de la CRL que aparece en los certificados.
- crl_nombre: nombre del archivo que se genera con la CRL.

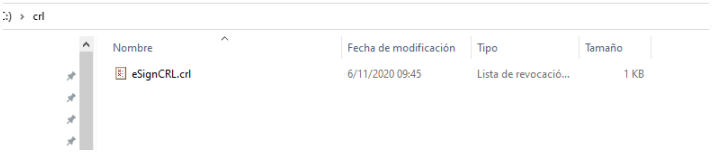

Fig 39. CRL generada en el sistema de archivos con extension .crl.

Tarea 3.4. Revocar, suspender y modificar certificados.

Para Revocar, Suspender o Modificar un certificado se procede de la siguiente forma:

Revocar

Se abre un certificado desde el menú “Repositorio certificados” y para los administradores está disponible la opción revocar certificado como se muestra a continuación.

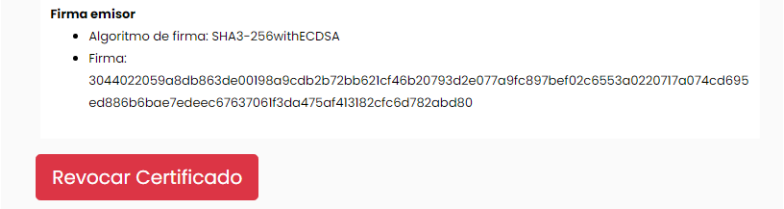

Fig 40. Opción de revocar certificado.

Para los administradores están disponibles todas las razones de Tabla 18. Y de igual manera se solicita la contraseña para firmar la actualización de la CRL.

Suspender

El proceso es idéntico al de una revocación, pero se debe seleccionar la opción CertificateHold (suspender) como razón de revocación.

Modificar

El proceso es idéntico al de una revocación, pero se debe seleccionar la opción Superseded (modificar) como razón de revocación y luego solicitar otro certificado.

5.4 Fase de Cierre

5.4.1 Pruebas

Se realizaron las pruebas para cada entregable de la Iteración. Estas se basaron en la ejecución, revisión y retroalimentación de las funciones previamente establecidas en la Pila del producto.

A continuación, se muestra la matriz de pruebas con la cual se validarán los casos de prueba involucrados en cada Iteración.

ID	Nombre de CP	Objetivo	Acciones	Resultado esperado	Resultado obtenido	Estatus

Tabla 21. Formato matriz de pruebas

Dónde:

- **ID:** Identificador del caso de prueba dentro de la matriz.
- **Nombre de CP:** Nombre del caso de prueba.
- **Objetivo:** Propósito del caso de prueba a realizar.
- **Acciones:** Pasos y datos de entrada necesarios para realizar el caso de prueba.
- **Resultado esperado:** Descripción de lo que debería ver tras haber completado el caso de prueba.
- **Resultado obtenido:** Descripción de lo que se encuentra después de haber realizado los casos de prueba.
- **Estatus:** Indica el resultado de la ejecución del caso de prueba.

Criterio de aceptación o fallo de los elementos

Exitoso	Una prueba de clasifica como exitosa si su estado de anomalía se encuentra en un nivel bajo o en el mejor de los casos no se encuentran anomalías
Fallo	Una prueba de clasifica como fallo si su estado de anomalía se encuentra en un nivel impeditivo, inutilizable o urgente.

Tabla 22. Criterio de aceptación para las pruebas.

5.4.1.1 Pruebas Iteración 1

ID	Nombre de CP	Objetivo	Acciones	Resultado esperado	Resultado obtenido	Estatus
1	Creación y modificación de usuarios y asignación de roles	Verificar registro y modificación y asignación de roles en usuarios	• Registrar usuario • Modificar usuario • Eliminar usuario • Asignar roles	Los cambios se guardan de manera sin errores.	Los cambios fueron guardados sin errores	Exitoso
2	Auto registro de usuarios.	Verificar el auto registro de usuarios	• Llenar formulario de auto registro • Enviar la información.	El usuario activa su cuenta con el link enviado al correo.	El usuario activa su cuenta con el link enviado al correo	Exitoso
3	Autenticación y autorización de usuarios	Validar el inicio de sesión y los permisos asignados	• Digitar credenciales • Presionar botón de iniciar sesión	El usuario aterriza en la pantalla principal y el menú se carga de acuerdo a su rol	El usuario aterriza en la pantalla principal y el menú se carga de acuerdo a su rol	Exitoso
4	Consulta de usuarios	Verificar la búsqueda de usuarios	• Buscar sin filtro • Buscar haciendo uso de filtros	Los usuarios se muestran obedeciendo el criterio de los filtros seleccionados	Los usuarios se muestran obedeciendo el criterio de los filtros seleccionados	Exitoso
5	Generar CSR	Validar la generación de las CSR's	• Completar información para la CSR • Presionar en generar CSR • Descargar llave privada • Copiar CSR	La CSR se genera de acuerdo a la información y parámetros criptográficos seleccionados y se descarga la llave privada y se copia la CSR	La CSR se genera de acuerdo a la información y parámetros criptográficos seleccionados y se descarga la llave privada y se copia la CSR	Exitoso
6	Crear solicitud de creación de certificado	Verificar la creación de solicitudes de certificados	• Completar información de solicitud • Pegar la CSR • Enviar solicitud	La solicitud es aceptada y se envía un correo de notificación al usuario	La solicitud es aceptada y se envía un correo de notificación al usuario	Exitoso
7	Consultar solicitudes CSR	Validar la consulta de solicitudes CSR	• Buscar sin filtro • Buscar haciendo uso de filtros	Las solicitudes se muestran de acuerdo a criterios seleccionados	Las solicitudes se muestran de acuerdo a criterios seleccionados	Exitoso
8	Aprobar y denegar solicitudes	Verificar la aprobación o denegación de certificados	• Abrir certificado • Cambiar el estado ha aprobado o denegado • Si es aprobado, ingresar la contraseña • Enviar datos.	El certificado se genera con la info y los parámetros criptográficos de la CSR y se envía un correo al usuario que incluye el archivo .crt	El certificado se genera con la info y los parámetros criptográficos de la CSR y se envía un correo al usuario que incluye el archivo .crt	Exitoso

