

DECLARACIÓN DE SEGURIDAD

KEYONE 3.0



© Copyright 1999-2006 Safelayer Secure Communications, S.A. Todos los derechos reservados.

KeyOne 3.0 Declaración de Seguridad

Este documento es propiedad intelectual de Safelayer Secure Communications, S.A. Su contenido es confidencial y el acceso está restringido a personal de Safelayer Secure Communications, S.A.

No se autoriza la copia, reproducción o almacenamiento de parte alguna de este documento de ninguna manera o por ningún medio, electrónico, mecánico, por grabación, o de ninguna otra manera, sin el permiso de Safelayer Secure Communications, S.A.

Safelayer Secure Communications, S.A.

Teléfono: +34 93 508 80 90

Fax: +34 93 508 80 91

Web: www.safelayer.com

Email: support@safelayer.com

CONTENIDO

1 – Introducción	3
1.1 Identificación	3
1.2 Visión general	3
1.3 Conformidad	5
1.4 Convenciones	5
2 – Descripción del TOE	7
2.1 Descripción de la Confianza del Sistema KeyOne 3.0	8
2.1.1 Servicios Principales del TOE	8
2.1.2 Servicios Adicionales del TOE	11
2.1.3 Usuarios del TOE	13
2.1.4 Arquitectura Global	14
2.1.5 Arquitectura Lógica	15
2.1.6 Servicios Soportados	16
2.1.7 Arquitectura Física	18
2.2 Casos Útiles	22
3 – Entorno de Seguridad del TOE	25
3.1 Hipótesis de uso seguro	25
3.1.1 Personal	25
3.1.2 Conectividad	26
3.1.3 Físicos	27
3.2 Amenazas	27
3.2.1 Usuarios Autorizados	27
3.2.2 Sistema	27
3.2.3 Criptografía	28
3.2.4 Ataques Externos	28
3.3 Políticas de Seguridad Organizativa	29
4 – Objetivos de Seguridad	31
4.1 Objetivos de Seguridad para el TOE	31
4.1.1 Usuarios Autorizados	31
4.1.2 Sistema	31
4.1.3 Criptografía	31
4.1.4 Ataques Externos	32
4.2 Objetivos de Seguridad para el Entorno	32
4.2.1 Objetivos de seguridad no-IT del entorno	32
4.2.2 Objetivos de seguridad IT para el entorno	34
4.3 Objetivos de Seguridad tanto para el TOE como para el Entorno	35
5 – Requisitos de Seguridad de la IT	39
5.1 TOE Security Requirements	39
5.1.1 TOE Security Functional Requirements	39
5.1.2 Requerimientos Funcionales de Seguridad Extendidos del TOE	54
5.1.3 TOE Security Assurance Requirements	65
5.2 Security requirements for the IT environment	80
5.2.1 Security Functional Requirements for the IT environment	80
5.2.2 Requisitos de Seguridad Extendidos Propietarios para el entorno IT	95
5.2.3 Requisitos No IT de Seguridad Extendidos Propietarios para el entorno	95
5.2.4 Requisitos Funcionales de Seguridad Extendidos CIMC	96



6 – Especificación resumida del TOE.....	99
6.1 Funciones de seguridad del TOE.....	99
6.1.1 <i>Gestión de datos de auditoría</i>	99
6.1.2 <i>Base de datos segura</i>	111
6.1.3 <i>Gestión del Control de Acceso</i>	121
6.1.4 <i>Identificación y Autenticación</i>	133
6.1.5 <i>Comunicaciones seguras</i>	142
6.1.6 <i>Gestión de certificados</i>	157
6.1.7 <i>Almacén privado seguro</i>	166
6.1.8 <i>Gestión del archivo de claves</i>	169
6.1.9 <i>Copia de seguridad y recuperación</i>	170
6.2 Tabla de correspondencia entre requisitos funcionales y funciones de seguridad	172
6.3 Fortaleza de las funciones.....	177
6.3.1 <i>Mecanismos de autenticación</i>	177
6.3.2 <i>Módulos criptográficos</i>	177
6.4 Medidas de control.....	180
6.5 Funciones de seguridad que usan mecanismos probabilísticos o permutacionales	187
7 – Reivindicaciones	189
8 – Razonamiento	191
8.1 Security Objectives Rationale.....	191
8.1.1 <i>Security Objectives Coverage</i>	191
8.1.2 <i>Security Objectives Sufficiency</i>	195
8.2 Security Requirements Rationale.....	205
8.2.1 <i>Security Requirements Coverage</i>	205
8.2.2 <i>Security Requirements Sufficiency</i>	210
8.2.3 <i>Rationale for operations of Security Requirements</i>	216
8.3 Internal Consistency and Mutual Support	220
8.3.1 <i>Rationale that Dependencies are Satisfied</i>	220
8.3.2 <i>Rationale that Requirements are Mutually Supportive</i>	227
8.4 Rationale for Strength of Function	229
8.5 Assurance Requirements Rationale.....	230
8.5.1 <i>Rationale for CIMC security level 3</i>	230
8.5.2 <i>Rationale for EAL4</i>	231
8.6 Razonamiento para los requisitos de seguridad extendidos propietarios	232
8.6.1 <i>Requisitos de seguridad extendidos propietarios</i>	232
9 – Bibliografía, Definiciones y Acrónimos.....	235
9.1 Bibliografía	235
9.2 Definiciones.....	237
9.3 Acrónimos	240
Apéndice A – Consideraciones sobre el fichero de licencia	243

1 Introducción

1.1 Identificación

ID del Documento	7BFD7697 v1.9
Título	Declaración de Seguridad KeyOne 3.0
Fecha de Emisión	Diciembre 27, 2005
ID de la Release	3.0 04S2R1
Autores	Safelayer Secure Communications S.A..
Estado	Emitido
CC Version	2.2
Evaluated TOE	KeyOne 3.0 04S2R1: KeyOne CA, KeyOne LRA, KeyOne RA, KeyOne VA y KeyOne TSA Parches: 3.0_04S2R1_B01, 3.0_04S2R1_B02, 3.0_04S2R1_B03, 3.0_04S2R1_B04, 3.0_04S2R1_B05, 3.0_04S2R1_B06, 3.0_04S2R1_B07

Para cumplir con las garantías de seguridad EAL4+ del producto KeyOne incluido en esta Declaración de Seguridad, el fichero de licencia usado en el TOE no tiene que permitir la ejecución de *scripts* en modo inseguro (activación del flag `--unsecure`). Para más información sobre sobre el fichero de licencia, ver el Apéndice Consideraciones sobre el fichero de licencia, page 243.

1.2 Visión general

El propósito de este ST es especificar los requisitos funcionales y de certera seguridad implementados por TWS KeyOne 3,0 04S2R1, que es el Objetivo de la Evaluación.

El contenido del documento se organiza en los capítulos siguientes:

Capítulo 1, proporciona información etiquetada y descriptiva acerca del ST y del TOE al cual se refiere, un TOE resume de forma narrativa y como reivindicación de conformidad con los requisitos CC.



Capítulo 2, proporciona una descripción de los servicios del TOE, da una vista general de los usuarios de TOE que interactuarán con él, describe la disposición de las arquitecturas físicas y lógicas del sistema y la contribución de cada subsistema a los servicios identificados. Finalmente, una lista de los servicios de seguridades más comunes cubiertos por el TOE y las aplicaciones potenciales empresariales donde sería útil.

Capítulo 3, proporciona una definición del problema de la seguridad, mostrando las suposiciones de uso seguro, las ventajas, las amenazas, y las políticas de seguridad de la organización que deben ser sostenidas, protegidas, contrarrestadas y reforzadas por el TOE y su entorno operacional.

Capítulo 4, contiene la solución a este problema de la seguridad proporcionando los objetivos de seguridad para el TOE y para el entorno.

Capítulo 5, proporciona una traducción de los objetivos de la seguridad en un conjunto de requisitos funcionales y fiables en la manera de los requisitos CC parte 2, requisitos funcionales extendidos, CC parte 3 y los requisitos extendidos de confianza.

Capítulo 6, proporciona una explicación de cómo estos requisitos de seguridad se aplican en el TOE.

Capítulo 7, contienen las reivindicaciones de conformidad con el Perfil de Protección y estándares internacionales.

Capítulo 8, proporciona los objetivos de seguridad fundamentales evidenciando que el problema de seguridad se resuelve si todos los objetivos de la seguridad se alcanzan, y los requisitos de seguridad fundamentales mostrando el trazo efectivo entre ellos y los objetivos de la seguridad. Dos secciones más son incluidas para demostrar la consistencia y comprensión del todo el conjuntol.

Capítulo 9, indica la política de la seguridad impuesta por este objetivo de seguridad

Capítulo 10, incluye un glosario de términos utilizados dentro de este documento, una bibliografía aplicable al ámbito del ST, y una lista de Acrónimos.

El TOE definido por este ST proporciona los servicios siguientes de CSP:

- Registro de información del titular (Servicio de Registro)
- Generación de Certificados (Servicio de Generación de Certificados)
- Gestión de revocación de certificados (Servicio de Gestión de Revocación)
- Provisión de estado de revocación de los certificados (Servicio de Estado de Revocación)
- Funciones de Estampado de Tiempo (Servicio de Estampado de Tiempo)
- Producción de Creación de Firma/Creación de Firma Segura con Dispositivo (Servicio de provisión de dispositivo del titular)

1.3 Conformidad

KeyOne 3,0 opera conforme a la "Familia de Perfiles de Protección de las Componentes de Gestión y Emisión de Certificados" (CIMC) Perfil de protección nivel de seguridad 3 , versión 1,0, 31 de octubre de 2001, Agencia Nacional de la Seguridad (NSA). Adicionalmente KeyOne 3,0 cumple todos los Requisitos de Confianza para el nivel de certificación Criterios Comunes EAL4, ampliados con ALC_FLR.2.

