OIU.151
L19
I

GOBIERNO DE MENDOZA

SECRETARÍA ADMINISTRATIVA LEGAL Y TÉCNICA

UNIDAD DE REFORMA DEL ESTADO

44706

firma Digital

CONSEJO FEDERAL DE INVERSIONES
BIBLIOTE
Dr Manuel

Primer Informe Parcial

CONSEJO FEDERAL DE INVERSIONES

CONSULTOR: LIC. PABLO GUILLERMO LIOY

ÍNDICE

I. INTRODUCCIÓN

Se presentan a continuación, como Primer Informe de Etapa, las actividades y tareas desarrolladas en el marco de planificación del proyecto de Firma Digital Mendoza.

Actividad	Estado
1. ***Implementación de experiencia piloto de Sitio Seguro:***	Concluida
Tareas	
• Relevamiento del circuito operativo de carga de información	Concluida
• Explicitación de la necesidad puntual	Concluida
• Determinación de mejoras	
• Desarrollo de documentación explicativa de sitio seguro para ser incluida (como servicio de información al ciudadano) en el sitio de la Guía de Trámite	Concluida
•Determinaciones sobre la provisión de certificados	Concluida
• Capacitación de los referentes de la guía	Concluida
• Instalación de protocolos y configuración de los servidores web de la Guía de Trámite	Concluida
• Gestión/emisión de certificados de servidor y de usuarios finales para el sitio de la Guía de Trámite, para sus interfaces de administración y carga y para los referentes de la misma.	Concluida

• Desarrollo un plan de pruebas e instrumentación de las pruebas previstas	Concluida
• Implementación efectiva de sitio seguro	Concluida
• Evaluación de Resultados	Concluida
Actividad	**Estado**
2. Implementación de un prototipo de PKI:	En desarrollo
Tareas	
• Evaluación de herramientas de libre distribución para el desarrollo o implementación de aplicaciones PKI	Concluida
• Expicitación del modelo PKI para la constitución de una CA con una RA de pequeña escala para la provincia de Mendoza	Concluida
• Desarrollo de un Prototipo del diseño preliminar haciendo uso de la tecnología seleccionada. El prototipo debe ser capaz de realizar funciones básicas de una CA y de una RA: Emisión de certificados, gestión del CVS de certificados, gestión de CRL, etc.; bajo las condiciones de interoperabilidad, seguridad y escalabilidad pre-establecidas en el Estudio de Factibilidad para una PKI de pequeña escala.	En desarrollo

PRIMER INFORME DE AVANCE: la actividad 1 y el avance de la 2 se presentará a los dos meses de iniciadas las tareas.

II. IMPLEMENTACIÓN DE EXPERIENCIA PILOTO SITIO SEGURO

Siguiendo la tendencia de generar interrelaciones entre los proyectos y aprovechar la sinergia que ellas aportan, el equipo de firma digital ha desarrollado el sistema de Sitio Seguro para el esquema de carga de la Guía de Trámites.

Hablamos de un "Sitio Seguro" cuando nos referimos a un lugar virtual confiable en Internet, perteneciente a una empresa u organización que lo mantiene en línea por medio de un servidor de www (World Wide Web).

Cuando una persona se conecta a un sitio seguro, el servidor presenta un certificado emitido y firmado por la Entidad Emisora de Certificados.

Los programas habitualmente utilizados para navegar por Internet (Browser o Navegador) deben estar configurados para aceptar certificados, que garantizan la confiabilidad del sitio, los que son emitidos por la Entidad Emisora de Certificados. Además, existe la posibilidad de ir más allá y asegurar la identidad de los usuarios del sistema utilizando la misma tecnología de certificación digital, es decir, garantizando al servidor que la persona que accede es aquella a la cual este le ha dado privilegios de acceso y si así lo establecen previamente podría ingresar la base de datos y agregar o modificar información sensible con total seguridad de que la única persona que pudo hacerlo es la titular del certificado digital.

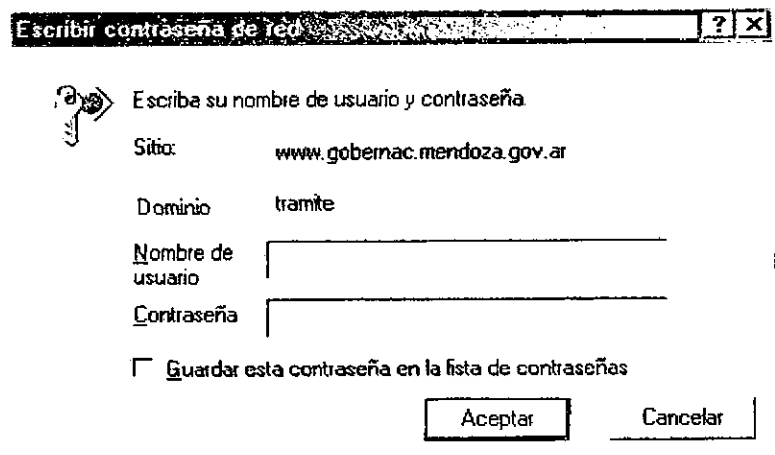
Iremos desarrollando detalladamente estos conceptos a lo largo del Informe.

A) Relevamiento del Circuito operativo de carga

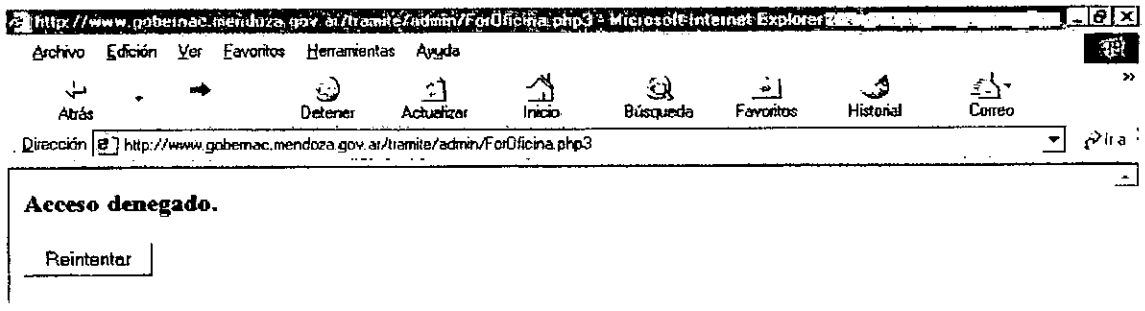
Hemos relevado aquí el procedimiento de carga, detallando específicamente los pasos en los que el sistema de Sitio Seguro con autenticación de usuarios concretamente va a actuar, es decir, se evitan los pasos habituales de la carga propiamente dicha y se hace foco en las instancias de acceso relevantes a los efectos de mejorar la seguridad del sistema.

Descripción del procedimiento actual:

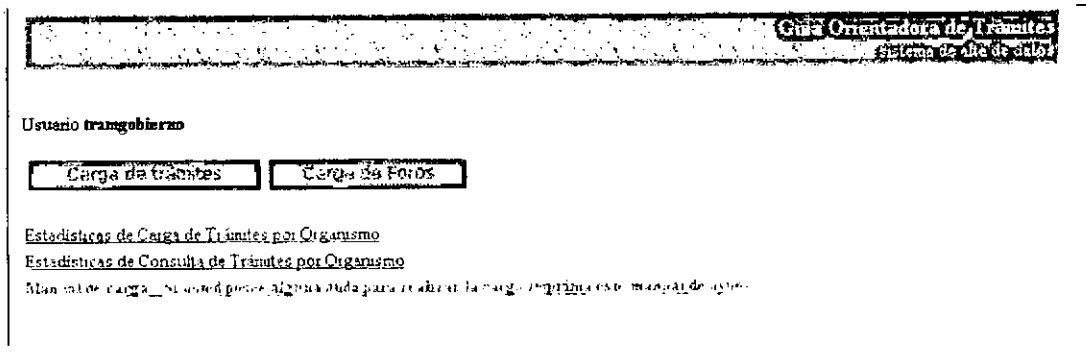
1. **Referente:** Para comenzar la carga de Trámites debe ingresar a: www.tramite.mendoza.gov.ar/admin/usuarios.php3 y en la ventana "Ingresar al Sistema", debe colocar el nombre de usuario **y la** contraseña **asignada.**



2. **Sistema**: si el usuario o la clave no son correctos el sistema lo indicará mediante una página, especificando que se cometió un error. Dando la posibilidad al usuario a volver a ingresar por medio del botón Reintentar que se muestra en dicha página.



3. **Referente:** luego de ingresar correctamente al sistema, presiona botón **Carga de Trámites** y continúa con el procedimiento específico de carga.



Se trata de un procedimiento distribuido, autónomo y totalmente descentralizado para la carga de datos en la que la designación de los responsables se hace en un primer momento y luego debe confiarse en el acceso remoto de los mismos asegurando su identidad a través de medios lógicos de autenticación. Un reto verdaderamente difícil para los mecanismos de autenticación electrónica tradicionales.

Desventajas de un sistema login password

Cuando usamos un típico sistema de usuarios y contraseñas, como el usado actualmente por la Guía, tenemos un nivel de identificación de usuarios débil, es decir, nos encontramos en el umbral de la seguridad para sistemas informáticos y dadas las condiciones y variables del entorno puede resultar adecuado o no. Tales conclusiones deberán surgir de la consideración del tipo de información que se está intercambiando con el sistema y la valoración de las posibles acciones que se puedan realizar en pos de quebrar la seguridad que éste método sugiere en función de la evolución del poder computacional disponible. Dichos planteos se resolverán en el (apartado III) del presente informe, por ahora debemos tener claro que el método utilizado actualmente no garantiza:

- ***Identificación unívoca***: el usuario no sabe que está ingresando a su sitio o a una réplica.
- ***Confidencialidad***: la información puede ser interceptada.

- ***Integridad:*** los datos pueden llegar incompletos y con posibilidad de error.
- ***No repudio:*** la información no es digitalmente firmada probando así que fue enviado por cierta persona evitando el rechazo de la misma.

Sin duda son éstas falencias las que se han tenido en cuanta al momento de determinar mejoras en la seguridad del sistema (apartado IV). Tal situación la retomaremos más adelante

Determinación de Actores:

A continuación se detalla una lista de actores que tiene a su cargo la explicitación y la actualización de información en la Guia de Trámites:

MINISTERIO	REFERENTE
Secretaría Administrativa Legal y Técnica	Luis Benvenutto
Desarrollo Social y Salud	María José Santamarina
Justicia y Seguridad	Deolinda Suarez
Hacienda	Ada Curti
Economía	Susana Choren
Ambiente y Obras Públicas	Andrea Paredes
Dirección General de Escuelas	María Selva Trevisán
Subsecretaria de Cultura	Fabián Wunkhaus
Gobierno	Patricia Galán
Dirección de Cooperativas	Paola Mauvezin
Dirección de Deportes	Pedro Castroviejo
Municipalidad de Maipú	Ricardo Cirrincione, Horacio Lena
IPV	Beatriz Rinaldi - Planificación
ADUANA	Guillermo Filippini
IRRIGACION	Cecilia Lopez
LEGISLATURA	Salvador Navarria / Ma. Del Carmen Scheble
MIGRACIONES	Hebe Gazzola
Penitenciaria	Marcelo Lavisari
Municipalidad de Guaymallén	Claudio Ocampo

Municipalidad de Godoy Cruz	Martín Ramirez
Dirección de Catastro	Raquel Godoy
Personas Jurídicas	Alberto Cruz

Una vez planteado el escenario general en el que nos encontramos, pasaremos ahora a desarrollar precisiones operativas.

B) Explicitación de la necesidad puntual

Retomamos aquí, el desafío planteado en el apartado anterior de determinar claramente cuáles son las necesidades de seguridad que el sistema demanda y cuáles son los fundamentos de la aplicación de Sitio Seguro en la Guía de Trámites.

Estrategia para identificación de procedimientos aptos

Sometimos al circuito de carga de la Guía de Trámites a consideración de nuestra "Estrategia para la Identificación de Procedimientos Aptos", ya que consideramos importante hacer una adecuada elección del alcance y los ámbitos de aplicación de los servicios de criptografía asimétrica para lograr ahorros y no incurrir en sobrecostos de implementación.

Según los "Criterios de selección de circuitos administrativos" de nuestra estrategia, las conclusiones fueron:

- ***Circuitos administrativos de transferencia de información con exigencias de calidad en la información:*** resulta uno de los objetivos primordiales de la "Guía de Trámites" asegurar la calidad de sus contenidos en términos de información fidedigna, actualizaciones oportunas y responsabilidad de los referentes por la carga. No resulta ilógico además, pedir que dicha información no pueda ser alterada mientras viaja al servidor que la almacena para luego ofrecerla al ciudadano.

- ***Circuitos que requieren autenticación de las partes involucradas:*** si lugar a dudas, el circuito de carga de los referentes responsables de ésta tarea necesita autenticación unívoca de las partes. Asegurando que las únicas personas que pueda acceder a los contenidos de la Guía son aquellas que fueron designadas para esa tarea. Desde el otro lado, garantizando a los referentes de carga que la información que están transmitiendo es recibida por el sistema de la Guía y no por otro.

- ***Circuitos administrativos que enlazan importantes distancias geográficas:*** la extensión territorial que abarca el circuito de carga resulta importante. Si observamos la lista de referentes encontraremos municipalidades como la de Malargüe que se encuentran alejadas por más de 350 kilómetros de nuestra Unidad de Reforma, la aplicación de técnicas criptográficas en las instancias del procedimiento, como por ejemplo la autenticación remota de los referentes, soluciona muchos problemas de la no presencialidad y ahorra considerables costos de traslado y tiempo.

- ***Circuitos que incluyen información estrictamente confidencial:*** este criterio no hace alusión al circuito de carga, sino más bien tiene en cuenta la seguridad de los datos que los usuarios de la guía mandan vía web para la realización de tramitaciones on-line. Datos que no pueden ser publicados sin el consentimiento expreso de sus propietarios

Necesidades puntuales

Nuestra necesidad es la de dotar al sistema de carga de la Guía de Trámites de seguridad en la autenticación y el intercambio de los datos provenientes de los usuarios autorizados para la carga y de la información contenida en la base de datos, mediante métodos de encriptación que aseguran la identidad de las partes involucradas y el traslado seguro de los datos transmitidos.

En concreto debemos asegurar:

- **Protección de los sistemas de transferencia o transporte.** En este caso debemos garantizar, en el diseño del sistema la transferencia segura de la información de forma transparente al usuario final. Ejemplos de este tipo de planteamientos serían el establecimiento de un nivel de transporte seguro, de un servicio de transmisión de datos seguro.
- **Gestión de claves: Éste** es un tópico de capital importancia, al que se aplica el uso de la criptografía y debe estar resuelto antes de que el usuario esté en condiciones de enviar un solo bit seguro. En el caso de las claves secretas el problema mayor consiste en mantener su privacidad durante su distribución, en caso de que sea inevitable su envío de un punto a otro. En el caso de clave pública, los problemas tienen que ver con la garantía de que pertenecen a su titular y la confianza en su vigencia (que no haya caducado o sido revocada).
- **Autenticación del cliente:** representa la necesidad de la identificación inequívoca del referente por parte del el servidor al cual está accediendo y, se pueda garantizar categóricamente que la persona designada responsable de la carga es quién está modificando la base de datos de trámites.
- ***Identificación unívoca del servidor***: desde el otro lado necesitamos asegurarle al referente que está ingresando a su sitio y no a una réplica.
- ***Confidencialidad***: resulta de vital importancia garantizar que la información que se le carga al sistema llegue a la base de datos de una manera segura, evitando bajo todo punto de vista la interceptabilidad de la información
- ***Integridad:*** evitar la posibilidad de que los datos pueden llegar incompletos y con posibilidad de error.
- ***No repudio:*** debemos probar que la información cargada ha sido enviada por cierta persona evitando el rechazo de la misma, para asegurar los atributos de calidad buscados por la Guía de Trámites en sus contenidos.

Entorno Seguro

En la actualidad, la falta de medidas de seguridad en las redes es un problema que está en crecimiento. Cada vez es mayor el número de atacantes y cada vez están más organizados, por lo que van adquiriendo día a día habilidades más especializadas que les permiten obtener mayores beneficios en su labor de piratería.

La propia complejidad de la red utilizada por la Administración Pública Provincial, es una dificultad para la detección y corrección de los múltiples y variados problemas de seguridad.

Se expone a continuación una lista exhaustiva de potenciales problemas analizados y que se transcribe por considerarla como un importante antecedente tenido en cuenta en el desarrollo de Sitio Seguro para la Guía de Trámites.

PROBLEMAS EN SISTEMAS CONECTADOS A INTERNET

Fuentes: "Firewalls and Internet Security. Repelling the Wily Hacker"

1.- De todos los problemas, el mayor son los fallos en el sistema de passwords.

2.- Los sistemas basados en la autenticación de las direcciones se pueden atacar usando números consecutivos.

3.- Es fácil interceptar paquetes UDP.

4.- Los paquetes ICMP pueden interrumpir todas las comunicaciones entre dos nodos.

5.- Los mensajes ICMP Redirect pueden corromper la tabla de rutas.