Tabla 23. Tabla de pruebas Iteración 1.

5.4.1.2 Pruebas Iteración 2

ID	Nombre de CP	Objetivo	Acciones	Resultado esperado	Resultado obtenido	Estatus
9	Generar y almacenar de manera segura llave privada de la AC.	Verificar la seguridad de la generación y almacenamiento de la llave privada de la AC	• Generar llave privada y certificado • Almacenar llave privada cifrando con AES	La llave privada se almacena cifrada con AES 128 en la base de datos, su certificado y llave pública en claro	La llave privada se almacena cifrada con AES 128 en la base de datos, su certificado y llave pública en claro	Exitoso
10	Descifrar llave privada para firmar certificados	Validar el descifrado de la llave privada de la AC para emitir certificados, revocaciones y generar la CRL	• Acceder a opciones que requieran firma de la AC (aprobar certificados, aprobar revocaciones, generar CRL) • Digitar la contraseña solicitada • Enviar datos	La llave se descifra y se firma de manera correcta	La llave se descifra y se firma de manera correcta	Exitoso
11	Generar el archivo del certificado, almacenarlo en un repositorio y la base de datos	Validar la generación del repositorio de certificados	• Aprobar certificados	Los certificados se almacenan en el sistema de archivos con extensión .crt y se guarda registro en la base de datos	Los certificados se almacenan en el sistema de archivos con extensión .crt y se guarda registro en la base de datos	Exitoso
12	Consultar y descargar certificados	Verificar la consulta y descarga de certificados	• Buscar sin filtro • Buscar con filtro • Presionar en la opción descargar de cada certificado • Digitar Captcha solicitado • Presionar en descargar	Los certificados se muestran de acuerdo al filtro seleccionado y se descargan después de digitar el Captcha correcto	Los certificados se muestran de acuerdo al filtro seleccionado y se descargan después de digitar el Captcha correcto	Exitoso
13	Verificar certificados	Garantizar la verificación de certificados	• Pegar el contenido del certificado • Presionar en verificar	El resultado de la validación del certificado se muestra de forma correcta	El resultado de la validación del certificado se muestra de forma correcta	Exitoso
14	Renovar certificados	Verificar la renovación de certificados	• Entrar a “Mis Certificado” • Presionar en renovar certificado • Completar información para la renovación	La renovación del certificado se procesa y envía un correo de notificación al usuario	La renovación del certificado se procesa y envía un correo de notificación al usuario	Exitoso

Tabla 24. Tabla de pruebas Iteración 2.

5.4.1.3 Pruebas Iteración 3

ID	Nombre de CP	Objetivo	Acciones	Resultado esperado	Resultado obtenido	Estatus
15	Solicitar revocar un certificado	Validar la solicitud de revocaciones	• Entrar a “mis certificados” • Presionar en “Solicitar revocación” • Seleccionar razón • Enviar datos	La solicitud se envía y es consultable por el usuario	La solicitud se envía y es consultable por el usuario	Exitoso
16	Generar CRL	Verificar la generación de la CRL	• Entrar a “Generar CRL” • Escribir contraseña para descifrar llave privada de la AC • Presionar en generar CRL	La CRL se genera en el directorio establecido	La CRL se genera en el directorio establecido	Exitoso
17	Descargar CRL (Punto de distribución)	Verificar el punto de distribución de la CRL	• Se abre un certificado • Se copia la URL en el campo Punto de Distribución • Se pega en el navegador y presiona enter	La CRL inicia su descarga	La CRL inicia su descarga	Exitoso
18	Revocar, suspender y modificar certificados	Validar la revocación, modificación y suspensión de certificados	• Abrir un certificado • Presionar en revocar certificado • Seleccionar la razón correspondiente (suspensión, modificación o revocación) • Escribir contraseña solicitada para llave privada de la AC	El certificado es revocado y añadido al repositorio CRL	El certificado es revocado y añadido al repositorio CRL	Exitoso

Tabla 25. Tabla de pruebas Iteración 3.

5.4.2 Revisión de Iteraciones

Los requerimientos o requisitos completados (“sprint review”), en SCRUM se realiza cuando el equipo presenta al cliente los requisitos completados en la iteración en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo, haciendo un recorrido por ellos lo más real y cercano posible al objetivo que se pretende cubrir.

El equipo de trabajo se apoyó de las pruebas realizadas para garantizar el funcionamiento del software y la seguridad de las operaciones criptográficas. Siendo todos satisfactorios.