En la construcción de los requisitos de seguridad se ha utilizado las siguientes referencias:

- a. Los requisitos funcionales de seguridad de la parte 2 de CC (Criterios Comunes para la Evaluación de la Seguridad de la Tecnología de la Información Parte 2: Requerimientos funcionales de seguridad, Versión 2,2, enero 2004).
- b. Los requisitos de confianza de seguridad de la parte 3 de CC extraídos del paquete EAL4 (Criterios Comunes para la Evaluación de la Seguridad de la Tecnología de la Información Parte 2: Requisitos de confianza de la seguridad, Versión 2,2, enero 2004).
- c. Los requisitos funcionales de seguridad extendida, expresado según la forma de CC, de "Familia de Perfiles de Protección de Componentes de Gestión y Emisión de Certificados" (CIMC) Perfil de Protección de Nivel de seguridad 3, versión 1,0, 31 de octubre de 2001, Agencia Nacional de la Seguridad (NSA).

1.4 Convenciones

El objetivo de esta sección es ayudar al lector a entender el uso del estilo específico y poner en claro las convenciones de información.

- Siempre que una operación ha sido aplicada a un requerimiento de seguridad funcional, el tipo de la operación será indicada, y el texto correspondiente aparecerá en cursiva. Tanto el tipo de la operación como el texto correspondiente se incluirán entre corchetes (por ejemplo, [selección: tipo de acontecimiento], [tarea: ningun atributo adicional]).
- Siempre que un requisito funcional de seguridad se haya utilizado más de una vez en este documento, el título del requisito funcional de seguridad es seguido por un número de iteración (por ejemplo, iteración 1) para distinguir entre las diferentes iteraciones del requisito funcional de seguridad.
- Las operaciones instanciadas del Perfil de la Protección de CIMC aparecerán en cursiva entre corchetes (por ejemplo, [*entorno 17*] proporcionará los registros de la auditoría de una manera conveniente para que el usuario interprete la información).

2 Descripción del TOE

El TOE incluido en este Objetivo de Seguridad ha sido diseñado e implementado para llevar a cabo las siguientes Directivas, Recomendaciones y Requerimientos:

- Directiva de la Comunidad Europea *1999/93/EC del Parlamento Europeo y del Concilio de 13 Diciembre 1999 en un marco común para firmas electrónicas*, 1999.
- CEN/ISSS *Workshop en Firmas Electrónicas. Requerimientos de Seguridad para la Confianza en Sistemas de Manejo de Certificados para Firmas Electrónicas*, Junio 2003.
- ETSI TS 101 456, *Requerimientos de Políticas para Autoridades Certificadoras Emisoras de Certificados Cualificados*.

No obstante, el objetivo de esta evaluación no es la comprobación de la conformidad de este producto contra las especificaciones¹ de seguridad previamente mencionadas.

Aunque [Eur99b] tiene una oratoria y aproximación general a las firmas electrónicas de cualquier tipo, la suposición subyacente en este documento es que las firmas electrónicas son creadas por medio de la criptografía de clave pública, que el titular usa un par de claves criptográficas consistente en una componente privada y una pública, y que un certificado producido por un sistema considerado en este documento esencialmente asocia la clave pública del titular a la identidad y posiblemente a otra información del titular por medio de una firma electrónica que es creada con la clave (clave de firma certificada) del TOE. Otras formas de firmas electrónicas están fuera del ámbito de este documento.

Aunque los requerimientos de seguridad para el Servicio de Suministro por Dispositivo de Titulares opcional, el cual abastece de subministro SCD/SSCD a los Titulares están incluidos en el ámbito de este ST, los requerimientos de los actuales dispositivos SSCD, como los usados por los Titulares del TOE, están fuera del ámbito de este documento. Los requerimientos de seguridad para SSCDs están facilitados en un documento separado Dispositivos de Creación de Firmas Seguras [CEN01b].

Siguiendo los principios de [Eur99b] este ST es tan tecnológicamente neutral como es posible. No se requiere o define ningún protocolo particular de comunicación o formato para las firmas electrónicas, certificados, listas de revocación de certificados, información de estado de los certificados y sellos de tiempo, excepto aquellos estándares internacionales que garantizan la interoperatividad global. Sólo

¹ Esta declaración es aplicable a cualquier referencia a las anteriores directivas de seguridad contenida en este Objetivo de Seguridad.



se asume ciertos tipos de información presentes en los certificados de acuerdo con el Anexo 1 de la Directiva Europea. Interoperatividad entre los sistemas TOE y los sistemas subscritos están fuera del ámbito de este documento.

2.1 Descripción de la Confianza del Sistema KeyOne 3.0

El KTS, con esta especificación, proporciona y maneja certificados usados para el soporte de firmas electrónica. Es una suposición elemental que el TOE usará una Infraestructura de Clave Pública (PKI) para la gestión de certificados. La aproximación adoptada de esta especificación es para que un TOE ofrezca un número de servicios, cada servicio dotado de funciones definidas para facilitar la seguridad en la entrega. Cada función definida es requerida para cumplir un estándar de seguridad mínimo y de ese modo conseguir un estado de confianza.

El TOE consiste en un número de subsistemas cada uno suministrando funcionalidades específicas del TOE. Aunque esta especificación considere requerimientos de seguridad de los subsistemas involucrados en los servicios del TOE, el objetivo es proveer al Titular y a la Parte de Confianza una única visión del TOE y de este modo una única visión de los subsistemas empleados por él. Para garantizar esto, la interficie del cliente, en esta especificación, es para el "Servicio del TOE" y no directamente para el servicio individual ofrecido por el TOE. Como subsistemas están mán descompuestos aún, cualquier funcionalidad, definida por otro estándar aceptable ha sido referenciada.

El TOE provee servicios desplegando subsistemas con Funcionalidad Principal y provee servicios opcionales desplegando subsistemas con Funcionalidad Suplementaria. El TOE implementa la Funcionalidad de Principal para cumplir con algunos requerimientos de [CEN01c]. El TOE también implementa algunos servicios opcionales, además de algunos servicios obligatorios, como los definidos por [CEN01c], y los requerimientos de seguridad especificados en [CEN01c] para ese servicio.

En efecto el TOE despliega subsistemas cumpliendo con los Requerimientos de Seguridad Principal y General. Es importante hacer notar que esta integración técnica/segura no impide necesariamente la libertad del TOE para ejecutar las diferentes componentes del servicio usando diferentes entidades empresariales.

2.1.1 Servicios Principales del TOE

Los servicios principales del TOE suministrados son:

Servicio de Registro: Verifica la identidad y, si es aplicable, cualquier atributo específico del Titular. Los resultados de este servicio se pasan al Servicio de Generación de Certificados.

- Petición de Certificados

La petición de Certificados es llevada a cabo por el Servicio de Registro después de que la identificación del Titular se haya dado siguiendo los requerimientos especificados en la Política de Certificado asociada.

- Gestión de los Datos del Titular

El Servicio de Registro por su naturaleza debe gestionar datos del titular entidad final. Los datos deben estar sometidos a varios requerimientos de protección de datos.

Servicio de Generación de Certificado: Crea y firma certificados basándose en la identidad y otros atributos del Titular como se verifica en el Servicio de Registro.

- Generación de Certificados

Tras recibir una petición de certificado del Servicio de Registro, KTSs generan un certificado usando la clave pública proporcionada. Esto garantiza que el CSP haya blindado la asociación entre la clave pública del Titular con su identidad. KTSs pueden también enviar su Claves Públicas de Infraestructura² o Control³ para ser certificadas por el Servicio de Generación de Certificados. Esto produce Certificados de Infraestructura o Control.

Siguiendo con la Generación de Certificados, los certificados son entregados al Titular directamente y adicionalmente pueden ser distribuidos via el Servicio de Disseminación de Certificados (publicación de certificados en un Directorio).

Los Certificados de Infraestructura y Control pueden ser suministrados directamente a la componente de confianza que requiera su uso.

- Renovación de Certificados

Durante el periodo previo a la expiración de un certificado, este periodo está definido en la política, el certificado puede ser renovado. La renovación del certificado consiste de la siguiente re-clave (una clave pública nueva se certifica utilizando la información de registro utilizada para generar el certificado previo) y el escenario de la renovación (la llave pública actual otra vez se certifica). La renovación del certificado cubre la Certificados de Infraestructura , Control y Titular.

- Copias de Seguridad de Claves Privadas

Las claves privadas del titular pueden ser guardadas como copia de seguridad por la Autoridad de Certificación. La recuperación de estas claves será controlada por un principio de multi-persona como se indicó en este documento.

Servicio de Gestión de Revocación: Procesan las peticiones e informes que relacionan la revocación para determinar la acción necesaria a tomar. Los resultados de este servicio se distribuyen por medio del Servicio de Estado de Revocación.

- Peticiones de Cambio de Estado de Certificados

²Las llaves de infraestructura son usadas por algunos componentes del TOE para procedimientos como autenticación en el subsistema, firma de entrada para la auditoría, transmisión encriptada, ...

³Las claves de control son usadas para gestión personal o uso de componentes del TOE, y pueden suministrar autenticación, firma o servicios confidenciales para estas interacciones personales con el sistema .



Dónde un Titular determina que su clave privada puede ser comprometida, una petición para la suspensión (revocación temporal) de su certificado es mandada a su KTS de CSP. Una petición correspondiente a restaurar un certificado de la suspensión al uso operacional puede ser hecha por el Titular.

Dónde el Titular sabe con toda seguridad que la clave privada se compromete, un pedido para la revocación de su certificado es mandado a su KTS de CSP.

El CSP puede solicitar también un cambio de estado de certificado vía este servicio. El estado de Certificados de Control e Infraestructura se puede controlar también por este servicio. Las peticiones para el cambio del estado de certificado son mensajes de autenticación y puede ser aceptadas o rechazadas por el CSP.