6.- El encaminamiento estático de IP puede comprometer la autenticación basada en las direcciones.

7.- Es fácil generar mensajes RIP falsos.

8.- El árbol inverso del DNS (Server Name Domain) se puede usar para

PROBLEMAS EN SISTEMAS CONECTADOS A INTERNET

Fuentes: "Firewalls and Internet Security. Repelling the Wily Hacker"

conocer nombres de máquinas.

9.- Un atacante puede corromper voluntariamente la caché de su DNS para evitar responder peticiones inversas.

10.- Las direcciones de vuelta de un correo electrónico no son fiables.

11.- El programa sendmail es un peligro en sí mismo.

12.- No se deben ejecutar a ciegas mensajes MIME.

13.- Es fácil interceptar sesiones telnet.

14.- Se pueden atacar protocolos de autenticación modificando el NTP.

15.- Finger da habitualmente demasiada información sobre los usuarios.

16.- No debe confiarse en el nombre de la máquina que aparece en un RPC.

17.- Se puede conseguir que el encargado de asignar puertos IP ejecute RPC en beneficio de quien le llama.

18.- Se puede conseguir, en muchísimos casos, que NIS entregue el fichero de passwords al exterior.

19.- A veces es fácil conectar máquinas no autorizadas a un servidor NIS.

20.- Es difícil revocar derechos de acceso en NFS.

21.- Si está mal configurado, el TFTP puede revelar passwords.

22.- No debe permitirse al ftp escribir en su directorio raíz.

23.- No debe ponerse un fichero de passwords en el área de ftp.

24.- A veces se abusa de FSP, y se acaba dando acceso a ficheros a quien no se debe dar.

25.- El formato de información de WWW debe interpretarse cuidadosamente.

26.- Los servidores WWW deben tener cuidado con los punteros de ficheros.

PROBLEMAS EN SISTEMAS CONECTADOS A INTERNET

Fuentes: "Firewalls and Internet Security. Repelling the Wily Hacker"

27.- Se puede usar ftp para crear información de control del gopher.

28.- Un servidor WWW puede verse comprometido por un script interrogativo pobremente escrito.

29.- El MBone se puede usar para atravesar algunos tipos de cortafuego.

30.- Desde cualquier sitio de la Internet se puede intentar la conexión a una estación X11 (X-Server).

31.- No se debe confiar en los números de puerto facilitados remotamente.

32.- Es casi imposible hacer un filtro seguro que deje pasar la mayoría del UDP.

33.- Se puede construir un túnel encima de cualquier transporte.

34.- Un cortafuego no previene contra niveles superiores de aquellos en los que actúa.

35.- Las X11 son muy peligrosas incluso a través de una pasarela.

36.- Las herramientas de monitorización de red son muy peligrosas si alguien accede ilegítimamente a la máquina en que residen.

37.- Es peligroso hacer peticiones de finger a máquinas no fiables.

38.- Se debe de tener cuidado con ficheros en áreas públicas cuyos nombres contengan caracteres especiales.

39.- Los caza-passwords actúan silenciosamente.

40.- Hay muchas maneras de conseguir copiar el password

41.- Registrando completamente los intentos fallidos de conexión, se capturan passwords.

42.- Un administrador puede ser considerado responsable -si se demuestra conocimiento o negligencia- de las actividades de quien se introduce en sus máquinas.

Entonces, la constitución de un SITIO SEGURO consiste en la seguridad en la comunicación a través de redes, especialmente Internet, consistente en prevenir, impedir, detectar y corregir violaciones a la seguridad durante la transmisión de información, más que en la seguridad en los ordenadores, que abarca la seguridad de sistemas operativos y bases de datos. Consideraremos la información esencialmente en forma digital y la protección se asegurará mayormente mediante medios lógicos, más que físicos.

Los mecanismos básicos pueden agruparse de varias formas para proporcionar los servicios previamente mencionados. Conviene resaltar que los mecanismos poseen tres componentes principales:

- Una información secreta, como claves y contraseñas, conocidas por las entidades autorizadas.
- Un conjunto de algoritmos, para llevar a cabo el cifrado, descifrado, hash (desmenuce de un mensaje compilado) y generación de números aleatorios.
- Un conjunto de procedimientos, que definen cómo se usarán los algoritmos, quién envía qué a quién y cuándo.

Asimismo es importante notar que los sistemas de seguridad requieren una gestión de seguridad. La gestión comprende dos campos bien amplios:

- Seguridad en la generación, localización y distribución de la información secreta, de modo que sólo pueda ser accedida por aquellas entidades autorizadas.
- La política de los servicios y mecanismos de seguridad para detectar infracciones de seguridad y emprender acciones correctivas.

C) Determinación de mejoras

Como ya mencionamos, un sitio seguro se reconoce a través de la emisión de **un certificado** (también conocido como certificado de clave-publica o identificador digital) es un documento electrónico, emitido por una

Autoridad Certificadora, que **identifica de forma segura al poseedor del mismo** evitando la suplantación de identidad por terceros. Podría compararse con el DNI (Documento Nacional de Identidad). Es justamente en las instancias de acceso e intercambio de datos al sistema en dónde se previeron y desarrollaron mejoras utilizando tecnología de firma digital.

¿Cómo funciona?

La idea es que un usuario o entidad "prueba" su identidad a otra demostrando que conoce un secreto, pero sin revelarlo (algo simple es usar criptografía simétrica o asimétrica). Por ejemplo, si un usuario comparte una clave secreta con el servidor, este le puede enviar un mensaje cifrado con una pregunta y si la responde correctamente, demostró que posee la clave. Con criptografía asimétrica el server encripta algo con la clave pública del usuario y este demuestra su identidad cuando desencripta correctamente

•Client Hello : El "saludo de cliente" tiene por objetivo informar al servidor qué algoritmos de criptografía puede utilizar y solicitar una verificación de la identidad del servidor. El cliente envía el conjunto de algoritmos de criptografía y compresión que soporta y un número aleatorio. El propósito del número aleatorio es para que en caso de que el servidor no posea un certificado para comprobar su identidad, aún se pueda establecer una comunicación segura utilizando un conjunto distinto de algoritmos. Dentro de los protocolos de criptografía hay un protocolo de intercambio de llave que define cómo cliente y servidor van a intercambiar la información, los algoritmos de llave secreta que definen que métodos pueden utilizar y un algoritmo de hash de una sola vía. Hasta ahora no se ha intercambiado información secreta, solo una lista de opciones.

•Server Hello: El servidor responde enviando su identificador digital el cual incluye su llave pública, el conjunto de algoritmos criptográficos y de compresión y otro número aleatorio. La decisión de qué algoritmos serán utilizados está basada en el más fuerte que tanto cliente como servidor soporten. En al-

gunas situaciones el servidor también puede solicitar al cliente que se identifique solicitando un identificador digital.

•Aprobación del Cliente: El cliente verifica la validez del identificador digital o certificado enviado por el servidor. Esto se lleva a cabo desencriptando el certificado utilizando la llave pública del emisor y determinando si este proviene de una entidad certificadora de confianza. Después se hace una serie de verificaciones sobre el certificado, tales como fecha, URL del servidor, etc. Una vez se ha verificado la autenticidad de la identidad del servidor. El cliente genera una llave aleatoria y la encripta utilizando la llave pública del servidor y el algoritmo criptográfico y de compresión seleccionado anteriormente. Esta llave se le envía al servidor y en caso de que el handshake tenga éxito será utilizada en el envío de futuros mensajes durante la sesión.

•Verificación: En este punto ambas partes conocen la llave secreta, el cliente por que la generó y el servidor por que le fue enviada utilizando su llave pública, siendo la única forma posible de desencriptarla utilizando la llave privada del servidor. Se hace una última verificación para comprobar si la información transmitida hasta el momento no ha sido alterada. Ambas partes se envían una copia de las anteriores transacciones encriptada con la llave secreta. Si ambas partes confirman la validez de las transacciones, el handshake se completa, de otra forma se reinicia el proceso.

Ahora ambas partes están listas para intercambiar información de manera segura utilizando la llave secreta acordada y los algoritmos criptográficos y de compresión.

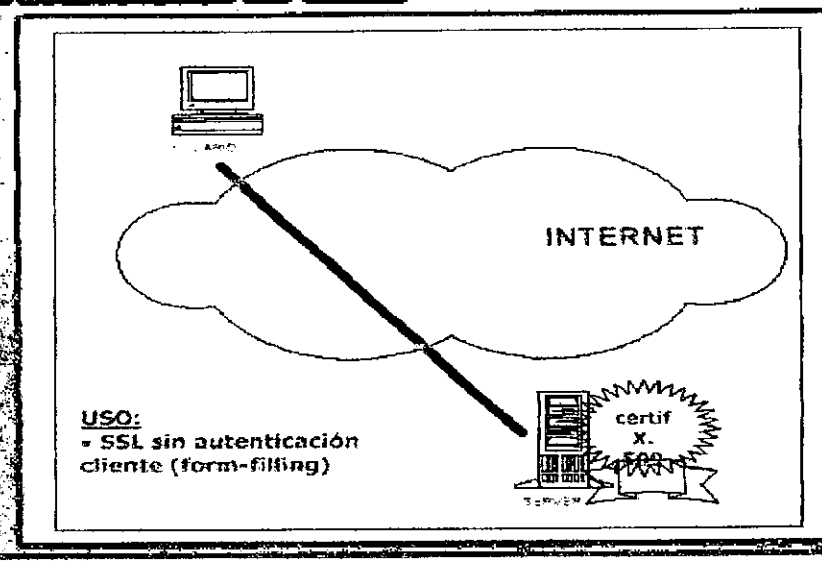
Modelos y mejoras puntuales

Concretamente existen dos modelos para mejorar la seguridad del sistema a través de la implementación de Sitio Seguro:

4 USUARIO PUBLICO GRAL

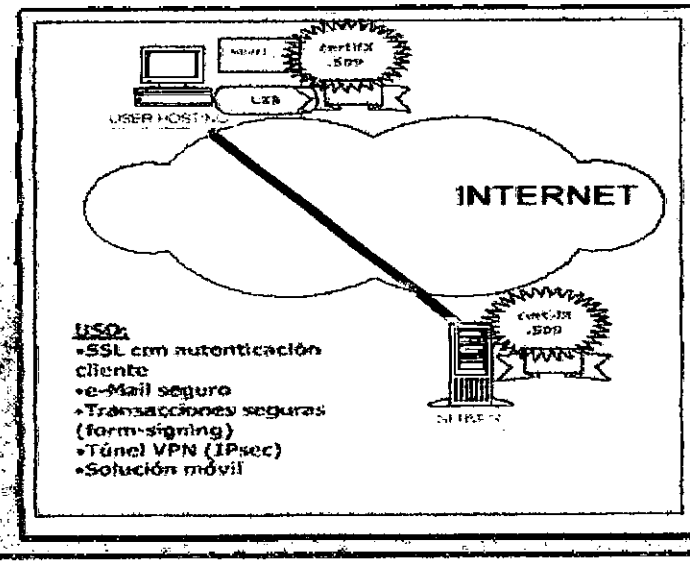
Modelo I

autenticación del Servidor con Certificado Digital, sin autenticación del cliente



Modelo II

autenticación mutua



V USUARIO RED INT.

Modelo I: En el primer modelo la aplicación de ésta tecnología proporciona:

- **Autenticación mutua entre el servidor seguro:** el cliente tiene la garantía de estar *hablando* con el servidor al que accede.
- **Privacidad en el intercambio de información:** sólo el cliente y el servidor seguro conocen lo que viaja por la red. Nadie distinto a ellos podrá leer la información que intercambien. Ahora que se ha establecido un canal de transmisión seguro SSL, es posible el intercambio de datos. Cuando el servidor o el cliente desea enviar un mensaje al otro, se genera un digest (utilizando un algoritmo de hash), encriptan el mensaje y el digest y se envía, cada mensaje es verificado utilizando el digest.

***"Este modelo será usado para la transmisión de datos confidenciales de los usuarios de e-trámite de la Guía de Trámites con el objeto de proveer una serie de garantías"**, a saber:*

- ***Identificación unívoca***: Constituye una mejora fundamental al momento de aportarle al usuario la total seguridad de que sus datos están siendo ingresados por el sitio Guía de Trámites y no en una réplica del mismo.
- ***Confidencialidad***: la información que viaje desde el usuario a la Base de Datos no podrá ser interceptada. Constituye una mejora radical en la seguridad en el traspaso de datos mediante herramientas de encriptado de información. Contribuye al aseguramiento de la integridad y veracidad de la información.
- ***Integridad:*** los datos envidos por usuarios llegarán completos y sin posibilidad de error, ya que la tecnología garantiza la verificación de la integridad de los mensajes mediante la aplicación de una función de hash. Cualquier diferencia, pérdida de datos, o modificación de los datos originales enviados será alertada por el sistema.

De esta forma, todos los datos provenientes de los usuarios que realizan trámites a través de la Guía se resguardan, mediante métodos de encriptación que aseguran la integridad y confidencialidad de la información que viaja por la web.

Modelo II: En el segundo modelo la aplicación de ésta tecnología proporciona:

- **Autenticación mutua entre el servidor seguro y el cliente.** El servidor sabe con total seguridad quien es el cliente que esta al otro lado y el cliente tiene la garantía de estar *hablando* con el servidor al que accede.
- **Privacidad en el intercambio de información.** Sólo el cliente y el servidor seguro conocen lo que viaja por la red. Nadie distinto a ellos podrá leer la información que intercambien. Ahora que se ha establecido un canal de transmisión seguro SSL, es posible el intercambio de datos. Cuando el servidor o el cliente desea enviar un mensaje al otro, se gene-

ra un digest (utilizando un algoritmo de hash), encriptan el mensaje y el digest y se envía, cada mensaje es verificado utilizando el digest.

***"Este es el modelo que se ha elegido para el circuito de carga de información por parte de los referentes de la Guía de Trámites con el objeto de proveer una serie de garantías"**, a saber:*

- ***Identificación unívoca***: Constituye una mejora fundamental al momento de aportarle al referente la total seguridad de que sus datos están siendo ingresados por el sitio Guía de Trámites y no en una réplica del mismo, por otro lado garantiza la identidad del referente que está cargando información ante el Sitio Guía de Trámites, aporte fundamental a la calidad de los contenidos on-line.
- ***Confidencialidad***: la información que viaje desde el referente a la Base de Datos no podrá ser interceptada. Constituye una mejora radical en la seguridad en el traspaso de datos mediante herramientas de encriptado de información. Contribuye al aseguramiento de la integridad y veracidad de la información.
- ***Integridad:*** los datos envidos por los referentes llegarán completos y sin posibilidad de error, ya que la tecnología garantiza la verificación de la integridad de los mensajes mediante la aplicación de una función de hash. Cualquier diferencia, pérdida de datos, o modificación de los datos originales enviados será alertada por el sistema.
- ***No repudio:*** la información es digitalmente firmada por el referente probando así que fue enviada por este y por nadie más, responsabilizándolo por la calidad de la información cargada y por la oportunidad en las actualizaciones que se realicen

De esta forma, todos los datos provenientes de los usuarios autorizados para la carga y la información contenida en la base de datos se resguar-

dan, mediante métodos de encriptación que aseguran la identidad de las personas autorizadas a realizar modificaciones en los datos.

Hemos pasado de un sistema de identificación débil a un sistema de identificación fuerte, calificado mundialmente como uno de los más seguros hasta el momento.

D) Desarrollo de documentación explicativa de Sitio Seguro

El siguiente documento forma parte de la información disponible en la Guía de Trámites y tiene como objetivo explicar el funcionamiento de la tecnología de Sitio Seguro.

Con esto se espera difundir el uso de la herramienta a través de sus características y utilidades promoviendo la conciencia en el usuario del cuidado de la seguridad de sus datos en internet

Sitio Seguro en la Guía de Trámites

Como norma general y mientras no se advierta de lo contrario, cuando usted rellena un formulario y pulsa el botón enviar, está enviando toda la información en forma de datos a través de la red. Datos que son transmitidos de servidor en servidor hasta llegar a su destinatario y que pueden ser interferidos o robados antes de llegar a su destino.

Por eso se hace necesaria la utilización de tecnologías que permitan salvaguardar la privacidad de sus datos.

El protocolo SSL permite la autenticación de servidores, la codificación de datos y la integridad de los mensajes. Con SSL sus comunicaciones en Internet serán transmitidas en formato codificado. De esta manera, la información que envíe llegará de manera privada y no será manipulada. Además, usted tendrá la seguridad de que está ingresando al sitio de la Guía de Trámites y no a una réplica.

¿Qué significa SSL?

Son las siglas inglesas correspondientes a Secure Sockets Layer. El protocolo SSL fue desarrollado por Netscape para permitir confidencialidad y autenticación en Internet.

SSL opera como una capa adicional entre Internet y las aplicaciones

De acuerdo con la convención establecida, la dirección de las páginas Web que requieren una conexión SSL comienza con https: en lugar de http:

"La Guía de Trámites garantiza que todos los procesos de captación y transferencia de información facilitada por los usuarios y referentes es transferida mediante el protocolo de seguridad SSL Secure Sockets Layer (servidor seguro) desde su navegador hasta nuestros servidores"

¿Cómo funciona un servidor seguro?