En función de lo antes expuesto y habiendo realizado pruebas a los elementos de la Pila del producto, así como sus tareas de la Iteración.

Se ha verificado de manera objetiva el cumplimiento del desarrollo de los requisitos planteados, cumpliendo con las expectativas en el marco del diseño de la propuesta de software.

CAPÍTULO VI

Discusión de seguridad y aplicabilidad

Capítulo VI. Discusión de seguridad y aplicabilidad

En el presente capítulo se discutirán aspectos de seguridad y aplicabilidad para la aplicación de gestión de certificados basada en la Ley de Firma Electrónica de El Salvador.

6.1 Discusión de seguridad

6.1.1 Posibles ataques

Los posibles ataques se han considerado tomando en cuenta los que se mencionan en (Rivera Zamarripa, y otros 2019) y otras particularidades del caso específico de la propuesta.

Falsificación de certificados

La falsificación de certificados es la generación de certificados que se consideran ilegales y podrían dar lugar a fraudes, algunas razones por las cuales se podrían falsificar son: Llave privada de la AC comprometida, Cálculo por lotes de GCD (Batch GCD), Funciones hash vulnerables a colisiones, etc.

Descarga masiva de certificados

Los certificados digitales que vinculan a una entidad final (usuario) con su llave pública son de acceso público, por lo tanto, descargar de manera masiva (miles o millones) certificados es posible, aunque no debería ser tan fácil. La descarga masiva de certificados daría lugar a realizar un ataque de cálculo por lotes GCD en certificados que utilizan el algoritmo RSA, encontrar la llave privada correspondiente a un certificado sería viable de encontrarse vulnerabilidades en el generador de números pseudoaleatorios. Un ejemplo de este ataque fue realizado al sistema de certificado digitales de México como se muestra en (Rivera Zamarripa, y otros 2019).

Ataques a las contraseñas de usuarios

Los ataques a las contraseñas de usuario están basados en ataques de fuerza bruta (de diccionario o tablas Rainbow) que se pueden llevar a cabo en línea o fuera de línea. En línea implicaría que la aplicación permita la explotación de intentos de usuarios y contraseñas en el inicio de sesión. En cambio, los ataques fuera de línea se realizan cuando es posible obtener la base de datos donde se encuentran almacenadas las contraseñas de los usuarios. El éxito de estos ataques depende de varios factores como: longitud de la contraseña, algoritmo hash de almacenamiento empleado, restricciones sobre los intentos, etc.

Las restricciones sobre la cantidad de intentos también podrían dar lugar a infligir un ataque de denegación de servicio al hacer muchos intentos incorrectos. Las contraseñas deben ser lo suficientemente complejas como para que la limitación no se produzca después de un número modesto de intentos erróneos, pero sí antes de que exista una posibilidad significativa de una suposición exitosa.

Colisiones en algoritmos hash

Una colisión de hash es una condición donde dos entradas distintas a una función hash producen una misma salida. Normalmente esto se debe a la longitud del espacio de salidas de la función, que por ejemplo en el caso de MD5 y SHA1 son 128 y 160 bits respectivamente. El algoritmo MD5 ha sido descartado desde hace mucho tiempo, y así mismo el SHA1 como ha sido documentado por el NIST (NIST, Recommendation for key management 2016). Un ejemplo de colisionador SHA1 puede ser encontrado en (Ellingsen 2017). Las colisiones en los algoritmos hash podrían dar lugar a falsificación de certificados como se demuestra en (Marc, y otros 2009). En el caso de esquemas de firma digital podría darse alteración y falsificación de documentos.

Problema del logaritmo discreto

El problema matemático de la criptografía de curva elíptica es el problema del logaritmo discreto. La seguridad de la criptografía de curva elíptica depende de la capacidad de calcular una multiplicación de puntos y de la incapacidad de calcular

el multiplicando dados los puntos originales y del producto. El tamaño de la llave determina la dificultad del problema. Por lo tanto, si la llave escogida es demasiado pequeña podría dar lugar a que el problema pudiera ser resuelto o incluso emplear ataques de fuerza bruta sobre ella. Lo ideal es hacer estos ataques inviables seleccionando tamaños de llaves adecuados como los que se señalan en (NIST, Recommendation for key management 2016).

Ataques de spam o phishing

El spam es correo electrónico no solicitado que se envía a un gran número de destinatarios con fines publicitarios o comerciales. Por otro lado, el phishing es un conjunto de técnicas que persiguen el engaño a una víctima, que podría ser a través de correo falsos, haciéndose pasar por una persona, empresa o servicio de confianza, para manipularla y hacer que realice acciones o brinde información que no debería (contraseñas, por ejemplo). Dado que los certificados dentro de sus campos y como lo dicta La Ley de Firma Electrónica cuenta con la dirección electrónica de los usuarios, esto podría dar lugar a tales ataques.

6.1.2 Aspectos de seguridad implementados

A continuación, se describen los aspectos de seguridad implementados en la propuesta técnica para mitigar los riesgos sobre los ataques planteados en la sección anterior.

Protección de contraseñas de usuario

Para la protección de contraseñas de usuario se toma en cuenta como línea base la guía en (NIST, Digital Identity Guidelines 2017), en su última actualización del 2020 que contradice muchas de las practicas que se utilizan actualmente con respecto a la definición de políticas de contraseñas, entre las cuales han sido implementadas la siguientes.