- Suspensión/Revocación de Certificados

El KTS al obtener una petición de suspensión o revocación vía este servicio cambia el estado del certificado a Suspendido o Revocado (Figura 2-1: mensaje A) en su Base de datos de Estado de Certificados, y de este modo es utilizado por el Servicio de la Posición de Revocación de CSP.

Servicio de Estado de Revocación: Proporciona información de estado de revocación de certificados a las partes de confianza. Este servicio se basa en información de estado de revocación que se actualiza en intervalos regulares.

- Datos de Estado de Revocación

El Servicio de Estado de Revocación proporciona información de estado de revocación de los certificados a las Partes de Confianza. El Servicio de Estado de Revocación refleja tantos cambios de estado de los certificados como cambios de estado solicitados por el Titular o por el CSP son procesados por el Servicio de la Gestión de Revocación. Estos datos pueden ser también accesibles a Titulares si la política requiere que los Titulares tengan acceso a datos de estado de revocación.

- Petición/Respuesta de Estado

Una Parte de Confianza que ha obtenido el/los certificado/s del Servicio de Diseminación de Certificados, requerido/s para la verificación de firma, necesita verificar el estado de estos certificados. El CSP proporciona un Servicio de Estado de Revocación para este propósito. En la arquitectura de sistema de KeyOne el Servicio de Estado de Revocación es un el servicio "en línea" que utiliza mensajería periódica entre el Servicio de Estado de Revocación y el Servicio de Gestión de Revocación.

En este servicio "en línea", un Parte de Confianza se comunica con este Servicio de Estado de Revocación y proporciona detalles del certificado/s para el que se requiere el estado. El Servicio de Estado de Revocación "en línea" cuando se usa mensajería periódica, pregunta a sus registros internos, que han sido actualizados por el último mensaje periódico. Una contestación se crea de este modo y es mandada a la Parte de Confianza indicando el estado del certificado/s solicitado/s.

Figura 2-1 muestra la relación entre el Servicio de Gestión de Revocación y el Servicio de Estado de Revocación. En la figura, el mensaje A actualiza la Base de datos de la

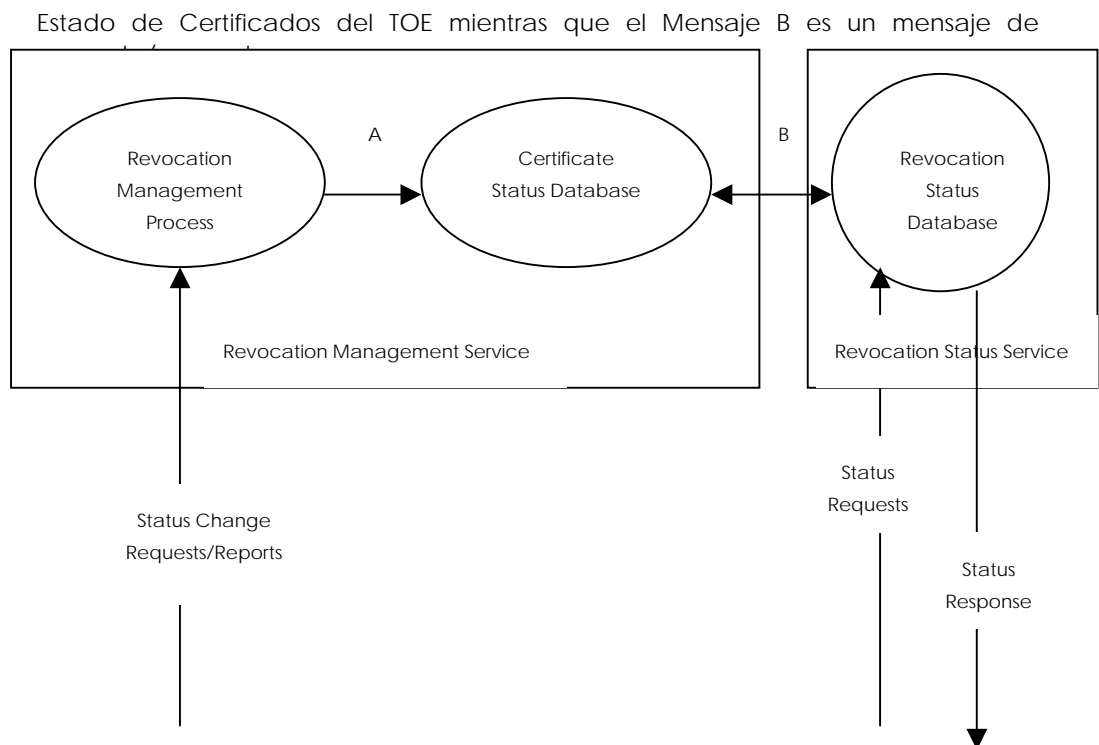


Figura 2-1. Envío de mensajes entre el Servicio de Gestión de Revocación y el Servicio de Estado de Revocación.

2.1.2 Servicios Adicionales del TOE

Identificado como optional en [CEN01c], KTS también provee los siguientes servicios adicionales:

Servicio de Provisión de Dispositivos de Titular: Prepara y proporciona un Dispositivo de Creación de Firma (SCD) a TitularesEs importante hacer notar que este servicio puede proporcionar un SCD y/o un SSCD. Dentro de este ST los requisitos de seguridad aplicables a SCDs son igualmente aplicable a SSCDs, donde SSCDs sigue los requisitos adicionales indicados en el Anexo III de [Eur99b].

- Preparación de SCD

El KTS de CSP prepara el SCD realizando la inicialización necesaria, formateo y creación de la estructura de archivos.

El KTS ordena al SCD que genere el par claves dentro del SCD.

- Provisión de SCD

La Provisión del SCD Provision es la distribución del SCD (después de la preparación) al Titular.

- Creación de Datos de Activación y Distribución



El SCD se protege con los (secretos) datos iniciales de activación para proteger el contenido del SCD. El CSP es responsable de la generación de estos datos iniciales de activación y la distribución segura subsiguiente de esto al titular.

Servicio de Sellos de Tiempo: Una tercera parte, de confianza para proporcionar un Servicio de Sello de Tiempo. El Servicio del Sello del Tiempo proporciona la prueba de que un artículo de datos existió antes de un cierto instante en el tiempo (la prueba de existencia). Si el artículo de datos ha sido firmado por el solicitante antes de ser sometido a la Autoridad del Sello del Tiempo (TSA), entonces el Servicio de Sello de Tiempo proporciona la prueba de que el artículo de datos existió y se firmó antes de un cierto instante en el tiempo.

- Comprobación de la Corrección de la Petición

Este componente se diseña para verificar la exactitud y la completitud de la petición. Si el resultado es positivo, el artículo de datos es enviado como entrada a la Generación del Sello del Tiempo.

- Generación del Parámetro del Tiempo

Este componente utiliza una fuente de confianza para entregar los parámetros exactos de tiempo. Estos parámetros se utilizan como entrada en el proceso de la Generación de Sello de Tiempo.

- Generación de Sello de Tiempo

Esta función es responsable de crear un sello de tiempo asociando el tiempo actual, un único serial, los datos proporcionaron para estampar de tiempo y asegurar cualquier requisito de la política asociada.

- *Token* del Sello del Tiempo (TST)

Este componente tiene la misión de computar el *token* de sello de tiempo que es devuelta al cliente. De manera efectiva, firma criptográficamente los datos proporcionados por la función de Generación de Sello de Tiempo.

El tiempo que estampa el servicio, asocia criptográficamente los valores de tiempo a valores de datos. La Figura 2-2 muestra un TSA conceptual que proporciona servicio de estampación de tiempo.