Para establecer una comunicación segura utilizando SSL se deben de cumplir una serie de requisitos:

1. Cuando un usuario accede a la Guía de Trámites a través de su dirección url segura https://www.tramite.mendoza.gov.ar se establece la conexión y el navegador solicita una conexión segura. Una vez se ha hecho la solicitud, el cliente y el servidor empiezan a negociar la conexión SSL.
2. Como el servidor al que accede es un servidor seguro, este responderá afirmativamente a la solicitud, enviándole un certificado electrónico de tipo RSA.

3. Después de recibir este certificado el navegador lo desempaquetará con la clave de la autoridad de certificación, ya integrada en el software, obteniendo de este modo la clave según el algoritmo RSA.

4. Por último, el navegador genera una clave de encriptación simétrica y se la envía encriptada al servidor (con su clave pública). A partir de este momento, tanto el navegador como el servidor pueden establecer una comunicación segura basada en esta clave simétrica, que ambos y sólo ellos conocen.

"Las claves simétricas son generadas aleatoriamente en cada sesión, por lo cual no hay posibilidad de que estas sean conocidas por eventuales hackers"

¿Cómo puedo saber si realmente estoy en un servidor seguro?

Es sencillo saber si hemos conectado con un servidor seguro. En primer lugar, la dirección de URL comienza por https:// en vez de http:// (a esta dirección se accede, a veces, sin intervención del usuario, debido a que se pulsa una palabra clave que la lleva incorporada, o bien intencionadamente cuando se desea acceder a un servidor en modalidad segura). Además, en la mayoría de los visualizadores tendremos una indicación de que la conexión segura se ha establecido.

Una llave o un candado cerrado en la parte izquierda (Netscape), o bien, un candado cerrado en la parte derecha (Explorer y Navigator). En el caso de Opera, aparece en la parte superior izquierda.

Iconos identificativos de los navegadores más usados:

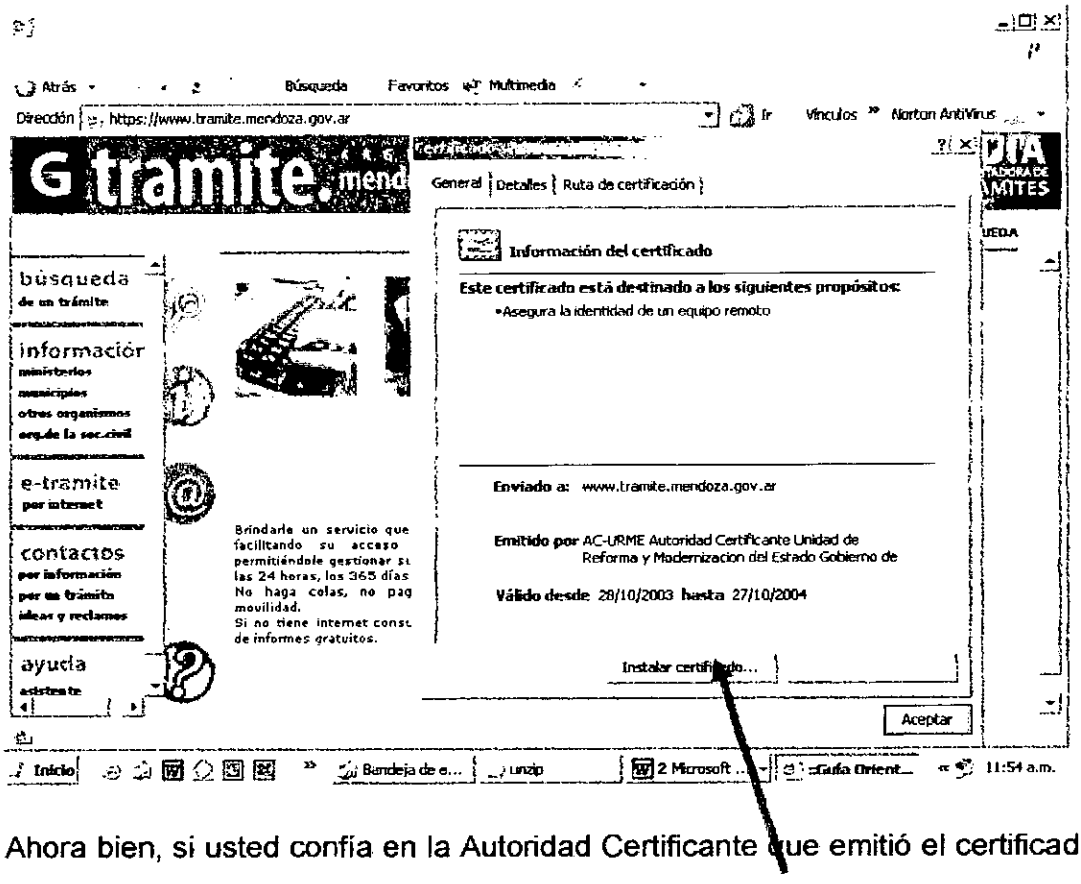
 Explorer

 Netscape

 Navigator

 Opera

Además, es importante comprobar que el certificado de seguridad otorgado es válido y vigente haciendo clic en el icono del candado:



Ahora bien, si usted confía en la Autoridad Certificante que emitió el certificado y además el certificado no ha caducado, haga clic en **instalar certificado**

Consejo!!!

Los sitios auténticos utilizan certificados del servidor de la red SSL para ofrecer comunicaciones seguras por desciframiento todos los datos a y desde el sitio. Siempre examine el certificado de un sitio antes de incorporar cualquier información.

Nunca ofrezca información confidencial por Internet sin ningún tipo de protección, especialmente si son números de tarjeta de crédito.

Muy Importante!!!

En el caso de la Guía de Trámites la dirección segura a través de la que se ejecutan todos los formularios es https://www.tramites.mendoza.gov.ar y el certificado ha sido otorgado por la **Autoridad Certificante de la Unidad de Reforma y Modernización del Estado de la Provincia de Mendoza (AC-URME).**

El servicio de Sitio Seguro en la Guía de Trámites es prestado por: firma Digital

Un proyecto de la Unidad de Reforma y Modernización del Estado

Para mayor información por favor dirigirse a www.firmadigital.mendoza.gov.ar

---

E) Determinaciones sobre la provisión de certificados

Provisión de Certificados Digitales

Habitualmente, resulta una decisión muy importante determinar cuál será el agente que emitirá y gestionará los certificados que se piensan aplicar a una experiencia piloto determinada. En su momento, cuando se implementó la experiencia piloto en e-democracia, las circunstancias particulares de la aplicación demandaban el uso de certificados digitales de alta confianza y renombre para la sociedad ya que:

- Era la primer experiencia piloto que se implementaba en la provincia de Mendoza

- Los certificados iban a ser usados en un circuito semiabierto, es decir, la provisión de los certificados alcanzaba a ciudadanos (candidatos a las elecciones)

- Las plataformas políticas eran publicadas y el proyecto de firma digital debía asegurarles garantías de integridad a sus autores y de autoría a los ciudadanos que ingresaban al sitio. Además el candidato que firmaba su propuesta, se hacía responsable ante la sociedad de cumplirla.

Cualquier falla en el sistema hubiera perjudicado fuertemente la credibilidad de una tecnología poco difundida e indirecta o directamente la imagen del candidato político que publicaba

- La planificación de una infraestructura provincial se encontraba en pleno proceso de planificación e investigación, por parte de los integrantes de este proyecto y por lo tanto, no se estaba en condiciones de afrontar las responsabilidades por la emisión y gestión de certificados.

Los factores precedentes impulsaron la celebración de un convenio entre la Unidad de Reforma del Estado y la Compañía Certisur S.A, representante de la prestigiosa Verisign en Latinoamérica. A través de este convenio el proyecto de firma digital 2003 contó con la infraestructura de una empresa de nivel mundial para la provisión de los certificados que serían usados en la experiencia piloto de e-democracia, que se ajustaban perfectamente a las demandas particulares de la aplicación.

Actualmente, el escenario y las condiciones particulares de la experiencia piloto de Sitio Seguro en la Guía de Trámites determinan la consideración de factores diferentes a la hora de tomar una decisión respecto de la provisión de certificados, a saber:

- El equipo de la Unidad de Reforma ha desarrollado un fuerte know how gracias a la experiencia piloto desarrollada en e-democracia

- Paralelamente, las investigaciones sobre la tecnología disponible para el montaje de una Infraestructura de Clave Pública han alcanzado una considerable madurez
- La planificación y el diseño organizacional realizado durante el proyecto de firma digital 2003 proporciona una base conceptual robusta

- El circuito de carga de la Guía de Trámites puede considerarse como un esquema cerrado de implementación, dónde la provisión de los certificados alcanza a agentes ubicados dentro de la estructura organizativa de la Administración Pública Provincial

- El avance en la normativa inherente al respaldo legal de la tecnología de firma digital, tanto a nivel nacional con la designación del nuevo organismo que hará las veces de Autoridad de Aplicación, como a nivel provincial con la Ley de adhesión provincial.

Por lo antedicho, el equipo del proyecto de firma digital ha decidido afrontar la provisión y gestión de certificados para la experiencia piloto de Sitio Seguro en la Guía de Trámites a través del **Prototipo de Infraestructura de Clave Pública AC-URME**, cuyas funciones básicas ya se encuentran implementadas, faltando desarrollos periféricos que no afectaran el correcto funcionamiento de la experiencia de Sitio Seguro. Dichos avances se informan en la segunda parte del presente informe.

Esquema de Emisión de Certificados Digitales

Se ha decidido acotar la experiencia piloto de Sitio Seguro a través de la emisión y gestión de los siguientes Certificados Digitales:

- Un (1) certificado de servidor (web server) para la identificación univoca del Sitio web de la Guía de Trámites ante sus usuarios y referentes
- Siete (7) Certificados Digitales personales para identificación unívoca de los referentes más importantes de la Guía de Trámites

La circunscripción de la experiencia a la emisión de 7 certificados digitales responde a criterios de aseguramiento de la confiabilidad del sistema, que en principio se ha implementado en paralelo con el esquema de seguridad hasta ahora sustentado por la Guía de Trámites.

En función de las conclusiones que arroje la evaluación general de resultados de la experiencia piloto de Sitio Seguro, se realizarán los ajustes necesarios y se tomarán las decisiones operativas sobre el futuro de la experiencia.

F) Capacitación de los referentes de la guía

Se ha realizado la capacitación, en forma personalizada, de los referentes de la guía de trámites en el uso del nuevo sistema de seguridad de Sitio Seguro para el circuito de carga de trámites.

La capacitación consistió en una serie de jornadas individuales en las que en algunos casos se citó al referente en la Unidad de Reforma y en otros el equipo de firma digital se trasladó al propio lugar de carga del referente.

El listado de referentes capacitados de acuerdo con las determinaciones sobre la provisión de certificados es el siguiente:

MINISTERIO	REFERENTE
Secretaría Administrativa Legal y Técnica	Luis Benvenuto
Hacienda	Ada Curti
Economía	Susana Choren
Ambiente y Obras Públicas	Andrea Paredes
Gobierno	Patricia Galán
IPV	Beatriz Rinaldi - Planificación
Penitenciaria	Marcelo Lavisari

Los temas y contenidos que se trabajaron el las jornadas individuales fueron los siguientes:

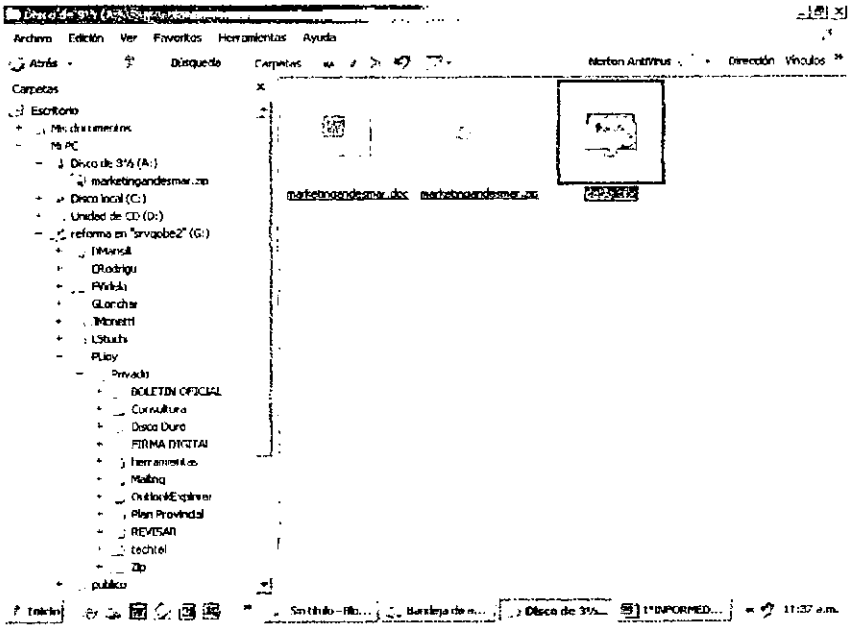
***Nociones básicas* sobre Sitio Seguro (Ver punto D)**

¿Cómo instalar un Certificado Digital?

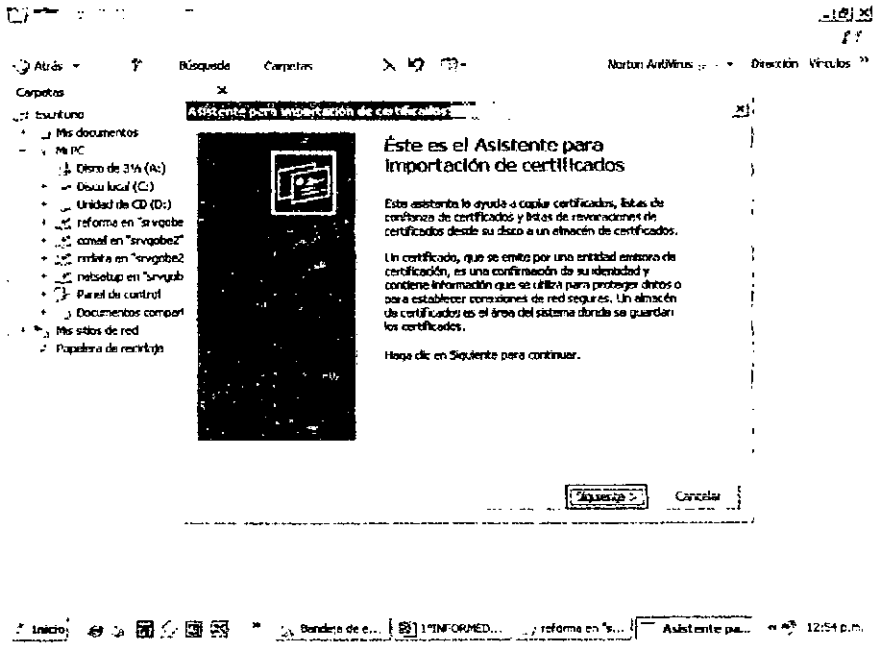
Acceso de referentes a la Guía de Trámites

¿Cómo instalar un Certificado Digital?