- Mínimo de 8 caracteres (en la propuesta son 12 mínimo) cuando un humano lo establece.
- Admite al menos 64 caracteres de longitud máxima.

- Todos los caracteres ASCII (incluido el espacio) deben ser compatibles
- El truncamiento del secreto (contraseña) no se realizará cuando se procese.
- Limitar los intentos de autenticación fallidos consecutivos en una sola cuenta a no más de 100 (en la propuesta 20) antes de bloqueo (evitar en la medida de lo posible la denegación de servicio)
- Sin requisitos de composición para complejidad
- Sin período de caducidad de la contraseña

Otras recomendaciones también son:

- Sin pistas de contraseña
- Sin autenticación basada en el conocimiento (por ejemplo, ¿quién era tu mejor amigo en la escuela secundaria?)
- Sin SMS para 2FA (usar una contraseña de un solo uso de una aplicación como Google Authenticator)

Nota: Esta última no está implementada.

Protección para ataques en línea a las contraseñas

Los ataques en línea (online) son prevenidos en cierta medida por los lineamientos anteriores, adicionalmente a ellos se han implementado a partir del quinto intento la solución de un Captcha, mecanismo con el cual se verifica que los intentos son llevados a cabo por una persona y no por un programa o hardware diseñado para realizar ataques sobre la plataforma en línea.

Protección para ataques fuera de línea a las contraseñas

Para los ataques fuera de línea (offline) se siguen las siguientes recomendaciones dadas por (NIST, Digital Identity Guidelines 2017).

- Los secretos (contraseñas) deben contener una sal (salt) y usando una función de hash adecuada (SHA-256 en la propuesta).
- Función de derivación de claves basada en contraseña 2 (PBKDF2)
- La sal debe tener al menos 32 bits de longitud y se elegirá arbitrariamente para minimizar las colisiones de valores de sal entre los hashes almacenados

Tamaños de llave seguros

Los algoritmos criptográficos basan su seguridad en dos factores importantes, la calidad del algoritmo (descrito más adelante) y el tamaño de la llave. La solución de los problemas matemáticos sobre los que están construidos es intratable siempre y cuando se seleccione un tamaño de llave adecuado. De esta manera los ataques de fuerza bruta o al problema matemático se vuelven imprácticos. Para la propuesta técnica se emplean los siguientes tamaños de llaves considerando las recomendaciones del NIST en (NIST, Recommendation for key management 2016):

Criptografía de curvas elípticas (en bits): 256, 384 y 512

Algoritmos de firma seguros

Existen criptosistemas que se consideran seguros. En la práctica, estos se utilizan ampliamente y la mayoría de especialistas competentes los consideran irrompibles, aunque el problema matemático aún se encuentre abierto. El problema del logaritmo discreto (Criptografía de curvas elípticas) y la factorización de números enteros (RSA) son algunos ejemplos. Para la propuesta actual se emplean los siguientes algoritmos de los aprobados por el NIST en su estándar (NIST, Recommendation for key management 2016).

- **ECDSA** Algoritmo de Firma Digital de Curva Elíptica, es una modificación del algoritmo DSA que emplea operaciones sobre puntos de curvas elípticas en lugar de las exponenciaciones que usa DSA.

  Aunque comercialmente el RSA es más utilizado la criptografía de curvas elípticas ofrece las siguientes ventajas con respecto a RSA, razón por la cual se decide usar ECDSA.

  Forward Secrecy (Secreto Perfecto): es la propiedad de los sistemas criptográficos que garantiza que el descubrimiento de las llaves utilizadas actualmente no compromete la seguridad de las llaves usadas con anterioridad (no las revela).

Tamaños de llave más pequeños: las llaves que emplea son en comparación con RSA las siguientes en bits

RSA	ECC (Elliptic Curve Cryptography)
2048	256
3072	384
7680	512

Tabla 26. Comparativa de llaves RSA y ECC

Como se aprecia en la tabla anterior, los tamaños de llave son comparativamente mucho más pequeñas en ECC de lo que son en RSA, por lo tanto, el manejo eficiente de llaves en cuanto a recursos se logra con ECC.

- **SHA3** de 256, 384 y 512 bits para huellas digitales, significa que las huellas digitales generadas para la firma en certificados pueden ser de las longitudes en bits mencionadas.
- **SHA256** para almacenamiento seguro de contraseñas de usuario.

Como conclusión los algoritmos criptográficos utilizados en la propuesta hacen impráctico cualquier tipo de ataque a los que se les quiera someter.

Captcha en descarga de certificados

Como medida para evitar la descarga masiva de certificados y prevenir ataques como Batch GCD se ha implementado la validación de un Captcha cada vez que un usuario desea descargar un certificado. Que, aunque sean de consulta pública debe existir este mecanismo de verificación para prevenir que se conecten bots u otros métodos y puedan descargar indiscriminadamente de forma masiva certificados, que podrían ser utilizados, además, para los ataques de Spam y Phishing.

Almacenamiento seguro de llave privada de la AC

Para mitigar el riesgo de que la llave privada de la AC se vea comprometida se implementa el protocolo de almacenamiento seguro de llaves con el cifrador por bloques AES-128. De esta manera la llave se almacena cifrada en la base de datos

y solo las personas autorizadas para su uso puedan hacerlo. El protocolo utilizado se describe en los siguientes diagramas de bloques.