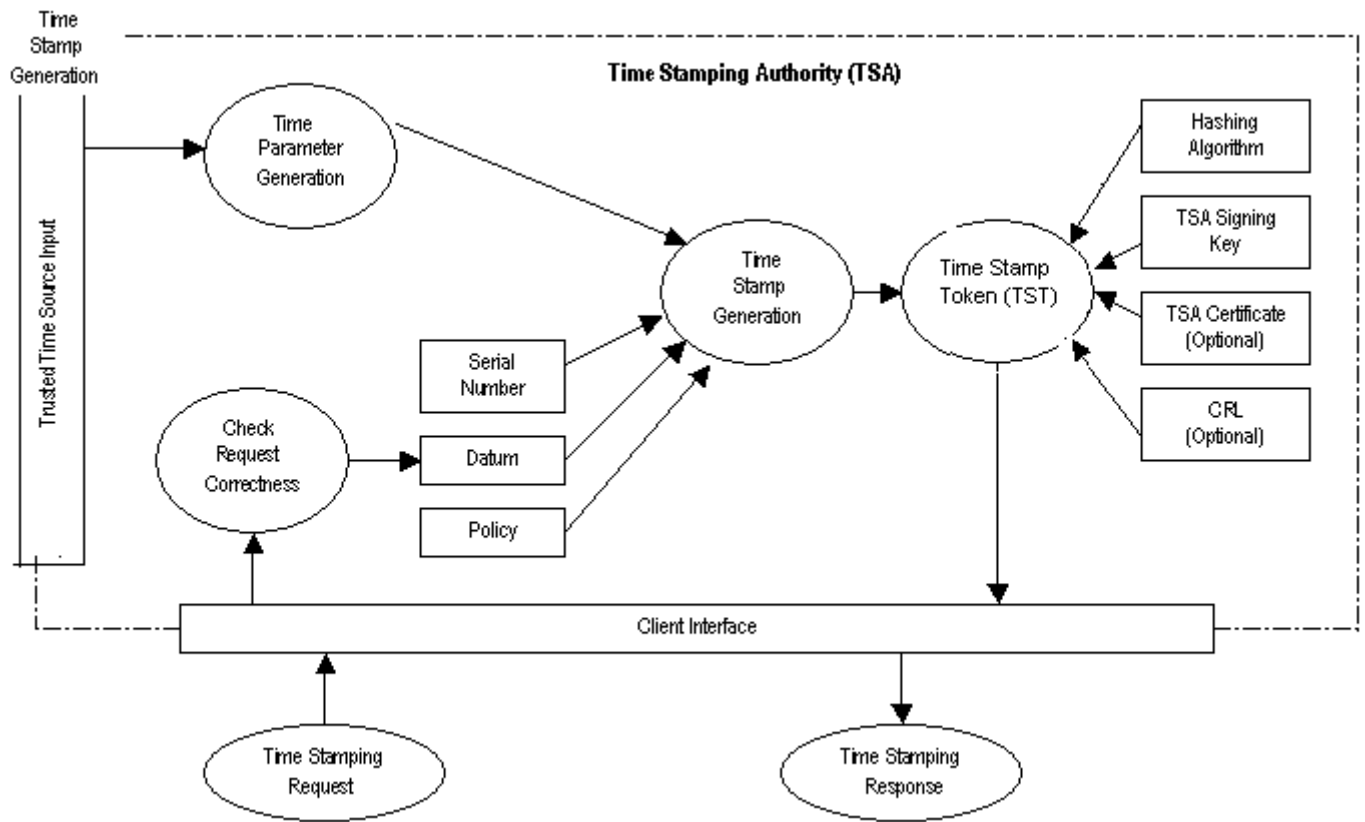


Figura 2-2. Servicios de Estampación de Tiempos

2.1.3 Usuarios del TOE

Los pretendidos usuarios de los servicios se clasifican en dos grupos principales:

2.1.3.1 Usuarios Externos

La Entidad de Certificado/ Usuario Final, es el sujeto del certificado que asocia su identidad con su clave pública. Hay otros tipos de entidades que se puede certificar, por ejemplo, aplicaciones, servicios,

Las Partes de Confianza, usuarios o agentes o cualquier servicio externo de confianza que confía en los datos de un certificado al hacer las decisiones, siguiendo los procesos de comprobación y limitaciones establecidos en las políticas de certificado para cada tipo de certificados publicados por KTS.

Los Auditores, que requieren acceder al registro de auditoría para evaluar y revisar las prácticas de certificación.

2.1.3.2 Usuarios Internos

Los Administradores del PKI, que pueden configurar y administrar las diferentes aplicaciones sostenidas por el TOE: Registro, Autoridad de Certificado, Autoridad de Validación, Autoridad de Estampado de Tiempo.

El Oficial de Registro es responsable de la operación de la Autoridad de Registro Ligera y de la Autoridad de Registro, según los procedimientos de registro establecidos

2.1.4 Arquitectura Global

Los Servicios Proveedores de Servicio de Certificación (CSP) se muestran en la Figura 2-3 , y pueden ser vistos para facilitar la producción y uso de una transacción firmada del Titular a una Parte de Confianza. Esta figura ilustra los sevicios junto con la interficie del TOE para sus Titulares y Partes de Confianza.

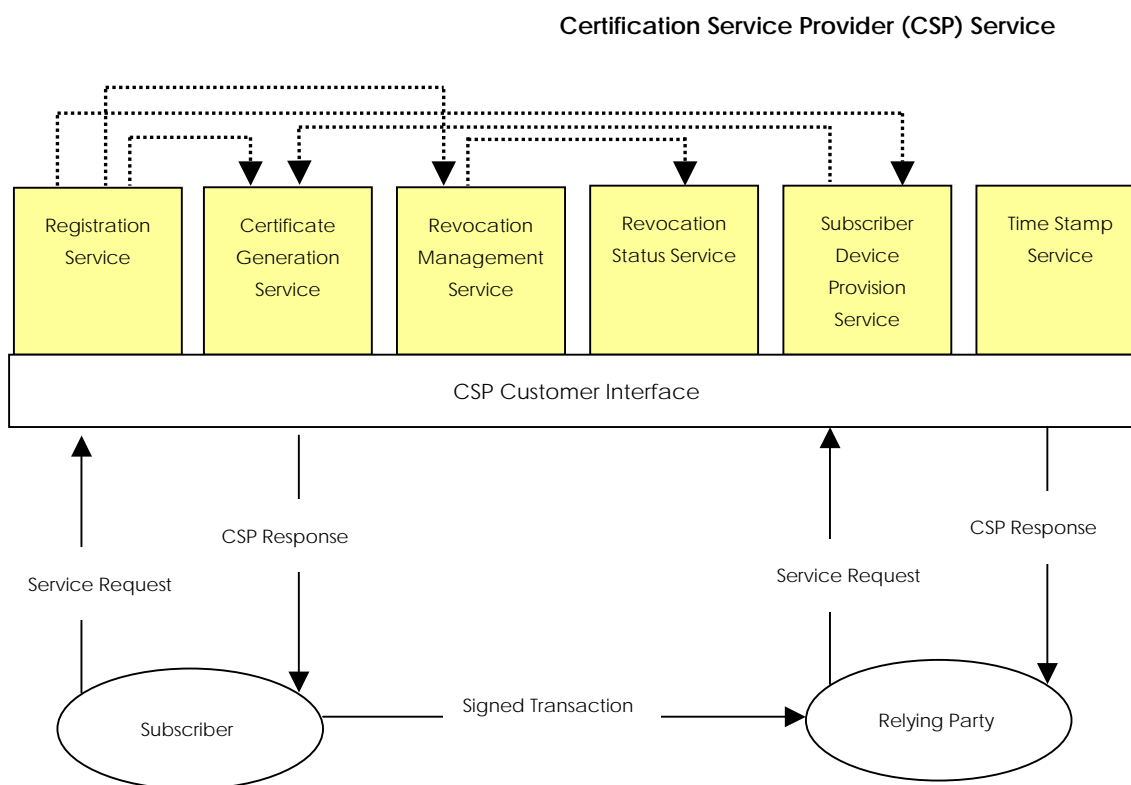


Figura 2-3. Servicios Proveedores de Servicio de Certificación (CSP)

Como se muestra, el TOE proporciona tanto el registro inicial como la generación de certificado. La gestión del ciclo vital primario del certificado (donde no existen estados de revocación o suspensión) es proporcionada por la medio de la Generación de Certificado y Registro. La gestión del ciclo vital secundario del certificado, donde existen estados excepcionales (por ejemplo. estados de revocación o suspensión) son proporcionados por la Gestión de Revocación y los

Servicios de Estado de Revocación. La Interficie del Cliente del TEO proporciona el acceso a los servicios del TEO para Titulares y Partes de Confianza.

2.1.5 Arquitectura Lógica

La arquitectura lógica de KTS se muestra en la siguiente figura:

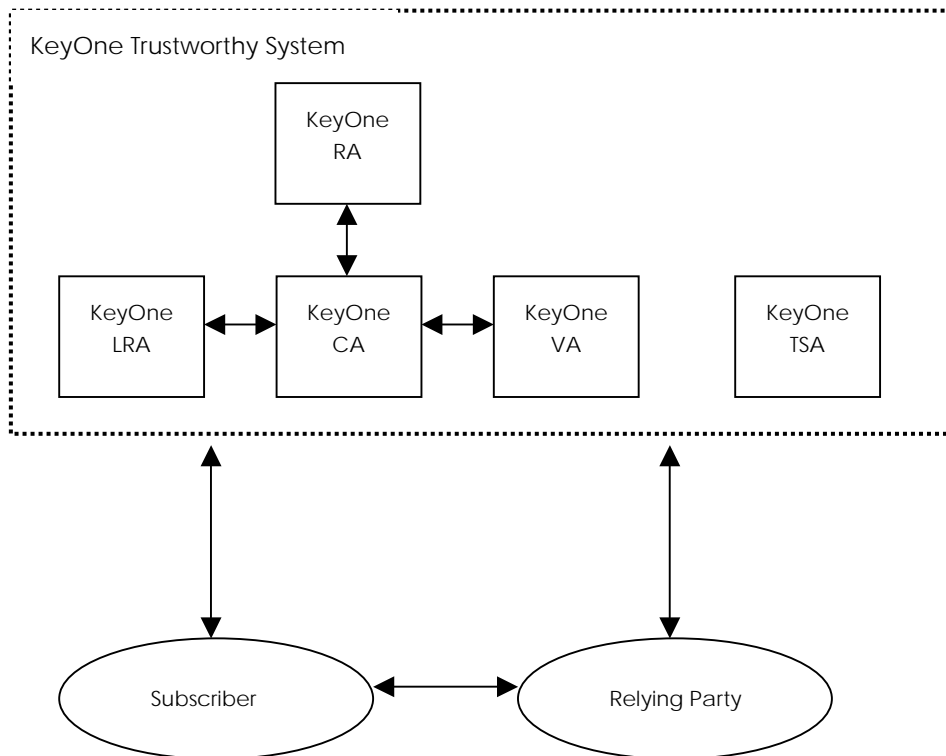


Figura 2-4. Arquitectura lógica de KTS

Así, el sistema KeyOne consiste de los siguientes elementos:

- KeyOne LRA. El servicio relacionado con este componente de KeyOne está explicado en la sección **Servicio de Registro** sección, página 8, y en **Servicio de Provisión de Dispositivos de Titular**, página 11
- KeyOne RA. El servicio relacionado con este componente de KeyOne está explicado en la sección **Servicio de Registro**, página 8.
- KeyOne CA. El servicio relacionado con este componente de KeyOne está explicado en la sección **Servicio de Generación de Certificado**, página 9 y en **Servicio de Gestión de Revocación**, página 9.
- KeyOne VA. El servicio relacionado con este componente de KeyOne está explicado en la sección **Servicio de Estado de Revocación**, página 10.
- KeyOne TSA. El servicio relacionado con este componente de KeyOne está explicado en la sección **Servicio de Sellos de Tiempo**, página 12.