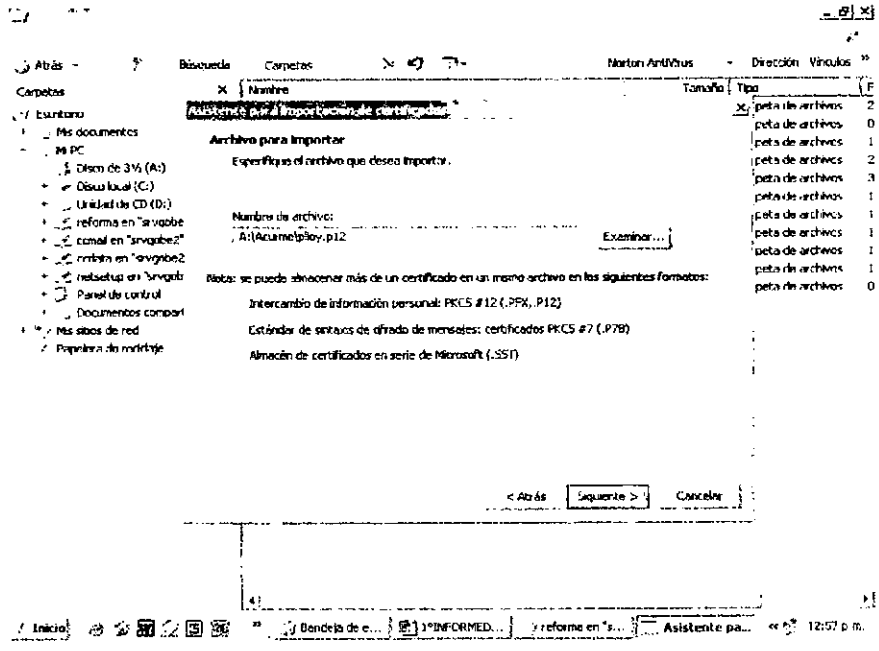
1.Utilizando el explorador de windows muestre el contenido de la unidad de disco A:\, a continuación haga doble click en el Certificado Digital:



2.Este es el asistente para la importación de su Certificado Digital y le ayudará a instalarlo haciendo clic en Siguiente

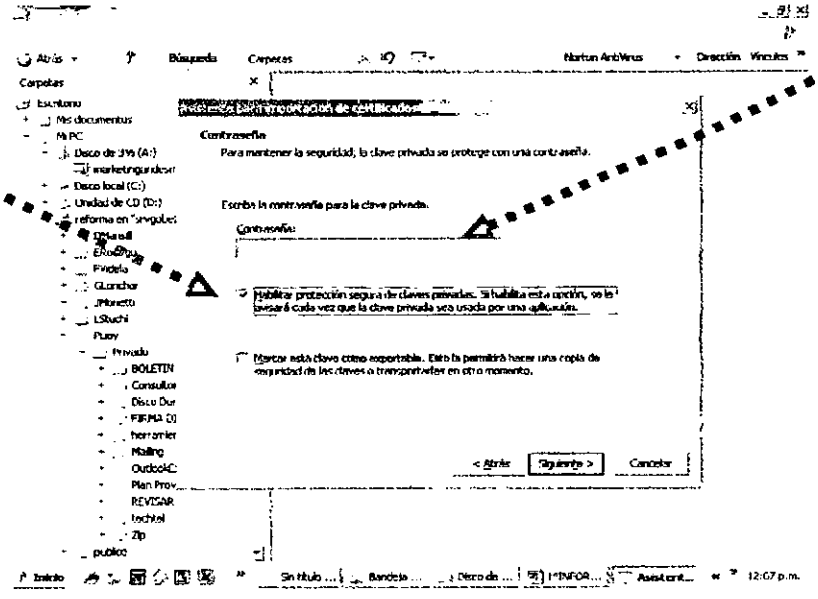


3.El asistente le mostrará la ubicación de la cuál importará el certificado, haga click en Siguiente si es correcta

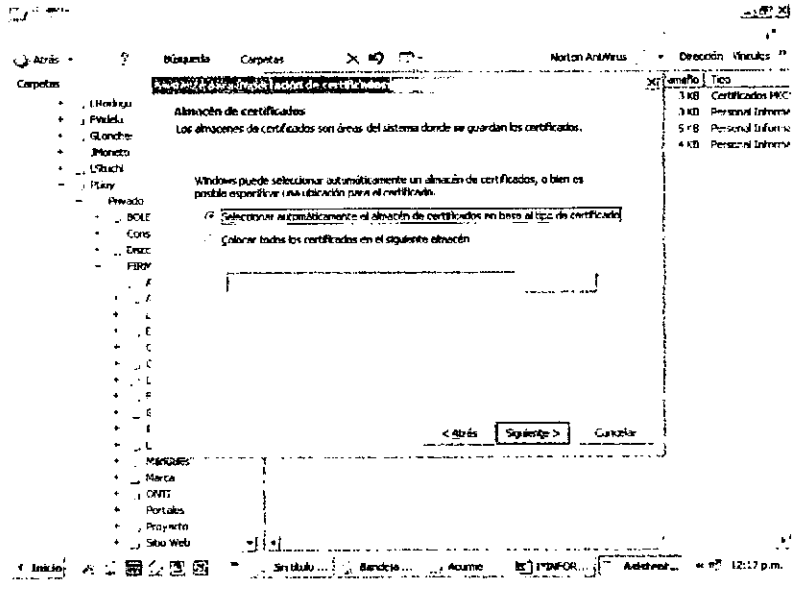


4.Luego, el asistente le mostrara una pantalla en donde deberá:

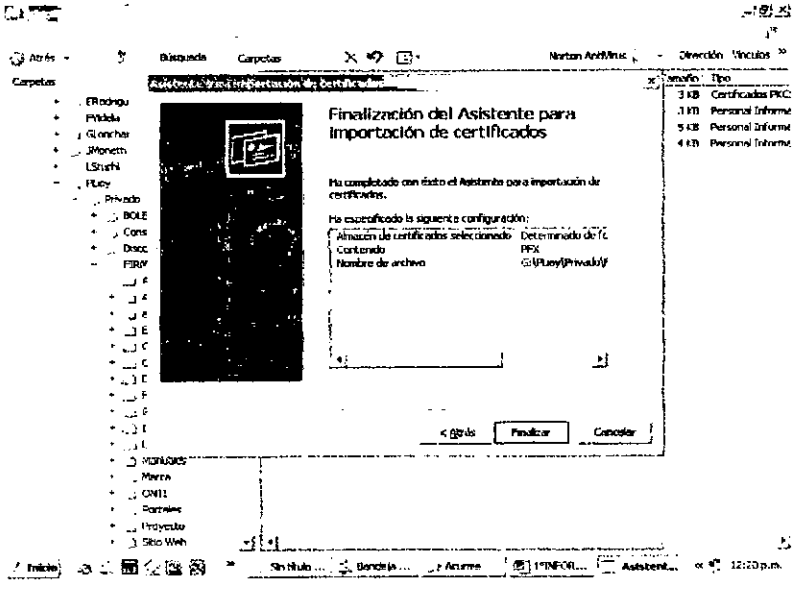
- Escribir una contraseña para proteger sus clave privada
- Habilitar protección segura para la clave privada



5.Automáticamente, el asistente seleccionará el repositorio en donde guardar sus Certificado Digital personal, presione Siguiente

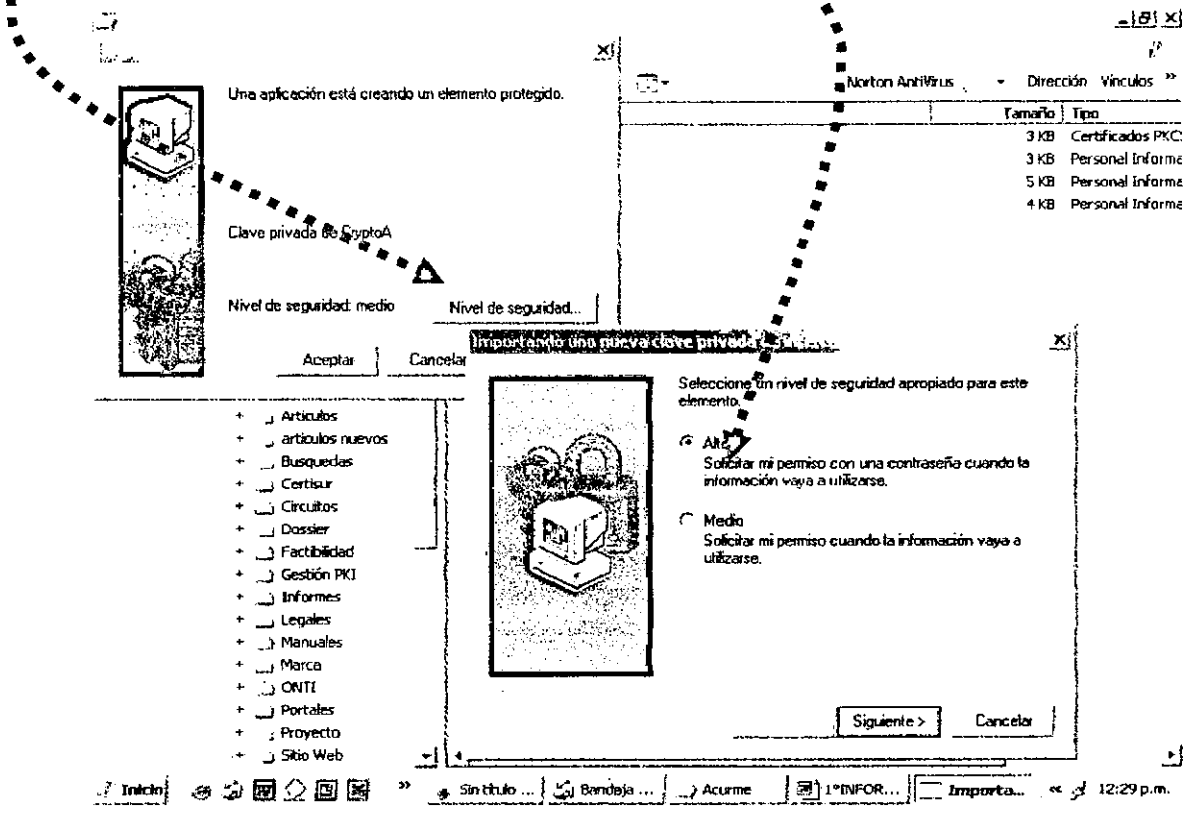


6.en la siguiente pantalla le mostrará cuál ha sido la ubicación elegida para sus Certificado, haga clic en Finalizar

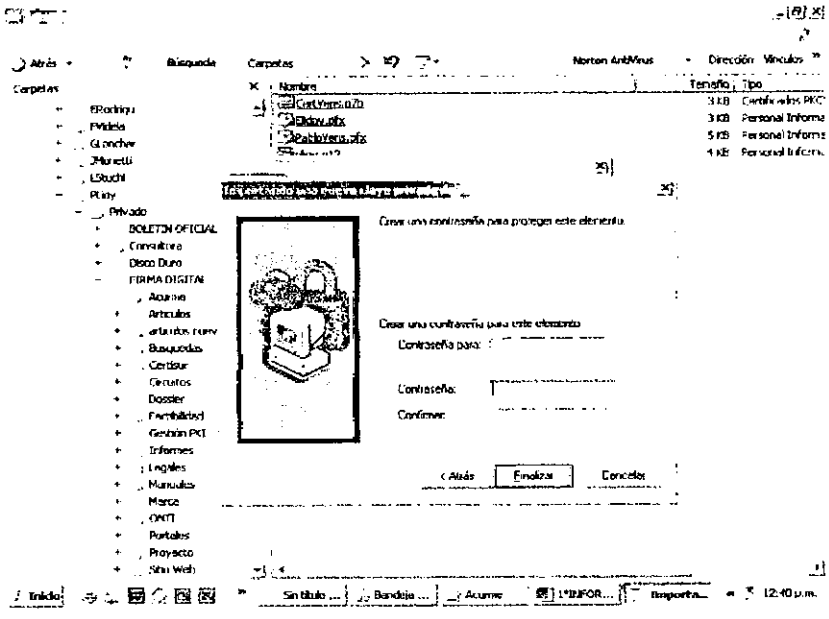


7.Luego usted deberá elegir el nivel de seguridad que quiere para proteger su clave privada en el "contenedor para claves privadas", entonces se recomienda:

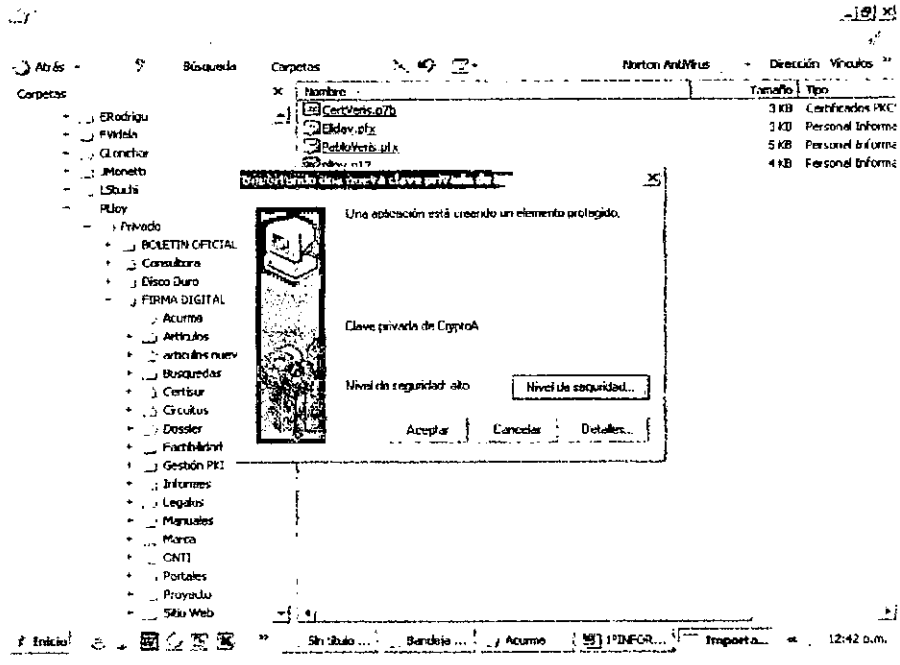
- Hacer click en nivel de seguridad
- Elegir un nivel alto de seguridad



- Escribir la contraseña que ingresó anteriormente y confirmarla



8. Finalmente seleccione Aceptar



Si todo salió correctamente el Asistente le informará que la importación fue exitosa y que ya puede disponer de su certificado.

Acceso de referentes a la Guía de Trámites

Para establecer una comunicación segura con la Guía de Trámites utilizando SSL se deben de cumplir una serie de pasos. Cada vez que un referente carga un nuevo trámite a la base de datos debe seguir el siguiente procedimiento:

Descripción del procedimiento

El presente procedimiento supone que el referente ya ha instalado en su computadora, y por lo tanto en su navegador de internet (browser) su Identificador Digital (Certificado emitido por la AC-URME: Autoridad Certificante de la unidad de Reforma y Modernización del Estado), tal como se explica en el documento ¿Cómo instalar un Certificado Digital?.Cabe señalar que una vez hecho esto, el referente sólo deberá realizar el primer paso del procedimiento dejando el resto a cargo del sistema.

1. **Referente:** accede a la Guía de Trámites a través de su dirección url segura https://www.admtramite.mendoza.gov.ar, establece la conexión y el navegador solicita una conexión segura.

2. **Servidor Guía de Trámites:** dado que es un servidor seguro, responderá afirmativamente a la solicitud, enviándole un certificado electrónico de tipo RSA emitido por la AC-URME.

3. **Browser del Referente**: Después de recibir este certificado el navegador lo desempaquetará con la clave de la autoridad de certificación, ya integrada en el software, obteniendo de este modo la clave según el algoritmo RSA.

4. **Servidor Guía de Trámites:** el servidor solicita al referente que se identifique a través de su identificador digital personal (Certificado emitido por la AC-URME). En algunos casos cuando hay más de un certificado instalado

en la máquina del referente, este deberá optar por el Certificado Digital para acceso a la Guía de Trámites.

5. **Servidor Guía de Trámites:** verifica la identidad del referente que solicita el acceso comprobando la validez del Certificado Digital instalado en su computadora. En el caso que el referente no posea el Identificador Digital que le fue otorgado o no sea el identificador correcto, se le deniega el acceso y el procedimiento concluye.

6. **Browser del Referente**: Por último, el navegador genera una clave de encriptación simétrica y se la envía encriptada al servidor (con su clave pública). A partir de este momento, tanto el navegador como el servidor pueden establecer una comunicación segura basada en esta clave simétrica, que ambos y sólo ellos conocen.

7. **Referente**: realiza la carga del trámite de acuerdo con el procedimiento de carga habitual del sistema.

G) Instalación de protocolos y configuración de servidores web

Para lograr la implementación de sitio seguro se trabajó en la instalación y configuración de tres componentes o aspectos básicos en los servidores de la Guía:

1. el Protocolo SSL y sus servicios
2. los Certificados de Servidor correspondientes
3. la configuración apropiada del Web Server para dar soporte SSL utilizando los certificados de servidor disponibles

Presentamos a continuación la documentación detallada de los pasos seguidos en cada uno de estos aspectos. Cabe aclarar que se no se necesitó instalación del protocolo SSL en los clientes, puesto que está embebido en la mayor parte de los browsers, tales como IE, Netscape Comunicator, etc.

Plataforma tecnológica del Servidor

- S.O. RedHat Linux 9.0
- WebServer: Apache Http Server
- SSL: mod-ssl
- Certificados: X509 v3 – PEM encoded

1. El protocolo SSL

Secure Socket Layer (SSL) es un protocolo desarrollado inicialmente por Netscape Communications Corporation para dar seguridad a la transmisión de datos en Internet. Utilizando la criptografía de clave pública, SSL provee autenticación de servidor y validación de cliente, encripción de datos sobre la capa de transporte, e integridad de los datos en las comunicaciones cliente/servidor.

Para la Guía de Trámites se implementó a través de SSL, cifrado de 128 bits totalmente compatible con los principales browsers Microsoft y Netscape.

Para activar un servidor seguro, se necesita, como mínimo, tener instalados los siguientes tres ***paquetes de software*** sobre la plataforma linux:

httpd

El paquete httpd contiene el demonio httpd y otras utilidades relacionadas, archivos de configuración, iconos, Servidor Apache HTTP módulos, páginas de manual y otros archivos utilizados por Servidor Apache HTTP.

mod_ssl

El paquete mod_ssl incluye el módulo mod_ssl, que proporciona criptografía fuerte para el servidor web, Servidor Apache HTTP a través de los protocolos SSL, Secure Sockets Layer y TLS, Transport Layer Security.

openssl

El paquete openssl contiene el conjunto de herramientas de OpenSSL. El conjunto de herramientas de OpenSSL implementa los protocolos SSL y TLS y también incluye una librería criptográfica de propósito general. Este paquete no es fundamental para el desarrollo de sitio seguro pero su instalación aporta herramientas útiles al desarrollo de algunas aplicaciones, a la realización de pruebas, comprobaciones y conversión de formatos.

Estos paquetes son incluidos por defecto en RedHat Linux 9 y sus servicios fueron habilitados y correctamente configurados.