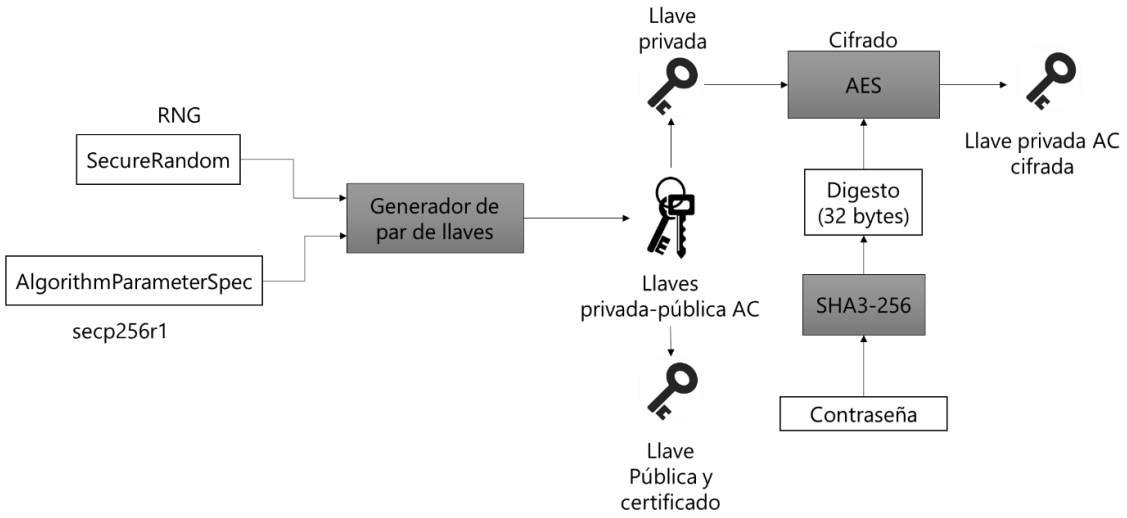

Fig 41. Proceso de generación y almacenamiento seguro de llave privada.

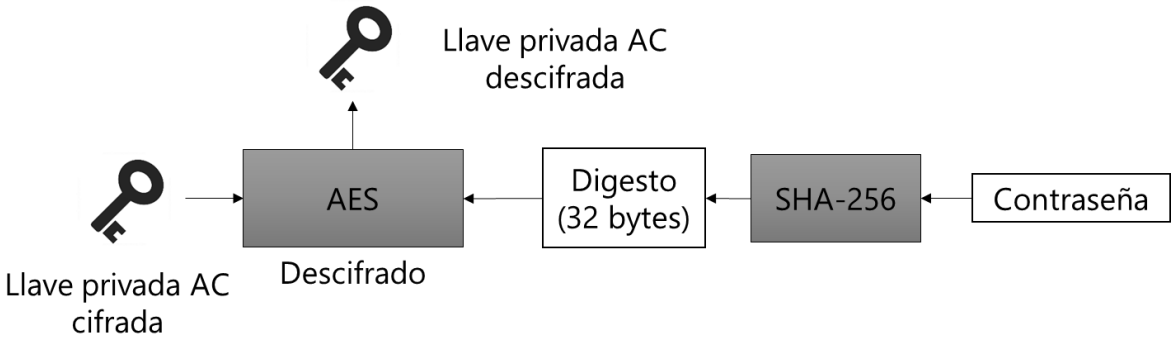

Fig 42. Descifrado de la llave privada

6.1.3 Aspectos de seguridad por implementar a futuro

Ataques de Spam y Pishing

Los ataques de spam y phishing requieren un estudio adicional por las implicaciones que tiene el hecho, que cada certificado es de acceso público y brinda acceso a la dirección de correo electrónico. Para posteriores iteraciones del producto se pueden analizar alternativas para mitigar este riesgo.

Seguridad doble factor

La seguridad de doble factor puede ser una alternativa adicional de seguridad que se podría implementar, con algún tipo de dispositivo de hardware o en su implementación más básica con un código de un solo uso enviado al correo electrónico.

Realizar ataque Batch GCD

Aunque el riesgo de los ataques Batch GCD ha sido mitigado, puede realizarse un test para probar si existe vulnerabilidad en el generador de números pseudoaleatorios y debe ser remplazado en caso de que el ataque sea exitoso.

Separar la administración de la plataforma

Actualmente se contempla una separación de roles y privilegios en la misma versión del producto como se muestra en Tabla 17. Como medida de protección adicional podría considerarse separar la administración del software (roles ADMIN, SA) en otra versión del producto, separada incluso físicamente de la versión que usarían los usuarios finales de los certificados.

6.2 Discusión de aplicabilidad

La aplicabilidad del sistema por diseño está destinado a obedecer los criterios que la Ley de Firma Electrónica de El Salvador expresa en cuanto a los procesos que debe seguir un usuario final y una AC, pero con el propósito que la propuesta pueda ampliar su alcance se hace el siguiente análisis.

Implementar certificados RSA

Como se mencionó anteriormente la propuesta utiliza criptografía de curvas elípticas, pero puede ser incluido el algoritmo de RSA para hacer más comercial su uso, debido a que RSA actualmente es la opción más utilizada en el mercado.

Licencias de las tecnologías de desarrollo.