2.1.6 Servicios Soportados

Esta tabla enumera los servicios soportados por el sistema, y los relaciona con los subsistemas de KeyOne donde residen

Subsistema	Servicios
KeyOne LRA	Servicio de Registro Servicio de Provisión de Dispositivo de Titular
KeyOne RA	Servicio de Registro
KeyOne CA	Servicio de Generación de Certificado Servicio de Gestión de Revocación
KeyOne VA	Servicio de Estado de Revocación
KeyOne TSA	Servicio de Estampado de Tiempo

Tabla 2-1. Servicios soportados por el sistema

Registro, Servicios de Generación de Certificados y Gestión de Revocación y Servicio de Provisión de Dispositivo del Titular usando la componente KeyOne LRA

Un titular del servicio de certificación hace una petición de certificación a la KeyOne LRA, que, después de verificar la identidad de titular, redirecciona la petición a KeyOne CA. Es realmente KeyOne CA quien entonces realiza la certificación solicitada y genera un mensaje de respuesta que es devuelto a KeyOne LRA. Una vez que el pretendido certificado se ha emitido, se incluye en el mensaje de la respuesta, para que KeyOne LRA pueda entregarlo al titular en un smartcard (SCD) usando el Servicio Provisión de Dispositivo del Titular. Este servicio prepara (generando el par de clave dentro del SC) y proporciona el SCD para que pueda ser entregado al titular.

Un titular del servicio de revocación hace una petición de suspensión, habilitación o revocación a KeyOne LRA, que, después de verificar la identidad de titular, redirecciona el pedido a KeyOne CA. Es realmente KeyOne CA quien realiza entonces el proceso solicitado de revocación o suspensión y genera un mensaje de respuesta que es devuelto a KeyOne LRA.

Registro, Servicios de Gestión de Ravocación y Generación de Certificados usando la componente KeyOne RA

Un titular del servicio de certificación (operario) hace una petición de certificación a KeyOne RA a partir de los datos de un titular de un certificado. Cuándo la petición de certificación se ha introducido, entonces otro titular del servicio de la certificación (aprobador) lo puede recuperar y después que verificar que la información contenida en el pedido es correcto, entonces él puede aprobar el pedido recuperado. Cuándo la petición se aprueba, es enviado a la KeyOne CA por un servidor interno de KeyOne. Es realmente KeyOne CA que entonces realiza la certificación solicitada y genera un mensaje de respuesta que es devuelto a KeyOne RA. Una vez que el presunto certificado se ha emitido, se incluye en el mensaje de respuesta, para que KeyOne RA lo pueda entregar al titular.



Un titular del servicio de revocación (operario) hace una petición de suspensión, habilitación o revocación a KeyOne RA. Cuando el pedido de revocación se ha introducido, entonces otro titular del servicio de certificación (aprobador) lo puede recuperar y después que verificar que la información contenida en la petición es correcta, entonces puede aprobar el pedido recuperado. Cuando el pedido se aprueba, es enviado a KeyOne CA. Es realmente KeyOne CA que realiza entonces el proceso solicitado de revocación o suspensión y genera un mensaje de respuesta que es devuelto a KeyOne RA.

Transacciones KeyOne LRA/KeyOne RA – KeyOne CA

El registro, la generación de certificados y servicios de revocación implican comunicación entre Keyone LRA/KeyOne RA y KeyOne CA. Los mensajes intercambiados durante este proceso de comunicación se llaman lotes y cumplen una sintaxis específica, que incluye una firma digital.

Además, estos mensajes se transfieren sobre una conexión de SSL. Por lo tanto, la confidencialidad, la autenticidad y la integridad de las transacciones Keyone LRA/KeyOne RA - KeyOne CA están garantizadas.

Los lotes se pueden clasificar en dos categorías, dependiendo del tipo de petición del que proceden:

- Lotes CR: Lotes que contienen una petición de certificación.
- Lotes RR: Lotes que contienen una revocación, la petición de suspensión o habilitación.

Servicio de Estado de Revocación

Un titular del servicio del estado de revocación envía una petición de OCSP a Servidor de KeyOne VA para determinar el estado de revocación de un cierto certificado. KeyOne VA, por su parte, genera un mensaje de respuesta de OCSP después de consultar su base de datos interna y lo envía de vuelta al titular, que debe proceder de acuerdo a la respuesta recibida. Además, tanto los mensajes entrantes como los mensajes salientes de OCSP se registran en la base de datos interna de la KeyOne VA, para que las auditorías posteriores sean posibles.

Servicio de Estampación de Tiempo

Un titular del servicio de estampación computa la impresión digital de algunos datos y emite una petición de sello de tiempo a la KeyOne TSA, según la sintaxis definida en RFC 3161. Entonces, KeyOne TSA obtiene el tiempo actual de un reloj de confianza y asocia la impresión digital de datos y el momento en que fue realizada la emisión del *token* de sello de tiempo firmado. Finalmente, el *token* de estampación de tiempo se encapsula en un mensaje de respuesta y se devuelve al titular.

El *token* del sello del tiempo emitido es una prueba de la existencia de los datos sobre los que se ha realizado un sello de tiempo, eso es, una evidencia inolvidable que los datos existieron antes de un cierto instante de tiempo. Tanto los mensajes entrantes como los mensajes salientes de OCSP se registran en la base de datos interna a la KeyOne TSA, para que las auditorías y comprobaciones posteriores sean posibles.



2.1.7 Arquitectura Física

La arquitectura física de KTS se muestra en la siguiente figura:

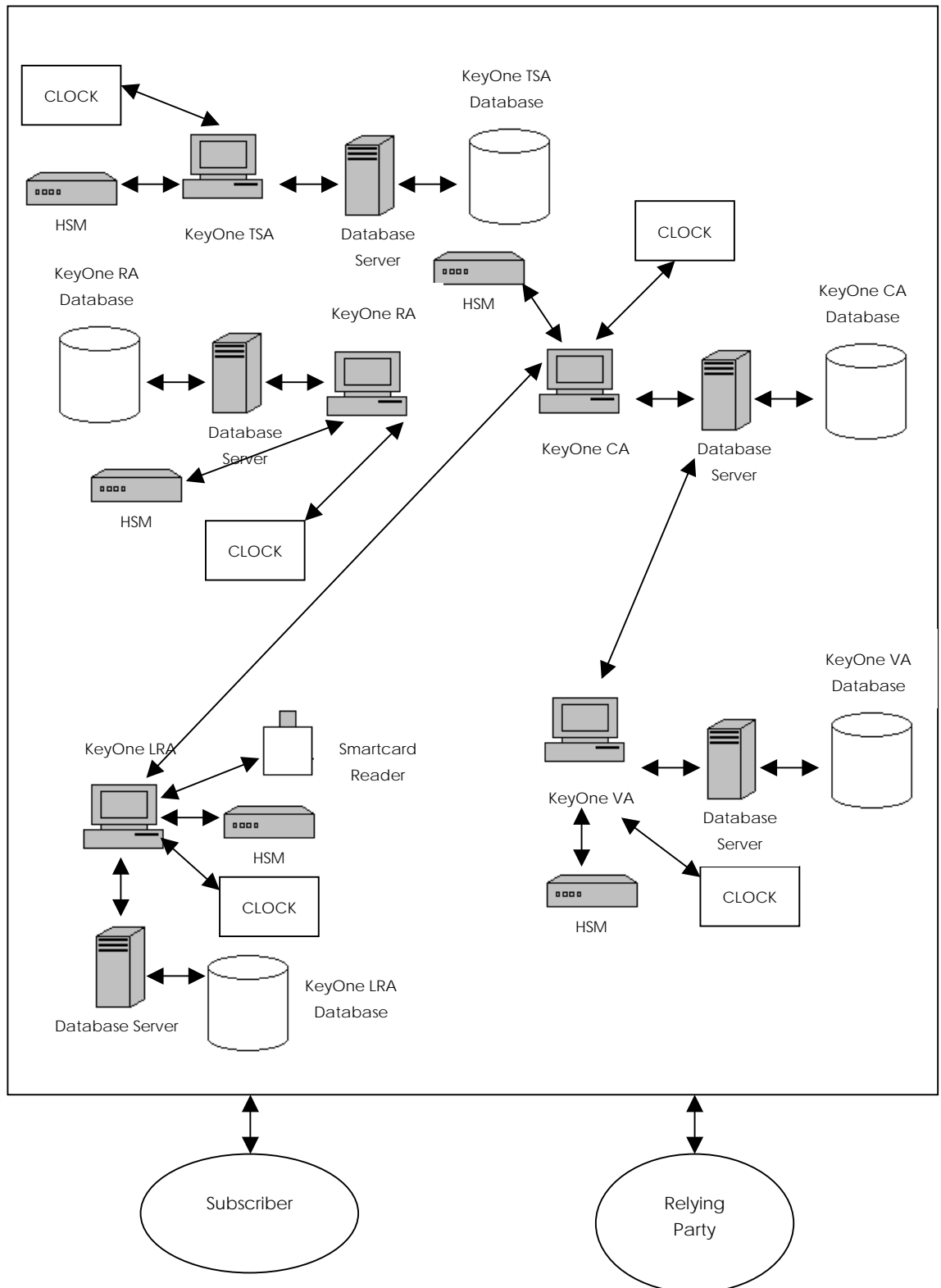


Figura 2-5. Arquitectura física de KTS



Esta figura muestra los componentes incluidos en la arquitectura física de KTS.