Otros paquetes opcionales que podrían cargarse para dar funcionalidades especiales son: httpd-devel, openssh, openssh-askpass, openssh-askpass-gnome, openssh-clients, openssh-server,openssl-devel, stunnel. Sobre ninguno de estos paquetes se trabajó particularmente para la implementación de sitio seguro sobre la Guía, pero se los investigó pensando en futuras aplicaciones tales como seguridad en VPN.

H) Gestión/emisión de certificados

Como se ha mencionado previamente, el desarrollo de sitio seguro proporciona seguridad usando una combinación del protocolo SSL (Secure Sockets Layer) y certificados digitales. SSL maneja las comunicaciones encriptadas y la mutua autentificación entre navegadores y el servidor seguro. Los certificados SSL proporcionan autentificación para el servidor seguro.

Ante la alternativa de trabajar con certificados autofirmados; se decidió utilizar los certificados que podía emitir la AC-URME (Autoridad Certificante de la Unidad de Reforma y Modernización del Estado), aún en su carácter de prototipo, con dos objetivos.

1. Probar los servicios del software PKI implementado y sus desarrollos complementarios, en una aplicación concreta y con un marco procedimental determinado.
2. Instaurar la necesidad de contar con una Autoridad Certificante a la hora de emprender este tipo de desarrollos. Esto tiende a generar conciencia de que es la Autoridad Certificante quien proporciona garantías concernientes a la identidad de la organización que provee el sitio web.

A continuación se presenta a modo de marco general, el ***procedimiento informático*** que debe desarrollarse para obtener un Certificado Digital firmado por una Autoridad Certificante (AC). Este procedimiento que algunas veces es transparente al usuario, es el que en general proponen la mayoría de la empresas líderes en Certificación Digital y los documentos de trabajo más aceptados en la industria. Así mismo, es totalmente coherente con los requisitos establecidos por la Ley 25.506 y sus normas complementarias.

1. El solicitante o suscriptor crea, haciendo uso de alguna herramienta proveedora de servicios criptográficos, un par de claves encriptadas, pública y privada.
2. Una vez creado el par de claves, el solicitante genera una petición de certificado basada en la clave pública. La sintaxis detallada de esta petición o CSR está descripta por el Estándar PKCS#10. La petición contiene información sobre el suscriptor. En el caso de que éste sea un servidor habrá datos referentes al dominio, responsables y hosting del mismo.
3. El solicitante deberá entonces enviar la petición de certificado o CSR, junto con los documentos que prueben su identidad a una AC que re-

sulte confiable para los usos a los que estará determinado el Certificado.

4. La Autoridad Certificante, a través de su Autoridad de Registro posiblemente, cumplimentará los procedimientos establecidos para verificar la identidad del suscriptor.

5. Una vez cumplimentadas las verificaciones pertinentes, la Autoridad Certificante firmará y enviará al suscriptor o responsable del sitio su certificado digital.

6. Luego los suscriptores deberán instalar los Certificados en su browser o en su servidor (en el caso de un certificado SSL) y utilizarlos para manejar transacciones seguras.

Presentado este marco general, documentamos a continuación detalladamente, el procedimiento informático que se siguió para emitir los Certificados SSL y los Certificados de Referentes de la Guía:

Los Certificados utilizados en la implementación de sitio seguro para la Guía de Trámite y su interfase de administración son Certificados X.509 con Extensiones SSL firmados por la AC-URME y generados en formato PEM-encoded.

Su estructura y contenido, ajustadas a las especificaciones de un X.509 v3, a la recomendación RFC 3280 y al diseño preliminar de los certificados emitidos por la AC-URME son respectivamente:

Certificados de Servidor / SSL

TbsCertificate

SIGLA	Nombre del Campo ASN.1	Descripción	Contenido Certificado SSL de la Guía de Trámite	Contenido Certificado SSL de AdmTrámite
V	*version*	Versión del Certificado	V3	V3
SN	*serialNumber*	Número de Serie del Certificado	1555 F01A 4004 5A20	
AI	*signature*	Algoritmo de firma	sha1RSA	sha1RSA
CA	*issuer*	Expedidor / Emisor	C = AR O = Gobierno de Mendoza OU = Secretaria Administrativa Legal y Tecnica del Gobierno de Mendoza OU = Unidad de Reforma y Modernizacion del Estado CN = AC-URME Autoridad Certificante Unidad de Reforma y Modernizacion del Estado Gobierno de Mendoza	C = AR O = Gobierno de Mendoza OU = Secretaria Administrativa Legal y Tecnica del Gobierno de Mendoza OU = Unidad de Reforma y Modernizacion del Estado CN = AC-URME Autoridad Certificante Unidad de Reforma y Modernizacion del Estado Gobierno de Mendoza
T^A	*Validity*	Validez	Válido Desde: Martes, 28 de	Válido Desde: Martes, 28 de

			Octubre de 2003 14:06:42 Válido Hasta: Miércoles, 27 de Octubre de 2004 14:16:42	Octubre de 2003 14:06:42 Válido Hasta: Miércoles, 27 de Octubre de 2004 14:16:42
UI	*subject*	UniqueIdentifier. Sujeto / Asunto	C = AR DC = www.tramite.mendoza.gov.ar S = Mendoza L = Capital O = Gobierno de Mendoza OU = Secretaria Administrativa Legal y Tecnica OU = Unidad de Reforma y Modernizacion del Estado CN = www.tramite.mendoza.gov.ar	C = AR DC = www.admtramite.mendoza.gov.ar S = Mendoza L = Capital O = Gobierno de Mendoza OU = Secretaria Administrativa Legal y Tecnica OU = Unidad de Reforma y Modernizacion del Estado CN = www.tramite.mendoza.gov.ar
A	*subjectPublicKeyInfo*	Información de la clave pública del sujeto	Clave pública RSA (1024) bits	Clave pública RSA (1024) bits
	Extensions	Extensiones	No Críticas •Uso mejorado de Claves: Autenticación de Servidor •Identificador de clave del sujeto •Indetificador de clave de la Autoridad Certificante Extensiones	No Críticas •Uso mejorado de Claves: Autenticación de Servidor •Identificador de clave del sujeto •Indetificador de clave de la Autoridad Certificante Extensiones

			Críticas •Restricciones Básicas Tipo de asunto=Entidad final Restricción de longitud de ruta=9279348 •Uso de la Clave: Sin repudio(4000) •Punto de Distribución de la CRL: [1]Punto de distribución CRL Nombre del punto de distribución: Nombre completo: Dirección URL=http://gocref04.gober-nac.mendoza.gov.ar:8080/ejbca/webdist/certdist?cmd=crl	Críticas •Restricciones Básicas Tipo de asunto=Entidad final Restricción de longitud de ruta=9279348 •Uso de la Clave: Sin repudio(4000) •Punto de Distribución de la CRL: [1]Punto de distribución CRL Nombre del punto de distribución: Nombre completo: Dirección URL=http://gocref04.gobernac.mendoza.gov.ar:8080/ejbca/webdist/certdist?cmd=crl
signatureAlgorithm: Sha1RSA			Sha1RSA	Sha1RSA
SignatureValue:			D30B 4E8F 3EC3 5246 71AD 6B5A 09C0 52D3 1419 5719	

Certificado modelo de usuario final para los referentes de la Guía

TbsCertificate

SIGLA	Nombre del Campo ASN.1	Descripción	Contenido Certificado SSL de la Guía de Trámite
V	*versión*	Versión del Certificado	V3
SN	*serialNumber*	Número de Serie del Certificado	SN del referente
AI	*signature*	Algoritmo de firma	sha1RSA
CA	*issuer*	Expedidor / Emisor	C = AR O = Gobierno de Mendoza OU = Secretaria Administrativa Legal y Tecnica del Gobierno de Mendoza OU = Unidad de Reforma y Modernizacion del Estado CN = AC-URME Autoridad Certificante Unidad de Reforma y Modernizacion del Estado Gobierno de Mendoza
TA	*Validity*	Validez	Válido Desde: Martes, 28 de Octubre de 2003 14:06:42 Válido Hasta: Miércoles, 27 de Octubre de 2004 14:16:42
UI	*subject*	UniqueIdentifier. Sujeto / Asunto	C = AR S = Mendoza L = Localidad de referencia O = Gobierno de Mendoza OU= Cargo o función del referente- OU = *Oficina del Referente*

			OU= *Dependecia* OU = *Ministerio u Organismo* T= *Titulo del referente* CN = *Nombre y apellido del referente*
A	***subjectPublicKeyInfo***	Información de la clave pública del sujeto	Clave pública RSA (1024) bits
	Extensions	Extensiones	No Críticas •Uso mejorado de Claves: Autenticación del servidor Autenticación del cliente Correo seguro Seguridad IP del sistema final Seguridad IP del usuario de seguridad •Identificador de clave del sujeto •Indetificador de clave de la Autoridad Certificante Extensiones Críticas •Restricciones Básicas Tipo de sujeto = Entidad final Restricción de longitud de ruta •Uso de la Clave: Firma digital Sin repudio
signatureAlgorithm: Sha1RSA			Sha1RSA
SignatureValue:			Huella digital del referente

Para emitir los certificados de servidor de la Guía de Trámite y de su interfase de administración, se siguió el siguiente procedimiento informático:

1. Se crearon dos usuarios en la base de datos de la ACURME. Un usuario con alias ***tramite*** para el servidor de la Guía de Trámite y otro con el alias ***admtramite*** para su interfase de administración. Los usuarios creados en la AC-URME permiten identificar a una entidad final o un servidor y gestionar el CVS de los Certificados asociados a esta entidad. A cada usuario en la ACURME se pueden asociar solicitudes y certificados en distintos estados: a la espera de ser aprobado, generado, revocado, etc. Las líneas de comandos para completar esta acción en la ACURME fueron:

   ./ra.sh adduser tramite password "C=AR, S=Mendoza, L=Capital, O=Gobierno de Mendoza, OU=Unidad de Reforma y Modernización del EtaOU=Secretaría Administrativa, Legal y Técnica, CN=www.tramite.mendoza.gov.ar" null null 1 4 SERVIDOR Certificado de Servidor

./ra.sh adduser admtramite password "C=AR, S=Mendoza, L=Capital, O=Gobierno de Mendoza, OU=Unidad de Reforma y Modernización del EtaOU=Secretaría Administrativa, Legal y Técnica, CN=www.admtramite.mendoza.gov.ar" null null 1 4 SERVIDOR Certificado de Servidor

Estas líneas de comandos responden a la sintaxis general

RA adduser <username> <password> <dn> <subjectAltName> <email> <type> <token> [<certificateprofile>] [<endentityprofile>]

donde:

DN es el Distinguished Name

SubjectAltName es de la forma "rfc822Name=<email>, dNSName=<host name>, uri=<http://host.com/>". En nuestro caso, no utilizamos esta norma.

Email: email del solicitante

Type: Es una constante numérica que indica el tipo de entidad solicitante. En nuestro caso 1 corresponde a ENDUSER

Token: Indica formatos para el Keystore: User Generated=1; P12=2; JKS=3; PEM=4

Existing certificate profiles : Es una constante numérica que indica el perfil de certificado a emitir: En nuestro caso SERVIDOR indica el perfil de un certificado SSL con la estructura y extensiones que se han definido en la AC-URME por defecto para este perfil.

Existing endentity profiles : Define el perfil de la entidad que será certificada. En nuestro caso Certificado de Servidor indica que la entidad se ajusta al perfil de un servidor, tal como se ha parametrizado la AC-URME.

2. Una vez cumplimentado el proceso de validación de identidad de los solicitantes, se emitieron los certificados ejecutando el script ***batch.sh*** provisto por el sofware PKI EJBCA. Este script ejecuta una llamada a la clase ***se.anatom.ejbca.batch.BatchMakeP12*** con los parámetros que esta requiere. La clase ***BatchMakeP12***, genera el par de claves y las CSR (Requerimientos de certificación) para todos los usuario en estado ***NEW*** en la Base de Datos de la AC-URME. Alternativamente, se pueden pasar como parámetros a la clase: el nombre de un usuario particular al cual se le va a emitir un certificado y el formato en que este debe ser emitido (PEM, P12, JKS). El comando ejecutado para emitir ambos certificados en un solo paso fue:

./batch.sh -pem

3. Los Certificados emitidos en formato PEM encoded se almacenaron en un subdirectorio del repositorio de certificados de la AC-URME, para luego ser transferidos al Administrador del Servidor de la Guía de Trámites a los efectos de que el mismo los instalara en los directorios apropiados según la configuración del Web Server.

Nota: otra alternativa para la generación de Certificados contempla el uso de la herramienta java keytool. Sobre este modelo de generación se realizaron pruebas y se observó que no aportaba

Instalación de los Certificados SSL en Apache Mod_SSL

Para activar el soporte Apache SSL se habilitó el uso del módulo de seguridad mod_ssl en los archivos de configuración del web server, dando acceso https a través del puerto 443 para la Guía de Trámite, y del puerto 442 para admtramite.

El módulo mod_ssl es un módulo de seguridad para el Servidor Apache que usa las herramientas proporcionadas por el Proyecto OpenSSL para añadir una característica muy importante al Servidor HTTP — la habilidad de tener comunicaciones encriptadas. En contraste, usando HTTP normal, las comunicaciones entre el navegador y el servidor Web son enviadas en texto plano, lo cual puede ser interceptado y leído por alguna persona no autorizada.

El archivo de configuración mod_ssl está ubicado en */etc/httpd/conf.d/ssl.conf*. Para que este archivo sea cargado, y por ende para que mod_ssl funcione, se habilitó la sentencia *Include conf.d/*.conf* en */etc/httpd/conf/httpd.conf*. en el archivo de configuración Servidor Apache HTTP en Red Hat Linux 9.

Las siguientes modificaciones se realizaron para habilitar tal servicio:

1. Se copió en un directorio seguro del Servidor, exclusivamente dispuesto para tal fin, los archivos correspondientes a la clave pública y privada; y el certificado de cada uno de los sitios: www.tramite.mendoza.gov.ar y www.admtramite.mendoza.gov.ar. El certificado de la Guía de Trámite www.tramite.mendoza.gov.ar se tomó como certificado principal del servidor para garantizar sesiones de 128 bits con los navegadores.
2. Se copió en el directorio correspondiente el certificado Raíz de la AC-URME.
3. Se incluyó el siguiente código el archivo *ssl.conf*, uno de los archivos de configuración de Apache.

```
LoadModule ssl_module modules/mod_ssl.so
Listen 443
AddType application/x-x509-ca-cert .crt
AddType application/x-pkcs7-crl    .crl
SSLPassPhraseDialog  builtin
SSLSessionCache        dbm:/var/cache/mod_ssl/scache
SSLSessionCacheTimeout  300
SSLMutex  file:logs/ssl_mutex
SSLRandomSeed startup builtin
SSLRandomSeed connect builtin

##
## SSL Virtual Host Context
##

<VirtualHost _default_:443>
```

```
# General setup for the virtual host
DocumentRoot "/var/www/html"
ServerName www.tramite.mendoza.gov.ar:443
ServerAdmin lvenbenuto@mendoza.gov.ar
ErrorLog logs/ssl_error_log
TransferLog logs/ssl_access_log
SSLEngine on
SSLCipherSuite
ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EX
P:+eNULL
SSLCertificateFile /etc/httpd/conf/ssl.crt/cert-tramite.pem
SSLCertificateKeyFile /etc/httpd/conf/ssl.key/cert-tramite-
key.pem
SSLCertificateChainFile /etc/httpd/conf/ssl.crt/cert-CA.pem
SSLCARevocationPath /etc/httpd/conf/acurme.crl

<VirtualHost>
DocumentRoot "/var/www/html/sites/admtramite"
ServerName www.tramite.mendoza.gov.ar:442
ServerAdmin lvenbenuto@mendoza.gov.ar
ErrorLog logs/ssl_error_log
TransferLog logs/ssl_access_log
SSLEngine on
SSLCipherSuite
ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EX
P:+eNULL
SSLCertificateFile /etc/httpd/conf/ssl.crt/cert-admtramite.pem
SSLCertificateKeyFile /etc/httpd/conf/ssl.key/cert-admtramite-
key.pem
```

SSLCACertificateFile /etc/httpd/conf/ssl.crt/cert-CA.pem
SSLVerifyClient require
SSLVerifyDepth 10
</VirtualHost>

Donde:

- Archivo de Certificado: *cert-tramite.pem*
- Archivo de Clave del Certificado: *cert-tramite-key.pem*
- Archivo de Cadena o Path de Validación del Certificado: *cert-ca.pem*
- Registro de transacciones SSL: /logs/ssl_access_log;
  /logs/ssl_error_log;
- Nivel de registro SSL: *inf*

I) Desarrollo de un plan de pruebas e instrumentación

Se realizaron pruebas para evaluar la operatoria completa del Sitio Seguro en función del modelo de comportamiento esperado. En función de los resultados se realizaron los ajustes necesarios en la configuración del web Server y en el formato de los Certificados emitidos.