- Actualmente se utiliza la versión de MySQL Server Community pero puede ser migrada a una versión Enterprise dependiendo el ámbito (público o privado) que lo requiera.
- El lenguaje de programación usado es Java, que a diferencia de la base de datos no tiene licenciamiento propietario. A pesar de lo anterior no existe razón para pensar que esto podría ser una limitación. En todo caso se puede hablar del .NET Framework que está pasando a ser open source junto con su IDE Visual Studio.

Compatibilidad con navegadores web

Los certificados emitidos son compatibles con los navegadores web (formato: Codificación binaria DER .cer). Por lo tanto, podrían utilizarse para la protección se servidores web. A continuación, se muestra pruebas realizadas con el servidor apache 2.4.46. en navegador Safari y Chrome.

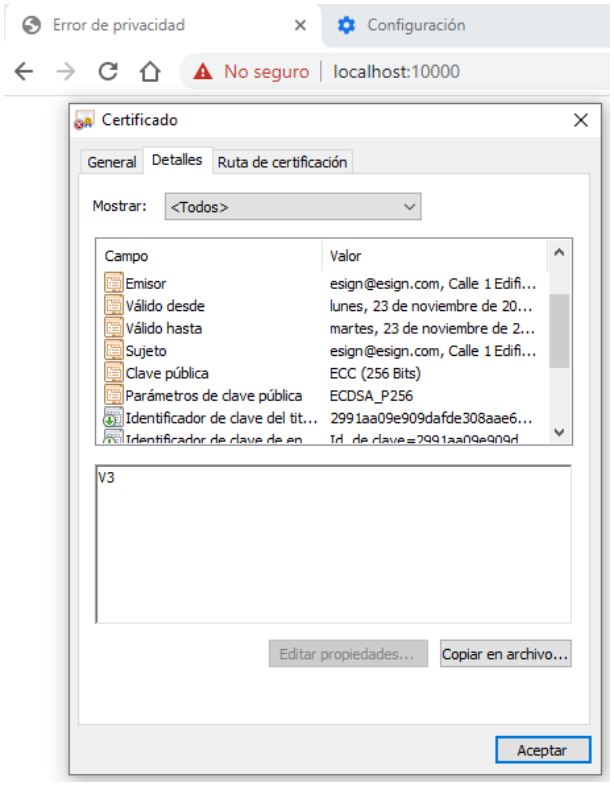

Fig 43. Certificado visto desde Chrome

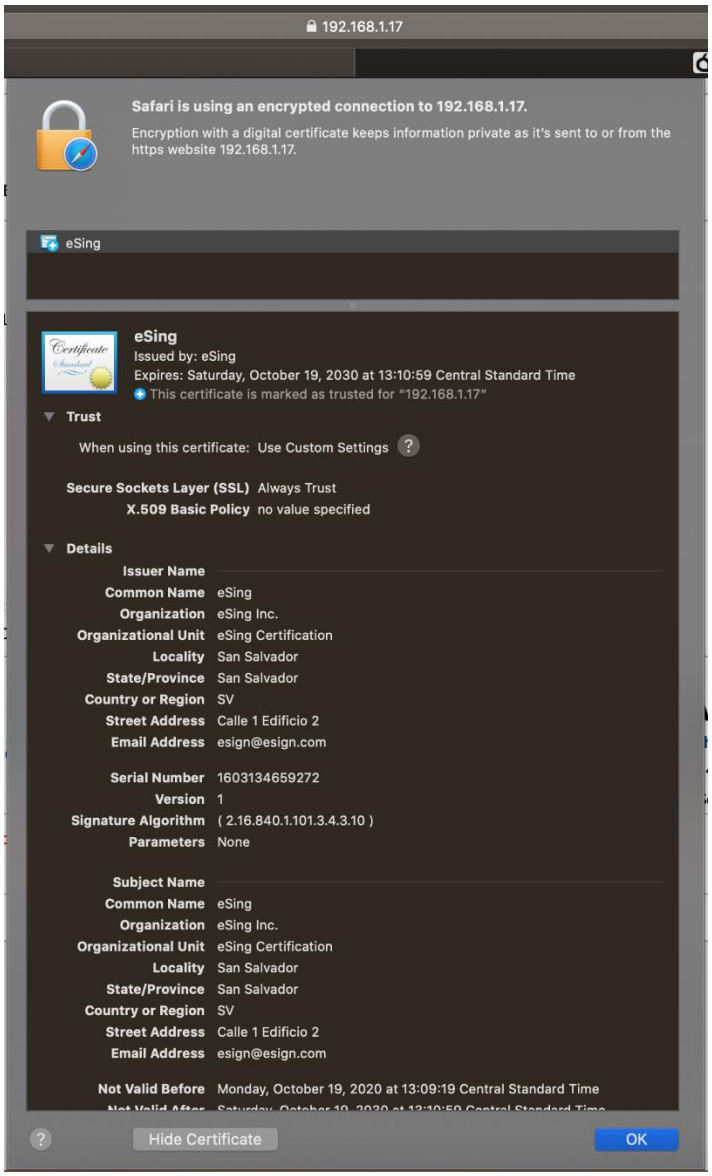

Fig 44. Certificado visto desde Safari

Cabe mencionar que la configuración del servidor apache en versiones más antiguas como la 2.4.2 genera el siguiente error al introducir el protocolo TLSv1.3 para soportar algoritmos con secreto perfecto como el ECC.