Todos los componentes de KeyOne están conectados a una Base de Datos donde la información relacionada al servicio que ese componente proporciona está almacenada. La base de datos relacionada con la componente KeyOne CA almacena certificados generados y CRLs, lotes de KeyOne (los lotes contienen los conjuntos de peticiones de certificación o revocación, o certificados, dependiendo de la entidad que los emite. El propósito principal de lotes utilizados en KeyOne es enviar peticiones de certificación o revocación y recibir las respuestas entre el RA y el CA) y los registros generados por el subsistema KeyOne CA. La base de datos relacionada con el componente KeyOne VA almacena el estado relativo a los certificados, los mensajes intercambiados con el componente KeyOne CertStatus (la parte del producto de KeyOne CA), y los registros generados por el subsistema de KeyOne VA. La base de datos relacionada con el componente de KeyOne TSA almacena las peticiones y respuestas de TSTs, y los registros generados por el subsistema de KeyOne TSA. La base de datos relacionada con el componente de KeyOne RA almacena certificados, lotes de KeyOne y registros generados por el subsistema de KeyOne RA. La base de datos relacionada con la componente KeyOne LRA almacena registros generados por el subsistema de KeyOne LRA.

Todos los componentes de KeyOne están conectados a un HSM (Módulo de Seguridad de Hardware) para generar y almacenar las claves relacionadas al servicio, y también ellos están conectados a un reloj que proporciona sellos fiables de tiempo para el uso del servicio.

En el componente KeyOne LRA, un usuario puede solicitar la generación de un certificado (la generación de las claves en un *smartcard*) o el cambio de estado de un certificado previamente generado. En este caso, un Operario de Registro verifica la identidad de la entidad solicitante, y aprueba o deniega la petición que firma con su certificado de firma almacenado en un *smartcard*. Cuando el Operario de Registro firma la petición, entonces se envía dentro de un lote KeyOne al Servidor de KeyOne CA; este servidor procesa la petición (cambia el estado del certificado en la base de datos de KeyOne CA o genera el certificado) y envía el resultado al Servidor de KeyOne LRA dentro de lote KeyOne. Si la petición implica la generación de un certificado, entonces se almacenará en el *smartcard* del titular

En el componente KeyOne RA, un operario puede solicitar la generación de un certificado o el cambio de estado de un certificado previamente generado. En este caso, la petición se introduce, y más tarde un operario aprobador verifica la petición introducida y aprueba o deniega la petición que firma con su firma. Cuando el Operario de Registro firma la petición, entonces se envía dentro de un lote KeyOne al Servidor de KeyOne CA; este servidor procesa la petición (cambia el estado del certificado en la base de datos de KeyOne CA o genera el certificado) y envía el resultado al Servidor de KeyOne RA dentro de un lote KeyOne

KeyOne CA (Servidor de KeyOne CertStatus) accede a la base de datos de KeyOne CA, donde la información sobre el estado de revocación de los certificados se almacena. De vez en cuando, KeyOne VA enviará peticiones a KeyOne CA (Servidor de KeyOne CertStatus) para obtener la lista de certificados que han cambiado su estado en el último lapso de tiempo. NDCCP (protocolo Near Domain Cert-Status Coverage) es un protocolo propietario de KeyOne, que se utiliza en la comunicación entre un módulo de Actualizador de la Base de Datos (en KeyOne VA) y un módulo Servidor de CertStatus (en KeyOne CA).



El Servidor de KeyOne VA (Autoridad de la Validación) aplica el Protocolo de Estado de Certificado "en Línea" (OCSP) y determina el estado actual de un certificado digital. Utilizando este servicio, una parte de confianza solicita a la Autoridad de Validación por el estado de un certificado.

Los componentes de entorno para cada uno de los componentes de KTS se listan en la siguiente tabla.

Subsistema	OS	Base de Datos	HSM	SCD/SSCD
KeyOne CA	Microsoft Windows 2000	Oracle 9i	nCipher nShield F3 Ultrasign SCSI, firmware version 2.0.5, Hardware version nC4032W	STARCOS SPK 2.4
KeyOne LRA	Microsoft Windows 2000	Oracle 9i	nCipher nShield F3 Ultrasign SCSI, firmware version 2.0.5, Hardware version nC4032W	STARCOS SPK 2.4
KeyOne RA	Microsoft Windows 2000	Oracle 9i	nCipher nShield F3 Ultrasign SCSI, firmware version 2.0.5, Hardware version nC4032W	STARCOS SPK 2.4
KeyOne VA	Microsoft Windows 2000	Oracle 9i	nCipher nShield F3 Ultrasign SCSI, firmware version 2.0.5, Hardware version nC4032W	STARCOS SPK 2.4
KeyOne TSA	Microsoft Windows 2000	Oracle 9i	nCipher nShield F3 Ultrasign SCSI, firmware version 2.0.5, Hardware version nC4032W	STARCOS SPK 2.4

Tabla 2-2. Componentes de Entorno

Adicionalmente, son necesarias las siguientes componentes:

- Cliente NTP instalado en el mismo *host* que los subsistemas KeyOne CA, KeyOne RA, KeyOne LRA, KeyOne TSA y KeyOne VA.
- Reloj fiable que obtiene obtains el Tiempo Universal Coordinado de una fuente fiable, y que sincroniza el reloj del sistema por medio del protocolo NTP, usando el cliente NTP instalado en la misma máquina que los subsistemas KeyOne CA, KeyOne RA, KeyOne LRA, KeyOne TSA y KeyOne VA.
- Windows 2000 Service Pack 4 para las componentes KeyOne CA, KeyOne LRA, KeyOne RA, KeyOne VA y KeyOne TSA.

2.2 Casos Útiles

La funcionalidad del TOE presentada en este documento puede resolver una gran variedad de casos de empresa, cumpliendo desde la identificación de usuario y control de acceso hasta los recursos internos, comercio electrónico, y muchos diversos sectores del mercado.



De todos modos, los servicios seguros que KTS provee pueden ser resumidos en los siguientes:

- 1 Autenticación de Usuario / Entidad / Aplicación
- 2 Encriptación de información
- 1 No-repudio e integridad, suministrada por avanzada firma.

Por lo tanto, cualquier negocio o aplicación que requiera cualquiera de los servicios de seguridad mencionados anteriormente, es capaz de utilizar un KTS.

El uso de este TOE es más conveniente a ciertos esquemas de registro. Esta configuración se adapta perfectamente a:

- Entornos de registro distribuido, debido al rápido y fácil despliegue de diferentes RAs, sin incrementar las necesidades de mantenimiento.
- Registros móviles o ambulantes, garantizando la seguridad del servicio de registro sin unas medidas de protección muy restrictivas físicamente.

3 Entorno de Seguridad del TOE

Esta sección incluye:

- Hipótesis de uso seguro,
- Amenazas y
- Políticas de Seguridad Organizativa

Esta información es el origen de los Objetivos de Seguridad especificados en la Sección 4, los requisitos funcionales de seguridad del OE y del entorno especificados en las Secciones 5 y 6 y los requerimientos de Garantía de Seguridad especificados en la Sección 8.

3.1 Hipótesis de uso seguro

Las hipótesis de uso se clasifican en tres categorías: personal (supuestos sobre los administradores y usuarios del sistema, así como cualquier agente que pueda suponer una amenaza), física (supuestos sobre el emplazamiento físico del OE o cualquier dispositivo periférico conectado), y conectividad (supuestos sobre otros sistemas de información necesarios para un funcionamiento seguro del OE).

3.1.1 Personal

A.Auditoría de Revisión de Logs

Es necesario recoger los registros de auditoría relativos a eventos de seguridad y éstos deben ser revisados por los Auditores.

A.Gestión de Datos de Autenticación

Se sigue una política de gestión de datos de autenticación que garantice que los usuarios actualizan sus datos de autenticación periódicamente y con los valores adecuados (ej. Longitudes de clave adecuada, históricos, variaciones, etc.) (Nota: esta hipótesis no aplica a datos de autenticación biométricos). An authentication

A.Auditores, Oficiales de Seguridad, Operadores y Administradores competentes



Se designarán Auditores, Oficiales de Seguridad, Operadores y Administradores competentes para gestionar el OE y la seguridad de la información que trata.

A.CPS

Todos los Auditores, Oficiales de Seguridad, Operadores y Administradores están familiarizados con la política de certificación (PC) y la declaración de prácticas de certificación (DPC), bajo las cuáles opera el OE. Esta documentación es conforme a la política de certificación de la Nato (NPKI).

A.Eliminación de los Datos de Autenticación

Siempre que se elimina un acceso se realiza un borrado seguro de los datos de autenticación y privilegios asociados (ej.: finalización de contrato de trabajo, cambio de responsabilidades).

A.Código malicioso sin firma

El código malicioso no está firmado por una entidad de confianza.

A.Notificación a las Autoridades de los temas de seguridad

Auditores, Oficiales de Seguridad, Operadores y Administradores y demás usuarios notifican a las Autoridades pertinentes de cualquier tema de seguridad que pueda tener impacto en sus sistemas, para minimizar posibles pérdidas o compromiso de los datos.

A.Educación en Ingeniería Social

Los Usuarios, Auditores, Oficiales de Seguridad, Operadores y Administradores son formados en técnicas para frustrar ataques de ingeniería social.

A.Usuarios coordinados

Los usuarios requieren un entorno informático seguro para llevar a cabo ciertas tareas o grupos de tareas. Los usuarios requieren acceso al mínimo de información del OE y se espera que actúen de forma coordinada.

3.1.2 Conectividad

A.Sistema Operativo

El sistema operativo seleccionado proporciona las funciones requeridas por este CIMC, para contrarrestar las amenazas detectadas para el nivel 3 del Perfil de Protección, tal y como se identifica en la Declaración de Seguridad (DS).

A.Cliente NTP

Todas las máquinas que componen el OE tienen instalado un cliente NTP para sincronizar sus relojes con el tiempo universal coordinado (UTC) proporcionado por un reloj fiable.

3.1.3 Físicos

A. Protección de las Comunicaciones

El sistema está adecuadamente protegido frente a pérdidas de la comunicación

A. Protección Física

El hardware, software y firmware del OE críticos para el cumplimiento de la política de seguridad será protegido frente a modificaciones físicas no autorizadas.

3.2 Amenazas

Las amenazas se clasifican en cuatro categorías: usuarios autorizados, sistema, criptografía y ataques externos.