En particular, se evaluaron los siguientes puntos:

Certificados SSL	*Certificados X509 v1 o superior*	Se comprobó el funcionamiento del web Server con Certificados X509 v1 y superior.
	Dominios	Se comprobó la necesidad de usar el URL real de cada sitio como CN en el Certificado de servidor para que los browsers no acusen diferencias entre el nombre del sitio y el nombre certificado.
	Formato de certificados	Se realizaron pruebas de instala-

		ción en Apache de certificados codificados en distintos formatos: PEM, JKS, P12
Browsers	*Internet Explorer 5 o superior* *Netscape Communicator 4 o superior*	Se comprobó la instalación de certificados emitidos por la AC-URME y el soporte SSL a 128 bits.
	Cadena de certificación	Se comprobó el correcto reconocimiento de la cadena de certificación una vez instalado correctamente el Certificado RootCA en el browser del cliente.
Seguridad	*Seguridad de las claves*	Se realizaron todas las pruebas asociadas con la encripción y protección de las claves privadas de los suscriptores. Y la protección de los Certificados de Servidor en el web Server.
Cifrado	*Cifrado SSL 40 bits* *Cifrado SSl 128 bits*	Se comprobaron ambos esquemas de cifrado

Host Virtuales	*Configuración de Apache*	En las pruebas preliminares, se encontró que ***no es posible*** usar host virtuales basados en nombre con SSL, porque el 'handshake' de SSL (cuando el navegador acepta el certificado del servidor web seguro) tiene lugar antes de que se solicite una página en HTTP la cual identifica la apropiada máquina virtual basada en el nombre. Por lo tanto y ante la imposibilidad de contar con números IP reales para cada sitio; se decidió servir el sitio seguro de la interfase de administración de la guía www.admtramite.mendoza.gov.ar en el puerto 442.

J) Implementación efectiva de Sitio Seguro

Al 24/12/03 se han desarrollado y ejecutado todas las actividades que, de acuerdo con los puntos anteriores del presente informe, significaron la implementación efectiva del Sistema de Sitio Seguro en la Guía de Trámites del Gobierno de la Provincia de Mendoza.

K) Evaluación de Resultados

Toda evaluación necesita marcos de referencia y herramientas que sean a la vez sólidas, adaptables y útiles, para que permitan aprender de las experiencias de manera sistemática. Los resultados de la evaluación son útiles si ayudan a mejorar estas experiencias con conocimientos nuevos, y si ayudan en la toma de decisiones y la formulación de políticas relevantes.

Para el proyecto de firma digital y de acuerdo con el Plan Provincial Hacia el Gobierno Digital, la evaluación constituye un proceso continuo que se realiza desde el inicio de las acciones, y no algo que sucede sólo al final. Más que una auditoria, la concebimos como un proceso de aprendizaje en el que participan diversos grupos y personas implicadas. Esto permite tener en cuenta las perspectivas de diversos actores y sus percepciones, y contribuye a afinar o mejorar las actividades antes de que sea demasiado tarde.

Para que los resultados de la evaluación sean útiles, es necesario recogerlos y darlos a conocer de manera adecuada a los diferentes sectores involucrados. La divulgación efectiva de los resultados es parte del proceso de diseño inicial de la evaluación.

Resulta necesario identificar las variables e indicadores más apropiados, y preparar o adaptar los instrumentos que se pueden utilizar para recoger la información, combinando métodos cualitativos y cuantitativos de investigación.

Hemos definido una serie de indicadores que son de gran utilidad a la hora de recoger feedback crítico. Nos servirán para evaluar los resultados de la experiencia piloto de Sitio Seguro en el tiempo y para diseñar las acciones correctivas que, en función de estos, resulten necesarias de implementar.

Indicadores Críticos

Experiencia piloto Sitio Seguro

(Mediciones realizadas al 24/12/03)

Indicadores Cualitativos	**Métricas y Resultados**
Satisfacción de los usuarios: # Quejas y Reclamos	No se han registrado quejas por el sistema de Sitio Seguro
Marco legal: Documentación de la experiencia	Procedimientos de emisión y solicitud de certificados (En desarrollo) Política de certificación (En desarrollo)
Alcance:	Ministerio de gobierno

Participación de los sectores relacionados	Ministerio de Hacienda Ministerio de Economía Ministerio de Ambiente y Obras Públicas Instituto Provincial de la vivienda Ministerio de Justicia y Seguridad
Indicadores Cuantitativos	**Métricas y Resultados**
Eficiencia:	
% de certificados emitidos correctamente	100 % (7 personales y 1 de servidor)
# de fallas del sistema	0 (No se produjeron fallas)
# de interrupciones del servicio	0 (El servició estuvo disponible 365/7)
Asistencia:	
# de actores capacitados	7 (siete) referentes de la guía de trámites
# de asistencias otorgadas	8 (siete) Acciones de asistencia técnica
% de asistencias exitosas	100%
Uso del Sistema:	
% de utilización de servicios	95% de accesos con certificado
(sobre el total de referentes con Certificado)	5% de accesos login password
# de comunicaciones seguras establecidas	53
Acciones correctivas detectadas	**Acciones correctivas implementadas**
No se han detectado hasta la fecha	Ninguna

Calificación ponderada final

Implementación exitosa de la experiencia piloto

III. IMPLEMENTACIÓN DE UN PROTOTIPO DE PKI

A) Evaluación de herramientas de libre distribución

Como se ha propuesto en el Estudio de Factibilidad precedente, el diseño general de una Autoridad Certificante para la Administración Pública de la provincia de Mendoza, debe garantizar una arquitectura PKI técnicamente confiable y ajustada a los estándares nacionales e internacionales de modo de garantizar confiabilidad, confidencialidad, integridad y disponibilidad permanente.

Así mismo, se han propuesto como premisas fundamentales para el modelo, asegurar la escalabilidad y la interoperabilidad con la mayor cantidad de aplicaciones posibles.

El prototipo AC-URME entonces, debe contemplar una infraestructura de pequeña escala, con una Autoridad Certificante (CA) y una Autoridad de Registro (RA), pero que contemple un mecanismo de ***crecimiento gradual y ordenado*** ajustado a políticas y procedimientos unívocos o al menos con un alto grado de consistencia.

Los expuesto anteriormente orienta y condiciona la selección del software sobre el cual se estructurará el desarrollo e implementación del prototipo AC-URME, puesto que es necesario que el mismo sea lo suficientemente flexible para garantizar el ajuste del desarrollo a los estándares de la Industria, a los requisitos del marco legal existente en el país; y a las posibilidades de escala y complejidad crecientes en el desarrollo de aplicaciones y demandas de certificados.

En este sentido y tomando como base las propuestas vertidas en la Factibilidad Técnica realizada en el marco del estudio precedente, se ha seleccionado un conjunto relevante de características funcionales que el software PKI debería cumplir y se han evaluado comparativamente dos herramientas alternativas en relación a su cumplimiento de estas características.

Cabe mencionar que en la selección preliminar sólo se consideraron herramientas de libre distribución y en lo posible de código abierto, compatibles con la plataforma Linux, puesto que el carácter de prototipo y los alcances definidos para la AC-URME, no justificaban en principio, pensar en costosas herramientas PKI comerciales ni en altos costos de licenciamiento de software de base.

Se debe considerar también, que solo se incluyeron en la comparación aquellas tecnologías que fueran compatibles con los estándares sobre los que se basa la legislación nacional. Esto descarta, por ejemplo el uso de software vinculado a PGP.

La siguiente tabla presenta los resultados de esta evaluación comparativa. Se debe tener en cuenta que esta comparación representa las características funcionales que el producto puede soportar aún cuando algunas impliquen desarrollo de aplicaciones complementarios.

Referencias

✔ característica funcional disponible

✗ característica funcional no disponible

· característica funcional susceptible de ser desarrollada

característica no evaluada

Requerimiento Funcional	**Aspecto**	**Alt. 1 EJBCA**	**Alt. 2 OpenCA**
Seguridad de la clave privada de la CA	-Reinicio de todos los servicios de la PKI luego de una caída del servidor de la CA sin compromiso de la clave de administrador ni otras claves maestras.	✔	✗
	-Duplicado y recuperación, en caso de desastre, de la clave de la CA, con mecanismos de seguridad que no comprometan su confidencialidad	✔	✔
Par de claves	-Posibilidad de generación centralizada de claves, backup de las claves privadas y la recuperación distribuida de claves.	✔	·
	-Esquemas de pares de claves		

	duales -Posibilidad de que el administrador pueda especificar fechas de vencimiento independientes para la firma de la clave privada y la comprobación de la clave pública, de modo que las comprobaciones puedan tener éxito después de que la clave de firma expira.		
Protección de claves privadas	-Protección de las claves privadas de usuarios finales, servidores o dispositivos de red con por lo menos, un esquema basado en password. -Soporte a la autenticación mediante: passphrase o PIN y smart cards.	✔ ✔	✔ ✔
Gestión de Certificados	-Interfaz web para que los usuarios finales puedan enviar solicitudes de enrolamiento, solicitudes de renovación, solicitudes de revocación, descargar las CRLs, etc. -Administración automatizada que permita la autenticación y revocación transparente de usuarios o dispositivos, utilizando directamente sistemas administra-	✔	✔

	tivos o bases de datos preexistentes. -Aprobación manual de solicitudes de emisión de certificados. -Emisión de cerfiticados para SSL, S/MIME y Object signing. -Soporte a la inclusión de extensiones propias y personalizadas a los certificados emitidos por la CA, sobre un modelo de información básico. -Soporte a la emisión masiva de certificados. -Posibilidad de exportar e importar certificados y, alternativamente su par de claves asociadas como un mensaje cifrado, mediante contraseña proporcionada por un responsable.	✔ ✔ ✔ ✔ ✔	✔ ✔ ✔ ✔ ✔
Revocación de Certificados	-Interfaz web para que los usuarios finales puedan enviar solicitudes de revocación de sus certificados personales. -Actualización y emisión automática de la Lista de Certificados Revocados (CRL) inmediatamente después de que un certificado ha sido revocado, de modo de garantizar plena actualización del estado de los certificados. -Posibilidad de descarga de la	-	

	CRL por parte de los usuarios para incorporarla a sus aplicaciones y hacer validaciones de certificados revocados off-line.	✔	
	-Logs o seguimiento de la frecuencia de actualización de las CRLs.	✔	
	-Posibilidad de revocación de certificados expirados de modo tal que puedan ser incluidos en las listas para verificación de firmas históricas. Es decir posibilidad de incluir certificados expirados en las CRLs.		
	-Ante una revocación por compromiso de la clave, posibilidad de ingresar una fecha que indique la última fecha cierta en la que la clave se supo no comprometida, de modo tal que esta información pueda ser tenida en cuenta por el usuario ante una comprobación de validez del certificado.	✘	✘
	-Posibilidades de revocación manual (por parte del administrador) y automática.	✔	✔
	-Posibilidad de revocación masiva de certificados.		✔

Actualización de Clave y actualización de Certificados	-Interfaz web para que los usuarios finales puedan enviar solicitudes de renovación de sus certificados personales. -Actualización simultanea del par de claves junto al certificado, de modo tal de asegurar la rotación de claves. -Tiempo de vida de los certificados configurables de acuerdo a las políticas de seguridad que se definan.	 ✔ ✔	 ✔
Repositorio de certificados / Base de datos de la CA	-Chequeo de la integridad de datos que asegure el mantenimiento adecuado de los datos de enrolamiento de los usuarios. -Backup periódico de la base de datos de la CA fuera de horarios picos, con previo chequeo de la integridad de los datos que se resguardan. -Encripción de la base de datos para almacenamiento seguro.	✔ ✔ ✔	
Mecanismos de gestión del servicio de directorios	-Servicio de directorio que permita administrar automáticamente certificados y listas de certificados revocados en directorios compatibles con LDAP. -Publicación automática de los certificados emitidos en el servi-	✔	✔

	cio de directorio de la CA, de modo de asegurar la inmediata disponibilidad de los certificados para otros usuarios. -Función de recuperación del directorio en caso de fallo. -Integración de listas de certificados y listas de certificados revocados en directorios compatibles con LDAP. -Soporte a la comunicación con múltiples servidores LDAP, para balancear la carga de trabajo, garantizar redundancia y proveer escalabilidad.	✔ ✔	✔ ✔
Gestión de reportes y pistas de auditoría	-Generación auditable de la clave raíz. -Historial completo de cada clave generada. -Log de transacciones del o los administradores centrales. -Reporte de certificados emitidos a una fecha y con una fecha de caducidad determinada. -Logs y reportes exportables para ser integrados en otras aplicaciones mediante interfase ODBC o para ampliar facilidades de consulta (SQL query). -Otros logs de transacciones ta-	✔ ✔ ✔ ✔ ✔	✔ ✔ ✔ ✘ ✘

	les como: habilitación y baja de usuarios, recuperación de certificados, cambios de nombres (DN) y certificados pendientes de aprobación. -Posibilidad de programar (schedule) la generación de reportes. Logs o seguimiento de la frecuencia de actualización de las CRLs.	✔ ✔	✘
Configuración de políticas de acceso a la PKI	-Configuración de tiempo límite de login de un usuario a la PKI y sus servicios de usuario. -Configuración de la cantidad de intentos fallidos de login. -Configuración de intervalo de tiempo transcurrido antes de permitir un nuevo login.	✔	✔ ✔ ✔
Integración con aplicaciones comunes	Los certificados emitidos deben poder incorporarse y ser usados en las aplicaciones clientes típicas tales como: Outlook outlook express Internet Explorer	✔ ✔ ✔	✔ ✔ ✔

	Netscape communicator Adobe Acrobat Writer	✔	✔
. Condiciones de interoperabilidad	-Soporte a múltiples conjunto de caracteres para lenguajes internacionales. -Integración abierta con la mayor cantidad de aplicaciones y servicios de Internet, basada en los estándares del mercado, como X509 v3, LDAP, PKCS#7, PKCS#10, PKCS#12 y PKIX. -Soporte integral de todos los tipos de certificados estándar - SMIME -SSL -IPSec	✔ ✔ ✔ ✔	 ✔ ✔ ✔
Algoritmo de generación de par de claves	(2048 bits) RSA RSA/DSA (1024 bits)	✔ ✔	✔ ✔
Algoritmo de firma	-Md5withRSAEncryption -Sha1withDSAEncryption	✔ ✔	✔ ✔
Solicitudes de certificados	-PKCS#10	✔	✔
Formatos de representación de Certificados	-PEM -DER -JKS	✔ ✔ ✔	✔ ✔ ✘

	-PKCS#12	✔	✔
CRL	-X.509 v2	✔	✔
Otras tecnologías que pueden asociarse directamente para enriquecer el desarrollo de la AC		J2EE Java IAIK	OpenLDAP OpenSSL Apache Project Apache mod_ssl PHP C
Documentación y soporte	Documentación gratuita y en línea FAQs Banco de código Foro de discusión Bibliografía	 ✔ ✔ ✔ ✘	 ✔ ✔ ✔
Actualización de versiones		✔	✔

Conclusiones sobre la evaluación comparativa

El criterio general que se adoptó para seleccionar el software PKI, fue optar por aquella herramienta que soportara la mayor cantidad de características funcionales, con los mínimos requerimientos de desarrollos complementarios y con las mejores condiciones en cuanto a:

- simplificación de la complejidad de diseño y desarrollo, instalación y puesta a punto.
- portabilidad de código
- soporte de tecnologías asociadas
- interoperabilidad
- escalabilidad

Atentos a este criterio general, concluimos en la conveniencia de utilizar EJBCA, puesto que representa, dentro de las opciones evaluadas, la mejor opción para prototipar una Autoridad Certificante de pequeña escala como la descripta.

B) Explicitación del modelo PKI

De acuerdo con los conceptos plasmados en el estudio de factibilidad precedente a la continuidad de este proyecto, se consideró estratégico otorgarle al espectro de criptografía de clave pública la planificación de una estructura sistémica que posibilite su implementación a través de aplicaciones relacionadas y con una idea coherente de conjunto.

"Sólo una completa y adaptada implementación de una Infraestructura de Clave Pública (con un determinado sistema de hardware, de software, de políticas, de procedimientos y de personas) hace factible proporcionar el conjunto de seguridades informáticas que la Administración Pública Provincial necesita"

Es por eso que para el año 2004 consideramos la implementación de un prototipo de PKI atendiendo dos razones de corte estratégico:

- Sentar un importante precedente provincial que nos permita fundamentar un futuro licenciamiento de la Autoridad Certificante de la Unidad de Reforma y Modernización del Estado (en adelante AC-URME)
- Posicionarnos con una infraestructura propia que nos permita la provisión y gestión de certificados de forma totalmente independiente

Misión del protoripo

Securizar las transacciones electrónicas de la Administración Pública Provincial en un entorno de prueba, proveyendo claves y gestionando eficientemente certificados confiables, para lograr las preciadas garantías de autenticación, integridad, confidencialidad y no repudiación.