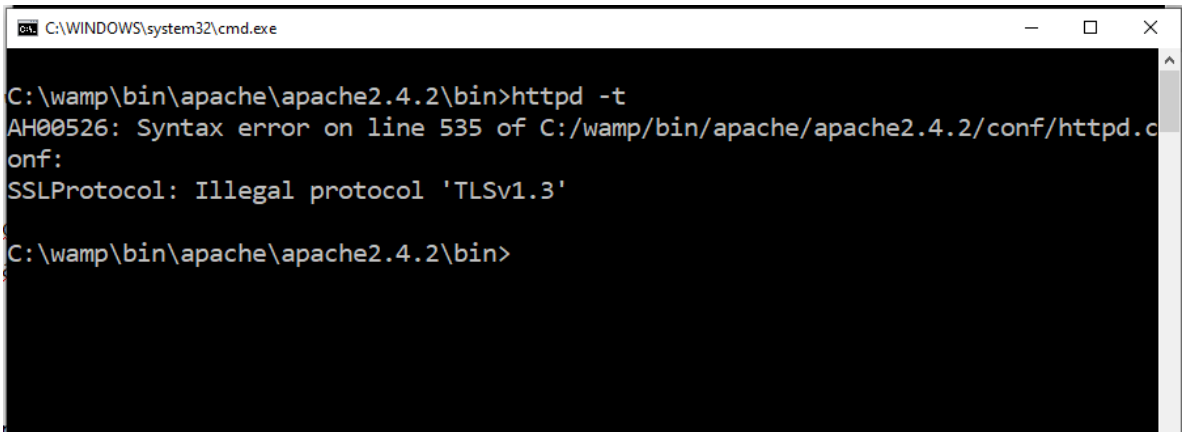

Fig 45. Error incorporando TLSv1.3 a Apache 2.4.2

Añadirse a la cadena de confianza de otro proveedor

Como se aprecia en Fig 43. Los certificados emitidos no son reconocidos por los navegadores como seguros, debido a que no están emitidos por un ancla de confianza reconocida (DigiCert, por ejemplo). Para ofrecer esta funcionalidad existe la alternativa de adquirir un certificado para una AC de parte de un proveedor autorizado y unirnos a la cadena de confianza para validar los certificados emitidos y que estos sean validados por los navegadores como seguros.

Proceso completamente en línea

El proceso diseñado aun requiere de validaciones presenciales de parte de los usuarios ante la AC, pues, el fin principal de los certificados es la firma electrónica, considerando que es posible emitir otro tipo de certificados con seguridad mínima para distintos usos, sería posible manejar el proceso completamente en línea, siempre realizando verificaciones de que el usuario tiene el absoluto control de la cuenta a través del correo vinculado en el sistema de gestión de certificado. Por ejemplo, para generar un certificado recibiría a su correo un enlace con un token, sobre el cual debería hacer click para obtener el certificado.

CAPÍTULO VII

Conclusiones y recomendaciones

Capítulo VII. Conclusiones y recomendaciones

7.1 Conclusiones

- La aplicación desarrollada permite gestionar el ciclo de vida de los certificados de entidades finales (EE) desde la creación de la CSR hasta la revocación o expiración del certificado garantizando la seguridad de las operaciones criptográficas involucradas en el proceso con algoritmos y estándares probados y sólidos en cuanto a seguridad.
- El producto obtenido puede servir como base fundamental para futuras implementaciones de las autoridades certificadoras del país, esto dado que se basa en los requerimientos explicados en la Ley de Firma Electrónica de El Salvador y los estándares aplicables como los emitidos por el NIST y el IETF con sus RFC's.
- El desarrollo de una Infraestructura de Llave Pública (PKI) que soporte a una Autoridad Certificadora (AC) es un desafío que implica el conocimiento de algoritmos de cifrado, de lenguajes de programación para su implementación.
- El algoritmo de curvas elípticas empleado representa una buena opción para el almacenamiento de llaves en dispositivos con poco almacenamiento o en sistemas donde se requieren almacenar grandes cantidades de llaves públicas, dado que por su tamaño ocupan menos espacio en disco y ofrece operaciones más eficientes.
- Los algoritmos utilizados, tanto simétricos, asimétricos o hash, responden a un tamaño mínimo de llave seguro contra ataques de fuerza bruta (128 bits), arma preferida de un hacker, y aún se desconocen vulnerabilidades en ellos.
- La gestión del ciclo de vida de un certificado es un proceso complejo en su interior, pero que, a través de la aplicación propuesta, con interfaz amigable y las consideraciones de usabilidad tomadas en cuenta se hace un proceso más sencillo para cualquier entidad final o entidad certificadora.

7.2 Recomendaciones

- La propuesta no contempla la gestión de pagos o vínculo con pasarelas de pagos, se recomienda incluirlo en futuros trabajos de investigación.
- La razón de revocación CACompromise (AC Comprometida) involucra revocar todos los certificados emitidos por esa AC, ya que ha dejado de ser confiable. Dicho proceso está más allá del alcance de la actual propuesta por lo que se recomienda hacer revisión de las implicaciones que esto conlleva y si acaso le compete a la misma AC.
- Contemplar el uso de múltiples llaves privadas para un AC, actualmente se da soporte a una única llave privada.
- El cese de operaciones de una AC, está contemplado en La ley de Firma Electrónica, pero habría que ampliar sobre las implicaciones y complejidad del proceso que también se ha dejado fuera del alcance de la propuesta actual, pues va más allá del ciclo de vida de los certificados de las entidades finales (EE).