3.2.1 Usuarios Autorizados

T. Errores de omisión administrativa

Los Auditores, Oficiales de Seguridad, Operadores y Administradores no realizan correctamente las funciones críticas para la seguridad..

T. Abuso de autorización del usuario que recopila y/o envía datos

El usuario abusa de las autorizaciones concedidas para recopilar y/o enviar datos sensibles o críticos para la seguridad

T. Error de usuario que provoca la inaccesibilidad de los datos

El usuario accidentalmente borra datos del usuario que los hace inaccesibles.

T. Auditores, Oficiales de Seguridad, Operadores y Administradores incurren en errores o acciones hostiles

Un Auditor, Oficial de Seguridad, Operador o Administrador Incurre en errores que cambian la política de seguridad del sistema o la aplicación o modifican intencionadamente la configuración del sistema para permitir que se produzcan violaciones de seguridad.

3.2.2 Sistema

T. Fallo de componente crítico para el sistema



El fallo de uno o más componentes del sistema provoca una pérdida de la funcionalidad crítica del sistema..

T.Ejecución de código dañino

Un usuario autorizado, sistema informático, o hacker se baja y ejecuta código dañino que dan lugar a procesos anormales que violan la integridad, disponibilidad o confidencialidad de los activos del sistema.

T.Modificación del contenido de un mensaje

Un hacker modifica la información que es interceptada por la línea de comunicación entre dos entidades autorizadas, antes de que llegue a su destinatario.

T.Código defectuoso

El desarrollador del sistema o aplicaciones entrega código que no está de acuerdo a las especificaciones o contiene fallos de seguridad.

3.2.3 Criptografía

T.Revelación de claves privadas y secretas

Una clave privada o secreta es revelada de forma inapropiada.

T.Modificación de claves privadas/secretas

Una clave privada/secreta es modificada.

T.El remitente niega el envío de información

El remitente de un mensaje niega el envío del mismo para evitar responsabilidades derivadas del envío del mensaje y acciones secundarias o falta de acciones.

3.2.4 Ataques Externos

T.Un hacker consigue el acceso

Un hacker se hace pasar por un usuario autorizado para realizar operaciones que serán atribuidas al usuario o proceso del sistema autorizado, o consigue acceso al sistema sin ser detectado debido a una falta, una vulnerabilidad y/o una implementación incorrecta del control de acceso ocasionando posibles violaciones de integridad, confidencialidad, o disponibilidad.

T.Acceso físico de un hacker

Un hacker interactúa directamente con el sistema para explotar vulnerabilidades del entorno físico, comprometiendo la seguridad a su capricho.

T.Ingeniería Social



Un hacker usa técnicas de ingeniería social para obtener información sobre la entrada al sistema, el sistema del usuario, diseño del sistema, u operación del sistema.

3.3 Políticas de Seguridad Organizativa

P.Uso autorizado de la Información

La información sólo podrá ser utilizada para propósitos autorizados.

P.Criptografía

Se utilizarán funciones criptográficas conformes con FIPS o recomendadas por el NIST, para todas las operaciones criptográficas.

4 Objetivos de Seguridad

Esta sección identifica y define los objetivos de la seguridad para el TOE y su entorno. Los objetivos de seguridad reflejan el objetivo establecido y se oponen a las amenazas identificadas, así como cumplen con las suposiciones y políticas de seguridad organizadas determinadas.

4.1 Objetivos de Seguridad para el TOE

Esta sección incluye los objetivos de seguridad para el TOE, divididos en cuatro categorías: usuarios autorizados, sistema, criptografía, y ataques externos.

4.1.1 Usuarios Autorizados

O.Certificados

El TSF debe cerciorarse de que los certificados, listas de revocación de certificados, e información de estado de los certificados son válidos.

4.1.2 Sistema

O.Preservación/recuperación fiable del estado seguro

Preserva el estado seguro del sistema cuando una componente segura falla y/o recupera el estado seguro.

O.Almacén suficiente de copias de seguridad y restauración efectiva

Proporciona suficientes copias de seguridad y restauración efectiva para garantizar que el sistema no pueda ser reproducido.

4.1.3 Criptografía

O.No-repudio



Previene de que los usuarios eludan su responsabilidad al enviar un mensaje proporcionando evidencias de que el usuario envió el mensaje.

4.1.4 Ataques Externos

O.Control del tráfico de comunicaciones de fuentes desconocidas

Control (por ejemplo., redirección o descarte) del tráfico de comunicaciones desde fuentes desconocidas para prevenir daños potenciales.

4.2 Objetivos de Seguridad para el Entorno

Esta sección especifica los objetivos de seguridad para el entorno.

4.2.1 Objetivos de seguridad no-IT del entorno

O.Documentación orientativa para Administradores, Operadores, Oficiales y Auditores

Impide errores de los Administradores, Operadores, Oficiales o Auditores proporcionando documentación adecuada acerca de la configuración segura y operando el CIMC.

O.Revisión por los Auditores de *Logs* de Auditoría

Identificación y custodia de los acontecimientos de seguridad más relevantes requiriendo que los auditores revisen los *logs* de auditoría con una frecuencia suficiente para localizar el nivel de riesgo.

O.Gestión de Datos de Autenticación

Asegura que los usuarios cambien sus datos de autenticación en intervalos apropiados y valores apropiados (por ejemplo, longitudes apropiadas, historias, variaciones, etc.) por medio de la gestión de datos de autenticación impuesta (Nota: este objetivo no es aplicable a datos biométricos de autenticación.)

O.Protección de Comunicaciones

Protege el sistema contra ataques físicos sobre la capacidad de comunicación proporcionando seguridad física adecuada.

O.Capacitados Administradores, Operadores, Oficiales y Auditores

Proporciona la gestión apta del TOE asignando a Administradores, Operarios, Oficiales y Auditores competentes para manejar el TOE y la seguridad de la información que contiene.

O.CPS

Todos los Administradores, Operarios, Oficiales y Auditores conocerán la política de certificado (CP) y la declaración práctica de certificación (CPS) bajo que la TOE opera.

O.Eliminación de los Datos de Autenticación

Proporciona la eliminación apropiada de los datos de autenticación y privilegios asociados después de que el acceso se ha eliminado (por ejemplo, la terminación del trabajo, el cambio en la responsabilidad).

O.Instalación

Aquellos responsables del TOE deben asegurar que el TOE sea entregado, instalado, gestionado, y operado de un modo que mantenga la seguridad.

O.Código Maligno No Firmado

Protege el TOE del código maligno asegurando que todo el código esté firmado por una entidad fiable antes de cargalo en el sistema.

O.Advierte a las Autoridades sobre cuestiones de Seguridad

Notifica a las autoridades apropiadas sobre cualquier cuestión de seguridad que impacte en su sistema para minimizar la potencial pérdida o compromiso de los datos.

O.Protección Física

Aquellos responsables del TOE deben asegurar que los componentes más relevantes para la seguridad del TOE se protejan del ataque físico que pueda comprometer la seguridad IT.

O.Aprendizaje de Ingeniería Social

Proporciona aprendizaje para los usuarios generales, Administradores, Operadores, Oficiales y Auditores en técnicas para frustrar los ataques de ingeniería social.

O.Usuarios Cooperativos

Asegure que los usuarios sean cooperativos para que puedan realizar alguna tarea o grupo de tareas que requieran un entorno IT seguro e información gestionada por el TOE.

O.Seguridad del ciclo de vida



Proporciona instrumentos y técnicas utilizadas durante la fase del desarrollo para asegurar que la seguridad se diseña dentro del CIMC. Detecta y resuelve los desperfectos durante la fase operacional.

O.Repara los desperfectos de seguridad identificados

El vendedor repara los desperfectos de la seguridad que han sido identificados por un usuario.

4.2.2 Objetivos de seguridad IT para el entorno

O.Funciones Criptográficas

El TOE debe aplicar los algoritmos cifrados aprobados para la codificación/decodificación, autenticación, y generación/verificación de firmas; las técnicas de generación de claves aprobadas y utilizar módulos criptográficos validados. (Validado se definido como validado FIPS 140-2.)

O.sistema Operativo

El sistema operativo utilizado se valida para proporcionar la seguridad adecuada, inclusive la separación del dominio y el no-desvío, de acuerdo con los requisitos de la seguridad recomendados por el Instituto Nacional de Estándares y Tecnología.

O.Integridad verificada periodicamente

Proporciona comprobaciones de integridad periódicas tanto en el sistema como en el software.

O.Roles de seguridad

Mantiene los *roles* relevantes para la seguridad y la asociación de usuarios con esos *roles*.

O.Función de validación de seguridad

Garantiza que el software, hardware, and firmware relevantes para la seguridad funcionen correctamente por procedimientos y características.

O.Ruta de confianza

Proporciona una ruta fiable entre el usuario y el sistema. Proporciona una ruta fiable para los datos relevantes a la seguridad (TSF) en los que ambos puntos finales han sido identificados firmemente.

4.3 Objetivos de Seguridad tanto para el TOE como para el Entorno

Esta sección especifica los objetivos de seguridad que están conjuntamente destinados al TOE y al entorno.

O.Gestión de Configuración

Implementa la gestión de la configuración para asegurar la identificación de la conectividad del sistema (software, el hardware, y firmware), y los componentes (software, el hardware, y firmware), auditando los datos de configuración, y controlando los cambios de los detalles de configuración.

O.Importación/exportación de datos

Protege los bienes de datos cuando son transmitidos a y desde el TOE, ya sea a través de la intervención de componentes no fiables o directamente a/desde usuarios humanos.

O.Detección de modificaciones de firmware, software, y datos de copia de seguridad

Proporciona protección de integridad para detectar modificaciones de firmware, software, y datos de copia de seguridad.

O.Responsabilidad individual y registros de auditoría

Proporciona responsabilidad individual para cada evento de auditoría. Registra en registros de auditoría: los datos y el instante de cada acción y la entidad responsable de la acción.