Objetivos

Nuestra definición de un Prototipo de Infraestructura de Clave Pública (AC-URME) de propósito general para la provincia de Mendoza sustenta los siguientes objetivos:

- Probar y determinar los alcances de la tecnología disponible

- Evaluar la implementación de una infraestructura propia de Firma Digital a través de un prototipo con alcances reales

- Posibilitar, desde una perspectiva administrativa y técnica, la utilización de servicios de firma digital de una variedad de aplicaciones piloto en la Administración Pública Provincial, atendiendo a nociones de eficiencia, optimización y despapelización del Estado

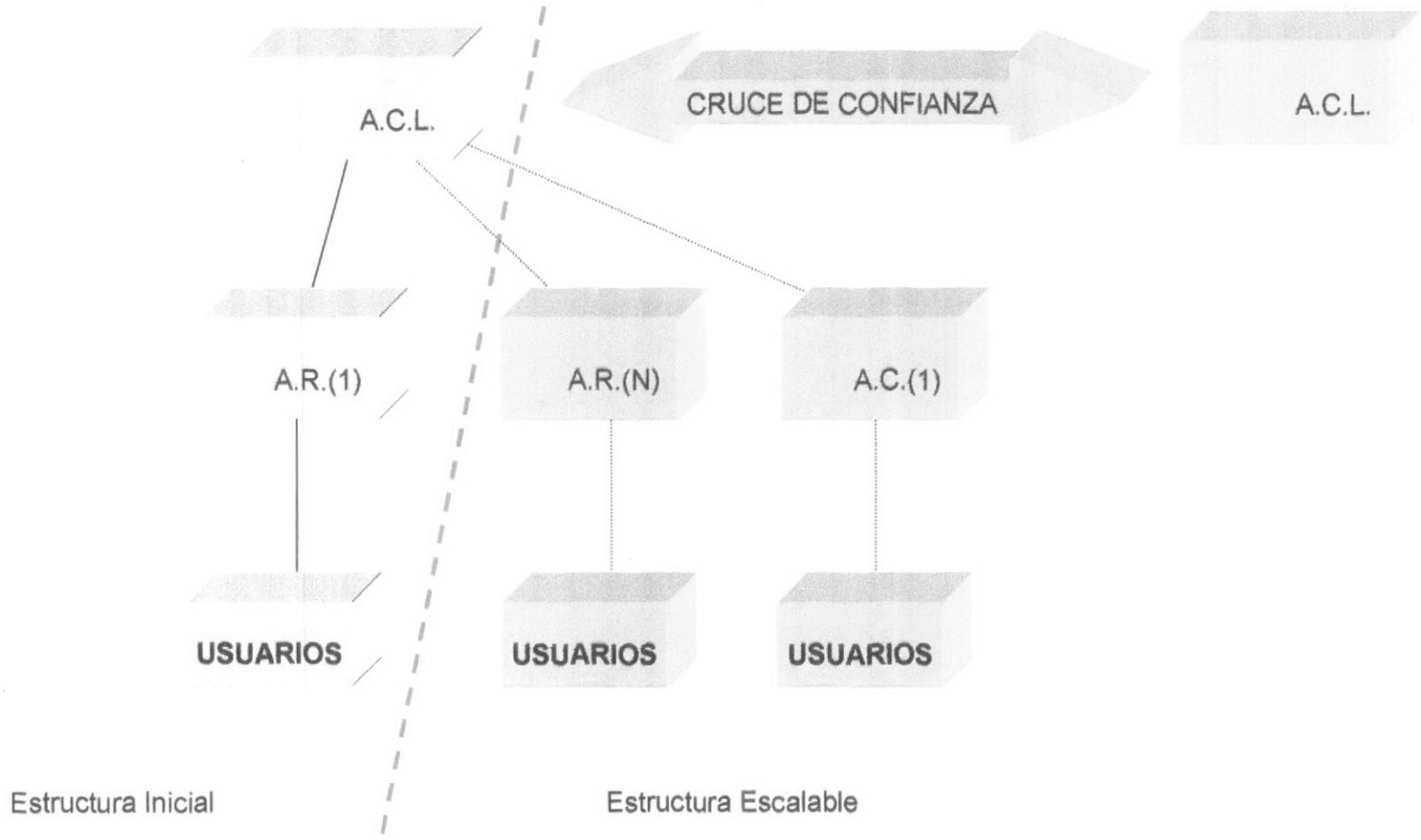
Estructura formal

Ha sido nuestro objetivo plasmar aquí la organización estructural que le daremos a nuestra implementación prototipo de la AC-URME. Es conveniente señalar que en su diseño se materializan las premisas planteadas en el estudio de factibilidad sobre condiciones de interoperabilidad y de escalabilidad realizadas en el proyecto antecedente.

Por consiguiente en la figura 1 se muestra tanto la estructuración inicial del prototipo, como también la tendencia ordenada y gradual de crecimiento planificad del mismo (Sombreado).

CONSEJO FEDERAL DE INVERSIONES
BIBLIOTE
Dr. Manu-
Belgran

Figura 1 – Estructura inicial del prototipo AC-URME y escalabilidad



Consejo Federal de Inversiones
Dr. Pablo Guillermo Tosi
Firma Digital

Componentes

Como vimos en la figura 1 la implementación inicial de la PKI Mendoza contempla la existencia de los siguientes entes en orden de jerarquía:

•Una Autoridad Certificante Licenciada (CA): Es el órgano responsable de la emisión de los certificados, luego de una correcta verificación por los métodos que considere en la política de certificación. Es el principal proveedor de la tecnología de criptografía asimétrica. Debe contar con medidas de seguridad que infundan la total confianza requerida para considerar a su gestión seria y exitosa y ostentar altos niveles de calidad en la prestación y disponibilidad de sus servicios.

La cualidad de "Licenciada" viene dada por la oportuna solicitud y obtención de la autorización por parte del Ente Administrador de firma digital una vez cumplidas las exigencias que, a la fecha del presente informe aún no han sido definidas. Sin embargo queremos dejar claro aquí, que sin perjuicio del funcionamiento piloto de la infraestructura, la intención es adherir al régimen de licenciamiento propuesto por ley.

Designación: se propone a la Gobernación por medio de su Secretaría Administrativa Legal y Técnica (UNIDAD DE REFORMA Y MODERNIZACIÓN DEL ESTADO)

•Una Autoridad de Registro (RA): Cuya misión es realizar meticulosamente la verificación de las personas (validación de identidad) que requieren la emisión de un certificado y realizar la solicitud formal pertinente (Registro de presentaciones). Gracias a esta descentralización se agiliza el proceso de certificación y se aumenta la eficacia en la gestión de solicitudes.

Designación: se propone que en un primer momento las funciones correspondientes a una (RA) sean llevadas en paralelo por el organismo encargado de certificar (C.A.L) hasta tanto la infraestructura se desarrolle y se identifiquen Auto-

ridades de Registro de acuerdo con los principios plasmados en su política y manuales de procedimiento.

•**Políticas de Certificación y Manuales de Procedimiento** que rigen el funcionamiento general de la PKI definiendo cuestiones tan esenciales como el tipo de certificado a emitir por la Autoridad de Certificación, el alcance de la información almacenada en el certificado, los procedimientos de registro, el tipo y alcance del compromiso de la Autoridad de Certificación con los usuarios y viceversa, las restricciones en el uso de los certificado, etc.

•**Suscriptores de certificados:** serán aquellos que se definan en las Políticas de Certificación particulares que se desarrollen en función de las aplicaciones piloto en las cuales se utilicen, como así también los servidores o equipos cuya identificación deba estar respaldada por un certificado de firma digital.

Modelo de Escalabilidad del prototipo

Como se puede ver en la figura 1 nuestra definición del prototipo posibilita en términos de escalabilidad y en función de requerimientos futuros:

•La incorporación de nuevas Autoridades de Registro (AR) que podrán tener funciones distribuidas por Ministerio o por Unidad de Gestión o alternativamente por tipo de certificados que se gestionen.

•La subordinación de eventuales Autoridades Certificantes que se ajusten a la Autoridad Certificante Licenciada y que posean una estructura orgánica consistente en términos de políticas, estándares y manuales de procedimientos. De ésta manera se puede favorecer los mecanismos de división de la carga del trabajo para garantizar la confiabilidad y flexibilidad de la PKI.

•La eventual interconexión de la jerarquía provincial con otras infraestructuras el país a través de cruces de confianza.

• El futuro licenciamiento de la Autoridad Certificante para dejar su condición de prototipo y obtener la plena validez de los certificados que emite y gestiona.

Alcance General de la Infraestructura

La infraestructura del prototipo propuesta pretende atender aquellas necesidades técnicas relacionadas con la firma digital y aquellas necesidades de apoyo y asesoramiento sobre tales temas a todos aquellos usuarios, funcionarios, agentes y dependencias del Poder Ejecutivo Provincial, dentro del marco de las experiencias piloto que se realicen.

Aplicaciones y Servicios

De acuerdo con la premisa de difundir y facilitar el uso de tecnología de firma digital así como también securizar las transacciones electrónicas se prevé que nuestro prototipo AC-URME desarrolle y experimente las siguientes prestaciones:

•Correo electrónico seguro/secure messaging, firma digital y no repudio. La naturaleza distribuida del correo electrónico y la necesidad de almacenar y reenviar información a muchos destinatarios encuentran en la criptografía de clave pública las capacidades de firma digital de mensajes y cifrado masivo sin establecimiento previo de claves secretas compartidas.

•Autenticación de identidad:

De Servidores (sitio seguro), para que los usuarios puedan comprobar el servidor con el que se comunican.

De clientes (control de acceso) para que los servidores puedan comprobar la identidad del cliente y en función de ésta tomar decisiones de control de acceso

•Canal Seguro (SSL): Confidencialidad en la transferencia de datos a través de enlaces públicos de Internet mediante protocolos de la capa de transporte.

•Secure Desktop: Cifrado de archivos (acuerdo de clave privada mediante clave pública) y cifrado masivo de datos (sin establecimiento previo de claves secretas compartidas).

•Secure e-forms: firma digital y seguridad para formularios basados en web.

•Encripción de bases de datos

Estándares Tecnológicos y Normas de Seguridad

El prototipo AC-URME adapta su diseño e implementación en función de las siguientes normas:

- **Resolución N° 54 / 99 y del Decreto-Acuerdo N° 1806 del 1999**, por el cual el Gobierno de la Provincia de Mendoza, a través del Comité de Información Pública (COM.I.P.) órgano dependiente de la Unidad de Reforma del Estado, adopta para el ámbito del Poder Ejecutivo Provincial el uso del COBIT (Objetivos de Control para la Información y Tecnología Relacionadas y sus posteriores actualizaciones)
- **Las Normas de Seguridad de Sistemas de Información**, sus posteriores modificaciones y agregados y fundamentalmente el uso de los **Es-**

tándares Tecnológicos de la Administración Pública Nacional (E.T.A.P y sus posteriores modificaciones) que fueron oportunamente desarrollados por la Subsecretaría de Tecnologías para el Sector Público, dependiente de la Secretaría Administrativa Legal y Técnica de la Gobernación de la Jefatura de Gabinete de Ministros,

- **Estándares Internacionales de Seguridad en Sistemas de Información**
- **Estándares sobre tecnología de Firma Digital** de vigencia provisoria dictados por la Secretaria de la Función Publica dependiente de la Jefatura de Gabinete de Ministros hasta tanto se aprueben las actualizaciones previstas por el Decreto Reglamentario 2628/2002 en su Art. 22.

L) Desarrollo del Prototipo AC-URME

1. EJBCA – SOFTWARE PKI

Los desarrollos de la AC-URME se estructuran sobre el software PKI EJBCA. EJBCA es básicamente una API java, de libre distribución y código abierto, mantenida y actualizada como un proyecto Sourcefoge www.sourceforge.net, que provee componentes para el desarrollo de todos los servicios básicos de una AC; con amplias posibilidades de escalabilidad puesto que puede ser utilizada como una aplicación standalone o integrada en cualquier aplicación J2EE.

Como se vio en la evaluación de herramientas alternativas, las principales características que justificaron el uso de esta herramienta:

- licencia Open Source (LGPL)
- construida sobre tecnología J2EE
- arquitectura basada en componentes flexibles que pueden ser incorporados o no en desarrollos particulares
- permite crear una jerarquía con múltiples niveles de CAs

- soporte al Enrroll individual o emisión batch de certificados
- soporte a certificados en formatos PKCS#12, PEM o DER
- APIs y herramientas diversas para el desarrollo de una interfase web de administración del CVS.
- GUI de administración web, asegurada a través de autenticación fuerte
- soporte a múltiples niveles de administradores con privilegios específicos y grupos de usuarios basados en perfiles
- profiles configurables para diferentes tipos de certificados y perfiles de usuario
- manejo de certificados bajos los estándares X509 y PKIX (RFC3280).
- manejo de CRLs (Certificate Revocation List) de acuerdo al estándar X509 v2
- generación programada y automática de CRL
- soporte a puntos de distribución de la CRL basada en URL de acuerdo a la recomendación RFC3280
- clases para notificación por email de nuevos usuarios agregados por la RA.
- almacenamiento de certificados y CRLs en múltiples motores de Bases de Datos
- soporte a la publicación LDAP de certificados y CRLs.
- soporte SCEP (Simple Certificate Enrollment Protocol)
- Soporte OCSP (Online Certificate Status Protocol)

2. DOCUMENTACIÓN DE INSTALACIÓN

Se documenta a continuación las tareas vinculadas con la instalación y configuración primaria de EJBCA en el servidor de prueba, su customización de acuerdo al diseño global de la AC-URME; y los primeros desarrollos realizados sobre este producto.

Esta documentación tiende a informar los aspectos más relevantes de la instalación, sin que sea posible realizar, en el contexto del presente informe, un abordaje pormenorizado de todos los detalles.

2.1. Plataforma instalada en el Servidor AC-URME

- Sistema Operativo: Linux Red Hat 8.0
- Base de Datos: PostgreSQL 7.2
- Aplication Server: JBOSS 3.2.0_Tomcat 4.1.24
- CA Autority: EJBCA 2.0.1

2.2. Instalaciones previas

Previo a la instalación del software de la CA Autority, se deben instalar los siguientes productos. Se documentan los puntos más relevantes de su configuración para un funcionamiento elemental; y las direcciones web desde donde se pueden descargar los productos y su documentación.

- ***Plataforma java - J2EE (Java II – Enterprise Edition)***. Como software de base a todo el desarrollo es necesario instalar la plataforma Java II. Se deberán configurar como mínimo las variables de entorno *JAVA_HOME, CLASSPATH* y *PATH*. www.javasoft.com / www.sunjava.com

- ***AplicationServer – JBOSS 3.2.1. con Tomcat 4.1.1.*** – EJBCA se desarrolla sobre el open source J2EE Aplication Server JBoss con un motor Servlet embebido, en nuestro caso Tomcat. JBoss debe descomprime en un directorio cuyo nombre no tenga espacios. Con su configuración inicial se puede levantar el servidor y testear su funcionamiento en modo localhost navegando la dirección http://127.0.0.1:8080 – Inicialmente da error porque no tiene ningún contexto configurado. Se debe configurar la variable de entorno *JBOSS_HOME* de modo que apunte al directorio raíz donde se instaló el AplicationServer. www.jboss.org

- ***Apache ant*** – Herramienta java utilizada para la compilación de las aplicaciones. Se debe configurar la variable de entorno *ANT_HOME* de modo que apunte al directorio donde se instaló la aplicación. Agregar ${ANT_HOME/bin} al paso del sistema. Puede testearse su correcto funcionamiento ejecutando el comando *prompt ant –help* en el prompt.

Otras dependencias que son incluidas junto al software son:

- ***Bouncycastle:*** Para sus servicios criptográficos y la creación de certificados y CRLs, EJBCA usa el open source JCE crypto provider de Bouncy Castle. Este software no requiere instalación particular por cuanto se provee junto a EJBCA como bcprovidk 14-1.19-.jar y bcmail-jdk14-1.19. La versión incluida en EJBCA 2.0 es 1.19. www.bouncycastle.org

- ***Log4J:*** Este producto provee el seguimiento de transacciones – logs de transacciones- sobre JBOSS. Es un proyecto de Apache Software Foundation . La versión provista por el release de EJBCA utilizado es la 1.2.7. www.ant.apache.org , www.jakarta.apache.org

- ***JUnit:*** Esta herramienta se utiliza para el desarrollo de test automatizados. EJBCA la utiliza para correr el runtest inicial de la instalación u otro tipo de pruebas particulares que pueden ser diseñadas. La versión utilizada es 3.7. y esta licenciada bajo una IBM Public License. www.junit.org

- ***OpenLDAP:*** Directorio LDAP de código abierto provisto por el grupo OpenLDAP. www.openladp.org

2.3. Intalación Primaria de la CA Autority – EJBCA 2.0.1

Se resumen a continuación los pasos básicos seguidos en la instalación primaria del software EJBCA.

1. Descomprimir el archivo ***ejbca2_0_tar.gz*** en un directorio.
2. Compilar la aplicación con la herramienta ***ant. Ant*** es un producto que opera sobre el archivo de configuración *build.xml* compilando las aplicaciones Java descriptas por este archivo.
3. Construir la documentación de EJBCA con ***ant javadoc***
4. Ejecutar ***ant deploy.*** Esta herramienta monta los servicios de la CA Authority sobre el AplicationServer JBOSS.
5. Ejecutar ***ant keystore*** esto copia el keystore primario (almacen de certificados de prueba) *(src/ca/keystore/server.p12)* al directorio $JBOSS_HOME/conf
6. Levantar los servicios del servidor JBOSS
7. Correr el test preliminar sobre la herramienta con el script ***runtest.sh***
8. Inicializar la CA para operación corriendo el script ***ca.sh init.*** Este script emite la primera CRL y ajusta algunos parámetro de operación inicial.