Bibliografía

A. Menezes, P. van Oorschot, and S. Vanstone. *Handbook of Applied Cryptography.* CRC Press, 1996.

BCR. *www.bcr.gob.sv.* 2019. https://www.bcr.gob.sv/bcrsite/uploaded/content/category/452132105.pdf (accessed 02 2020).

Cerén, Salvador Sánchez. *Reglamento de la Ley de Firma Electrónica.* 2016. http://www.transparencia.gob.sv/institutions /minec/documents/127305/download.

—. "Ley de Firma Electrónica." 2015. http://www.transparencia.gob.sv/institutions/dc/documents/139 830/download.

Ellingsen, Erling. *SHA1 collider.* 2017. https://alf.nu/SHA1 (accessed 10 10, 2020).

IETF. "RFC 3280." 2018. https://www.ietf.org/rfc/rfc3280.txt.

—. "RFC 3820." 2018. https://www.ietf.org/rfc/rfc3820.txt.

—. "RFC 4158." 2018. https://tools.ietf.org/html/rfc4158.

—. *RFC 5280.* 2018.

ISACA. *Cybersecurity Fundamentals Study Guide.* Rolling Meadows, 2017.

Marc, Stevens, et al. "Short chosen-prefix collisions for MD5 and the creation of a rogue CA certificate." 2009.

MINEC. *http://infotrade.minec.gob.sv/.* 2017. http://infotrade.minec.gob.sv/blog/minec-instala-comite-tecnico-consultivo-firma-electronica/ (accessed 02 2020).

Nakamoto, Satoshi. "Bitcoin." 2010. https://en.bitcoin.it/wiki/Secp256k1 (accessed 02 2020).

NIST. "Advanced Encryption Standard." 2001. https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf (accessed 02 2020).

—. *Digital Identity Guidelines.* 2017. https://pages.nist.gov/800-63-3/sp800-63b.html (accessed 10 15, 2020).

—. "Digital Signature Standard (DSS)." 2013. https://nvlpubs .nist.gov/nistpubs/fips/nist.fips.186-4.pdf.

—. "Public Key Infrastructure." 2001. https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-32.pdf (accessed 02 2020).

—. "Recommendation for key management." 2016. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r4.pdf.

—. "Secure Hash Standards (SHS)." 2018. https:// nvlpubs.nist.gov /nistpubs /FIPS/NIST.FIPS.180-4.pdf.

—. "SHA3 standard." 2015. https://nvlpubs.nist.gov/nistpubs /FIPS/NIST.FIPS.202.pdf.

Rivera Zamarripa, Luis, Lil Maria Rodríguez, Miguel Ángel León Chávez, Nareli Cruz Cortés, and Francisco Rodríguez Henríquez. "Security Analysis of the Mexican Fiscal Digital Certificate System." *Computación y Sistemas*, 2019.

scrum.org. *scrum.org.* 2020. https://www.scrum.org/resources/what-is-scrum (accessed 08 01, 2020).

Stalling, William. *Cryptography and Network Security.* Prentice Hall, 2011.

Stinson, Douglas. *Cryptography: Theory and Practice.* CRC Press, 1995.

STPP. *http://secretariatecnica.egob.sv/.* 2017. http://secretariatecnica.egob.sv/wp-content/uploads/2017/07/FOROEGOB-04.Avances-Firma-Electronica-El-Salvador.pdf (accessed 02 2020).

Anexos

Anexo 1. Cronograma

Objetivo	Actividad	Fecha inicio	Fecha fin	Indicador de Logro
Establecer los requerimientos de la aplicación de gestión de certificados	Definir procesos de una AC	17/02/20	01/03/20	Diagrama de procesos
	Establecer requerimientos funcionales y no funcionales	02/03/20	17/04/20	Lista de requerimientos
Definir y aplicar los algoritmos criptográficos con las garantías de seguridad en la ejecución de las operaciones	Definir algoritmos criptográficos y tamaños de llave	18/04/18	02/06/20	Lista de algoritmos y sus llaves
	Codificar clases para generar una librería	03/06/20	03/08/20	Librería para la aplicación
Crear módulos para la expedición y revocación de certificados	Construir aplicación web para emisión y revocación de certificados	04/08/20	04/11/20	Aplicación web funcional
	Realizar pruebas al software	05/11/20	20/11/20	Resultados de las pruebas

Duración total: 9 meses

Anexo 2. Hoja de anotaciones para especificaciones técnicas

Especificaciones técnicas certificado + firma digital

Especificación técnica/requerimiento	Descripción

Anexo 3. Cuadro comparativo de tamaños de llave

Security Strength	Symmetric key Algorithms	FFC (e.g., DSA, D-H)	IFC (e.g., RSA)	ECC (e.g., ECDSA)

Referencias

Anaranjado: No recomendado

Verde: Recomendado

Amarillo: No estandarizado por NIST

Anexo 4. Hoja de anotaciones para análisis de requerimientos

Tipo: Seguridad (S), Funcional (F), Usabilidad (U), No Funcional (N)

Aplicación	Requerimiento	Tipo

Anexo 5. Hoja de análisis para procesos de una AC

No.	Proceso	Análisis	Comentario