2.4 Configuración y puesta a punto

Establecemos ahora los pasos seguidos en la configuración específica de EJBCA según el diseño preliminar del prototipo AC-URME.

La customización puntal de esta herramienta y los desarrollos complementarios que se han realizado en el marco del Prototipo AC-URME, requieren un tratamiento exhaustivo que excede el alcance del presente informe. Por lo tanto se incluyen aquí solo aspectos de carácter general a modo de dar una guía orientadora de las actividades desarrolladas.

2.4.1. Configuración de la base de datos PostgreSQL 7.2

El funcionamiento de EJBCA ha sido testeado, según propone su documentación con las siguientes bases de datos:

- HypersoniqDB (usada por defecto en JBoss)
- PostegreSQL 7.2 (http://www.postgresql.org/)
- MySQL (http://www.mysql.com/)
- Oracle 8i (http://www.oracle.com/)
- Sybase

Luego de una evaluación de las alternativas posibles, se decidió utilizar el motor de base de datos *PostgreSQL* como repositorio de datos de la AC-URME. Esto implicó un ajuste en los archivos de configuración de EJBCA y de JBOSS, que originalmente trabajan con *HypersoniqDB* para que operen correctamente con Postgresql. Las tareas básicas implicadas en este ajuste son:

1. Instalar y configurar apropiadamente el motor PostgresSQL 7.2 www.postgress.org
2. Crear una base de datos, sin tablas, con un owner (propietario) y un password de usuario específico.
3. Configuración de EJBCA:
   - Renombrar el archivo *src/ca/META-INF/jbosscmp-jdbc-postgresql.xml* como *jbosscmp-jdbc.xml*
   - Realizar el mismo procedimiento con los archivos:

     src/ra/META-INF/jbosscmp-jdbc-postgresql.xml

     src/log/META-INF/jbosscmp-jdbc-postgresql.xml

     src/hardtoken/META-INF/jbosscmp-jdbc-postgresql.xml

     src/keyrecovery/META-INF/jbosscmp-jdbc-postgresql.xml
   - Reconstruir EJBCA con la herramienta *ant*
   - Montar los cambio sobre JBoss con el comando *ant deploy*
4. Configuración de JBOSS:
   - Borrar el archivo ***hsqldb-ds.xml*** de *JBOSS_Home/server/default/deploy/*. Esto remueve el uso de HypersoniqDB del contexto de EJBCA.
   - Poner el driver JDBC para postresql en el directorio JBOSS_Home/server/default/lib http://jdbc.postgresql.org/
   - Copiar el archivo *postgres-ds.xml* desde el directorio *JBOSS_HOME/docs/examples/jca* al directorio *JBOSS_HOME/server/default/deploy*. Cambiar en este archivo la en-

trada: *<jndi-name>PostgresDS</jndi-name>* por la entrada *<jndi-name>DefaultDS</jndi-name>*

- Editar el archivo *standardjbosscmp-jdbc.xml* cambiando el Typemapping por defecto de la siguiente manera:

  Type mapping:
  <datasource>java:/DefaultDS</datasource>
  <type-mapping>PostgreSQL 7.2</type-mapping>

- Arrancar JBOSS, depurar errores; y utilizar un cliente postreSQL para explorar la correcta construcción de las tablas de la AC en la base de datos.

2.4.2. Configuración el servicio de directorios LDAP

Para publicar certificados y CRLs en un directorio LDAP, se instaló en primera instancia el paquete OPEN LDAP www.openldap.org, elegido entre las alternativas posibles como el servicio de directorio a utilizar. Luego se realizaron los siguientes ajustes a los archivos de configuración:

1. Descomentar la sección para el session bean *'PublisherSession1'* en el archivo *ca/META-INF/ejb-jar.xml'*.

```
<! Descomentado para habilitar LDAP Publisher 20-11-2003
<session>
 <ejb-name>PublisherSession1</ejb-name>
 <home>se.anatom.ejbca.ca.store.IPublisherSessionHome</home>
 <remote>se.anatom.ejbca.ca.store.IPublisherSessionRemote</remote>
 <ejb-class>se.anatom.ejbca.ca.store.LDAPPublisherSessionBean</ejb-class>
 <session-type>Stateless</session-type>
 <transaction-type>Container</transaction-type>
```

```
    <env-entry>
      <description> host</description>
      <env-entry-name>gocref04.gobernac.mendoza.gov.ar</env-entry-name>
      <env-entry-type>java.lang.String</env-entry-type>
      <env-entry-value>localhost</env-entry-value>
    </env-entry>
  </session>
  END LDAP Publisher section -->
```

2. Descomentar y configurar la sección que define los permisos para el session bean anterior.

```
  <!-- Descomentado para habilitar LDAP Publisher 20-11-2003
   <method>
   <ejb-name>PublisherSession1</ejb-name>
   <method-name>*</method-name>
  </method>
  -->
```

3. Hacer lo mismo en el archivo *'ra/META-INF/ejb-jar.xml'*.

Un Publisher es un session bean que implementa la interfase *IPublishSession* usada para almacenar certificados y CRLS en repositorios LDAP. EJBCA soporta un número ilimitado de Publishers, que podrían habilitarse con solo agregar session beans (PublisherSession1, PublisherSession2, ...) sobre el servidor de aplicaciones.

Los parámetros configurados para el LDAP Publisher son:

- ldapHost es el host donde responde el servidor LDAP
- ldapPort es el puerto sobre el cual escucha el servidor LDAP
- loginDN es el DN de un usuario sobre el servidor LDAP con permisos para agregar y modificar entidades
- loginPassword es el password del usuario descrito en el punto anterior

2.4.3. Configuración de la Interfase de Administración WEB RA - ADMIN

La interfase de administración web permite administrar EJBCA remotamente mediante una conexión SSL (128 bits) con autenticación de Cliente. El Administrador de la AC-URME posee por tanto un Certificado de ***SuperAdmin*** que le permite acceder a esta herramienta.

Para habilitar este servicio debieron realizarse las siguientes tareas de configuración:

1. Editar el archivo *src/adminweb/WEB-INF/web.xml* cambiando la entrada *BASEURL* para reflejar el hostname de la AC-URME *gocref04.gobernac.mendoza.gov.ar*
2. Configuración del servidor de aplicaciones Tomcat:
   - Crear un usuario para el servidor de aplicaciones Tomcat en la CA Autority y emitir un certificado SSL para dicho usuario. Esto puede ser realizado con el script *batch.sh* o alternativamente con la herramienta *keytool*. Ambas formas permiten la generación del par de claves, la solicitud de certificación y la emisión del Certificado firmado por la CA.

     ./ra.sh adduser tomcat ****** "C=AR,O=Gobierno de Mendoza,CN=gocref04.gobernac.mendoza.gov.ar>" null null 1 3

Setear el password en clear text para el usuario Tomcat: ./ra.sh setclearpwd tomcat 12345

Generar un JKS-keystore para Tomcat: ./batch.sh tomcat -jks

- Editar el archivo */usr/jboss-3.2.0_tomcat-4.1.24/server/default/deploy/jbossweb-tomcat.sar/META-INF/jboss-service.xml* y setear la entrada '*keyStorePass*' de acuerdo al password del usuario Tomcat. Habilitar la conexión SSL en Tomcat con validación de cliente.
- Llamar al keystore generado '*.keystore*' y ponerlo en **$JBOSS_HOME**

cp ./p12/tomcat.jks $JBOSS_HOME/.keystore

- Descargar el certificado de rootCA en formato PEM desde http://gocref04.gobernac.mendoza.gov.ar:8080/ejbca/publicweb/webdist/cacert.jsp (llamarlo ejbca-ca.pem).
- Agregar el certificado de la rootCA al repositorio de certificados de confianza de la plataforma Java *$JAVA_HOME/jre/lib/security/cacerts*

keytool -import -trustcacerts -file ejbca-ca.pem -keystore $JAVA_HOME/jre/lib/security/cacerts -storepass changeit

3. Editar y customizar de acuerdo a parámetros de diseño el archivo de configuración *src/adminweb/WEB-INF/web.xml*
4. Crear el usuario SuperAdmin en la CA Autority con *CN=SuperAdmin* y el profile de *RAAdmin*. Emitir un Certificado en formato P12 para este usuario.

./ra.sh adduser raadmin ****** "C=AR, O=Gobierno de Mendoza, OU=Secretaria Administrativa Legal y Tecnica, OU=Unidad de Reforma y Modernizacion del Estado, CN=SuperAdmin" null null 65 1

./ra.sh setclearpwd raadmin 123456

./batch.sh

5. Incorporar el Certificado de SuperAdmin en el browser de la máquina cliente que usará el Administrador de la AC-URME desde http://localhost:8080/ejbca/publicweb/apply
6. Reiniciar *JBOSS* y testear el correcto funcionamiento de la web RA admin. Accediendo a https://gocref04.gobernac.mendoza.gov.ar:8443/ejbca

3. PUESTA EN MARCHA - INICIACIÓN PRIMARIA DE LA AC-URME

Hasta aquí se describió la instalación del software de base para la Autoridad Certificante y la configuración del mismo de acuerdo al conjunto de herramientas tecnológicas y servicios que se decidió utilizar para el desarrollo de la AC-URME.

Vamos a documentar ahora, los primeros pasos dados en la puesta en marcha de la AC-URME. Es decir, el procedimiento informático seguido para la creación de la CA y la RA de acuerdo al diseño PKI propuesto, la emisión del Certificado de la CA y la emisión de la primera CRL.

1. Crear el keystore de la CA raíz de la AC-URME, utilizando el script *ca.sh*

```
./ca.sh makeroot "C=AR, O=Gobierno de Mendoza, OU = Secretaria Administrativa Legal y Tecnica del Gobierno de Mendoza, OU = Unidad de Reforma y Modernizacion del Estado, CN = AC-URME Autoridad Certificante Unidad de Reforma y Modernizacion del Estado Gobierno de Mendoza" 1024 365 null $JBOSS_HOME/server/default/deploy/conf/server.p12 ******
```

El comando anterior crea la estructura para la CA Raíz de la PKI, su par de claves y su certificado. Se ha utilizado una longitud de clave de 1024 bits (RSA) y se ha dado validez de un año para el Certificado. El repositorio del Certificado es *server.p12*

Nota: Los nombres y directorios referenciados para los repositorios de certificados y otros datos se han dado a modo de ejemplo, pero no revelan nominaciones reales con el objetivo de preservar la seguridad de la AC-URME.

2. Editar el archivo de configuración *src/ca/ca/META-INF/ejb-jar.xml* para reflejar los valores asignados al Keystore y al KeyStorePass
3. Ejecutar el comando *ant keystore* para montar sobre JBOSS la nueva configuración.
4. Inicializar la CA luego de haber generado la clave del rootCA ejecutando el script *./ca.sh init*

PKIX requiere que la CRL esté siempre disponible aunque esté vacía. Por ello la CA debe ser inicializada conrriendo el script *ca.sh init* luego de que el root-CA ha sido creado.

Una vez inicializada, la Autoridad Certificante ya está en condiciones de emitir Certificados. A partir de este momento deben definirse aspectos del **diseño detallado** de la AC-URME, tales como los **perfiles de usuario** a Certificar, los **procedimientos y políticas de Certificación** y el formato e información general de **Certificado** que emitirá la CA.

La siguiente tabla describe sintéticamente el diseño de los certificados de usuarios finales y certificados SSL que emitirá la AC-URME con ajuste al estándar X509.

La sintaxis básica para un certificado X509 v3 consta de tres campos, cada uno de los cuales a su vez tiene su propia estructura tal como se describe a continuación.

tbsCertificate: Este campo incluye toda la información relacionada con el sujeto de la certificación y con la AC que emite el Certificado. El campo incluye la siguiente estructura...

SIGLA	Nombre del Campo ASN.1	Descripción	Resumen	Rango/Valores Alternativos que puede tomar según el estándar X509	Contenido para los certificados AC-URME
V	***version***	Versión del Certificado	Indica explícitamente el formato y el contenido permisible definido dentro de un certificado de una versión particular. En el caso de la AC-URME, los certificados emitidos se ajustan a la Versión 3.	V1 (0), v2 (1), v3 (2)	V3
SN	***serial-Number***	Número de Serie del Certificado	Es un número entero positivo que permite identificar de manera única a cada uno de los certificados expedidos por una AC. Este número lo genera la AC y puede ser un entero largo, siempre y cuando no supere los 20 octetos.	Entero positivo	Entero positivo
AI	***signature***	Algoritmo de firma	Identifica el tipo de función hash y el algoritmo de firma que se utilizó para firmar el certificado	Md5WithRSAEncryption Sha1RSA	sha1RSA
CA	***issuer***	Expedidor / Emisor	Identifica a la Autoridad Certificante que expidió y firmó el certificado	BIT String	C = AR O = Go-

					bierno de Mendoza OU = Secretaria Administrativa Legal y Tecnica del Gobierno de Mendoza OU = Unidad de Reforma y Modernizacion del Estado CN = ACURME Autoridad Certificante Unidad de Reforma y Modernizacion del Estado Gobierno de Mendoza
T^A	***Validity***	Validez	Define una fecha inicial y otra final entre las cuáles el certificado se puede considerar válido	Cualquier par de datos válidos en formato UTCTime o GeneralizadTime	Las prácticas de certificación de la ACURME preveen tiempos de validez de entre 3

					meses a 1 año dependiendo de la aplicación particular del certificado.
UI	***subject***	Uniquel dentifier. Sujeto / Asunto	Nombre del individuo o entidad que se va a identificar con el certificado que corresponde a la clave pública que se encuentra en el certificado. Nota: La AC-URME no admitirá subject en blanco. Osea que no se permiten subject con cadena vacía.	BIT String	C = AR DC = Domain Component. Nombre de Dominio, solo utilizado en certificadosde servidor S = Mendoza L = Localidad del suscriptor O = Gobierno de Mendoza OU= Cargo o funcion del suscriptor. Solo utilizado en certificados de usuario final OU = Ofici-

					na donde trabaja el suscriptor. Solo utilizado en certificados de usuario final OU = Dependencia para la cual trabaja el suscriptor o de la cual depende la responsabilidad por el mantenimiento del servidor OU=Ministerio/Organismo T=Tittle.Titulo del suscriptor. Sólo utilizado en certificados de usuario final. CN = Common Name. Nombre y Apellido

					del suscriptor o Nombre de dominio para servidores
A	***subject-PublicKeyInfo***	Información de la clave pública del sujeto	Contiene la clave pública del suscriptor que se va a certificar. Además contiene información que identifica el algoritmo para el cual se puede usar la clave pública	RSA, DSA, Diffie-Helman	Claves públicas RSA (1024) bits
UCA	***issuerUniquelD***	Identificador único del expedidor o emisor	Permite que el expedidor del certificado se identifique cuando el nombre de la entidad emisora pudiera ser ambigua o haberse reutilizado. Este es un campo opcional y solamente se puede usar si el número de la versión del certificado es v2 o v3.	BIT String	La RFC3280 recomienda sobre el estándar X509, que los nombres no sean reusados y que los certificados para Internet no hagan uso de Unique Identifiers. Por lo tanto la ACURME como AC que se ajusta a esta recomendación no hace uso

					de estos campos.
UA	***subjectUniquelD***	Identificador único del sujeto o suscriptor	Permite que el suscriptor del certificado se identifique cuando el nombre del sujeto pudiera ser reutilizado. Este es un campo opcional y solamente se puede usar si el número de la versión del certificado es v2 o v3.	BIT String	La RFC3280 recomienda sobre el estándar X509, que los nombres no sean reusados y que los certificados para Internet no hagan uso de Unique Identifiers. Por lo tanto la ACURME como AC que se ajusta a esta recomendación no hace uso de estos campos.
	Extensions	Extensiones	Las extensiones ***disponibles opcionalmente a partir de la versión 3***, permiten codificar información adicional en los certificados sin requerir modificaciones de formato del mismo. El apartado siguiente amplía sobre el uso de extensiones en la ACURME.	Extensiones estándar y no estándar. Críticas y no críticas según se definen en el estándar X509.	Extensiones incluidas en la RFC3280. La selección puntual se hará en la instancia de diseño detallado

					de la AC-URME para las clases de Certificados que se emitan y de acuerdo a consideraciones de interoperabilidad con distintas aplicaciones.

signatureAlgorithm: Este campo contiene el identificador del algoritmo criptográfico utilizado por la CA para firmar el certificado. Este campo ***DEBE*** contener el mismo Identificador de algoritmo que el campo ***Signature*** en el ***tbsCertificate***.

signatureValue: Este campo contiene la firma digital obtenida al firmar la información del tbsCertificate codificado en ASN.1 DER. Osea, el ASN.1 DER encoded tbsCertificate se usa como entrada a la función de firma. La firma obtenida se codifica como un BIT STRING y se incluye en este campo. Luego podemos utilizar esta Huella Digital, para que los usuarios verifiquen la autenticidad de este certificado. Esta huella varía dependiendo del algoritmo de firma utilizado.