O.Protección de integridad de datos de usuarios y software

Proporciona la integridad apropiada para los datos de los usuarios y software.

O.Limitación del acceso administrativo

Diseña las funciones administrativas para que Administradores, Operarios, Oficiales y Auditores no tengan acceso automáticamente a objetos de usuario, excluyendo excepciones necesarias. Controla el acceso al sistema por Operarios y Administradores que localizan fallos en el sistema y realiza actualizaciones de sistema.

O.Mantenimiento de los atributos de usuario



Mantiene un conjunto de atributos de seguridad (que puede incluir los privilegios de acceso de los miembros del rol, etc.) asociados con usuarios individuales. Esto se incluye además de la identidad de usuario.

O.Comportamiento de gestión de las funciones de seguridad

Proporcione la gestión de las funciones para configurar, operar, y mantener los mecanismos de seguridad.

O.Recuperación de los objetos y datos libre de código maligno

Recupera a un estado viable después de que código maligno se introduzca y algún daño ocurra. Ese estado debe ser libre del código maligno original.

O.Procedimientos para prevenir código maligno

Incorpora procedimientos y mecanismos para prevenir el código maligno.

O.Registros de auditoría almacenados de manera protegida

Protege los registros de auditoría contra el acceso no autorizado, modificaciones, o eliminación para garantizar la responsabilidad de las acciones de los usuarios.

O.Datos de usuario y TSF protegidos durante transferencias internas

Garantiza la integridad de los datos del usuario y TSF transferidos internamente al sistema.

O.Requiere inspección para descargas

Requiere inspección para descargas/transferencias.

O.Responde a posibles pérdidas de registros de auditoría almacenados

Responde a posibles pérdidas de registros de auditoría almacenados cuando el almacén de rastros de auditoría esta lleno o casi lleno restringiendo eventos auditables.

O.Restricta acciones antes de la autenticación

Restrige las acciones que un usuario puede llevar a cabo antes de que el TOE autentique la identidad del usuario.

O.Gestión de la configuración relevante a la seguridad



Maneja y actualiza los datos de la política de seguridad de sistema e impone funciones, y otros datos de configuración relevantes a la seguridad, para asegurar que sean consistentes con las políticas de seguridad administrativas.

O.Estampación de tiempo

Proporciona estampación de tiempo para garantizar que la secuencia de eventos pueda ser verificada.

O.Gestión de la autenticación de usuario

Maneja y actualiza la autorización de usuario y datos de privilegio para asegurar que sean consecuente con las políticas personales y seguridad administrativas.

O.Reacciona a los ataques detectados

Implementa automatizadas notificaciones (u otras respuestas) a los ataques TSF- descubiertos en un esfuerzo de identificar los ataques y crear un freno al ataque.

5 Requisitos de Seguridad de la IT

Some requirements in this chapter reference to the following roles: Administrator, Operator, Officer and Auditor. These roles have been extracted from the CIMC Protection Profile, and the definitions for these roles are listed in the section “5.2. Roles” of the [CIMC] document.

5.1 TOE Security Requirements

5.1.1 TOE Security Functional Requirements

The required minimum strength of function level is mandated as “SOF-basic”, for the functional requirements indicated in this section. With this SOF, the TOE shall be resistant to attackers with low attack potential, and remaining vulnerabilities shall only be exploitable by attacker with moderate or high attack potential.

This section specifies the security functional requirements that are applicable to the TOE. All these requirements has been extracted from the [CIMC] Protection Profile. Some of these requirements has been instantiated by means the use of the operations mechanism offered by the Common Criteria. The following table lists all the security functional requirements for the TOE, and the type of operation applied to them.

<i>Functional Requirement</i>	<i>Security Target Operation</i>
FAU_GEN.1.1 (FAU_GEN.1 iteration 2)	Selection, Assignment ⁴
FAU_GEN.1.2 (FAU_GEN.1 iteration 2)	Refinement, Assignment
FAU_GEN.2.1 (FAU_GEN.2 iteration 2)	None
FAU_SEL.1.1 (FAU_SEL.1 iteration 2)	Selection, Assignment
FAU_STG.1.1 (FAU_STG.1 iteration 2)	None
FAU_STG.1.2 (FAU_STG.1 iteration 2)	Selection

⁴ Regarding to the CIMC Protection Profile, a refinement operation has been applied.

FAU_STG.4.1 (FAU_STG.4 iteration 2)	Assignment, Selection
FPT_STM.1.1 (FPT_STM.1 iteration 2)	None
FMT_MOF.1.1 (FMT_MOF.1 iteration 2)	Assignment, Selection
FDP_ACC.1.1 (FDP_ACC.1 iteration 2)	Assignment
FDP_ACF.1.1 (FDP_ACF.1 iteration 2)	Assignment
FDP_ACF.1.2 (FDP_ACF.1 iteration 2)	Assignment
FDP_ACF.1.3 (FDP_ACF.1 iteration 2)	Assignment
FDP_ACF.1.4 (FDP_ACF.1 iteration 2)	Assignment
FDP_ITT.1.1 (FDP_ITT.1 iteration 3)	Assignment, Selection
FDP_ITT.1.1 (FDP_ITT.1 iteration 4)	Assignment, Selection
FDP_UCT.1.1 (FDP_UCT.1 iteration 2)	Assignment, Selection
FPT_RVM.1.1 (FPT_RVM.1 iteration 2)	None
FPT_ITC.1.1 (FPT_ITC.1 iteration 2)	Refinement
FPT_ITT.1.1 (FPT_ITT.1 iteration 3)	Selection, Refinement
FPT_ITT.1.1 (FPT_ITT.1 iteration 4)	Selection, Refinement
FIA_UAU.1.1 (FIA_UAU.1 iteration 2)	Assignment
FIA_UAU.1.2 (FIA_UAU.1 iteration 2)	None
FIA_UID.1.1 (FIA_UID.1 iteration 2)	Assignment
FIA_UID.1.2 (FIA_UID.1 iteration 2)	None
FIA_USB.1.1 (FIA_USB.1 iteration 2)	None
FPT_CIMC_TSP.1.1	None
FPT_CIMC_TSP.1.2	None
FPT_CIMC_TSP.1.3	None
FPT_CIMC_TSP.1.4	None
FDP_ACF_CIMC.2.1	None
FDP_ACF_CIMC.2.2	None
FDP_ACF_CIMC.3.1	None
FDP_SDI_CIMC.3.1	None
FDP_SDI_CIMC.3.2	Assignment
FDP_ETC_CIMC.5.1	None
FDP_CIMC_BKP.1.1	None
FDP_CIMC_BKP.1.2	None
FDP_CIMC_BKP.1.3	None

FDP_CIMC_BKP.1.4	None
FDP_CIMC_BKP.2.1	None
FDP_CIMC_BKP.2.2	None
FDP_CIMC_CSE.1.1	Assignment
FDP_CIMC_CER.1.1	Assignment
FDP_CIMC_CER.1.2	None
FDP_CIMC_CER.1.3	None
FDP_CIMC_CER.1.4	None
FDP_CIMC_CRL.1.1	None
FDP_CIMC_OCSP.1.1	None
FCO_NRO_CIMC.3.1	None
FCO_NRO_CIMC.3.2	Assignment
FCO_NRO_CIMC.3.3	None
FCO_NRO_CIMC.4.1	None
FCO_NRO_CIMC.4.2	None
FMT_MTD_CIMC.4.1	None
FMT_MTD_CIMC.5.1	None
FMT_MTD_CIMC.7.1	None
FMT_MOF_CIMC.3.1	None
FMT_MOF_CIMC.3.2	None
FMT_MOF_CIMC.3.3	None
FMT_MOF_CIMC.3.4	None
FMT_MOF_CIMC.5.1	None
FMT_MOF_CIMC.5.2	None
FMT_MOF_CIMC.5.3	None
FMT_MOF_CIMC.6.1	None
FMT_MOF_CIMC.6.2	None
FMT_MOF_CIMC.6.3	None
FCS_CKM_CIMC.5.1	None

Table 5-3. Functional Requirements for the TOE



5.1.1.1 FAU – Security audit

Security auditing involves recognizing, recording, storing and analyzing information related to security relevant activities (i.e. activities controlled by the TSP). The resulting audit records can be examined to determine which security relevant activities took place and whom (which user) is responsible for them.

Audit includes a chronological recording of events that occur in a system. The objective is to track what occurs to enable the reconstruction and examination of a sequence of events and/or changes in an event. This is useful in ensuring that the system is operated securely and in providing evidence when a suspected or actual security compromise has occurred. Audit also provides for reconstructing a specific state of a system. The objective in a PKI system is to enable an appropriate authority to determine whether a signature should have been accepted as valid.

The audit will be used to reconstruct important events that were performed by the TOE, such as issuance of a CA certificate, and the user or event (e.g., a signed certificate request) that caused them. The audit will be used to arbitrate future disputes by establishing the validity of a signature at a particular time.

The audit log records the security-relevant events that were performed by the TOE and the users or events (e.g., a signed certificate request) that caused them. This subsection specifies the security requirements for maintaining and protecting the integrity of the audit logs.

FAU_GEN – Security Audit Data Generation

This family defines requirements for recording the occurrence of security relevant events that take place under TSF control. This family identifies the level of auditing, enumerates the types of events that shall be auditable by the TSF, and identifies the minimum set of audit-related information that should be provided within various audit record types.

FAU_GEN.1 Audit Data Generation (iteration 2)

Audit data generation defines the level of auditable events, and specifies the list of data that shall be recorded in each record.

FAU_GEN.1.1

The TSF shall be able to generate an audit record of the following auditable events:

- a) Start-up and shutdown of the audit functions.
- b) All auditable events for the [*minimum*] level of audit; and
- c) [
 - *Any changes to the audit parameters, e.g., audit frequency, type of event audited. Any attempt to delete the audit log.*
 - *Audit log signing event.*

- *All security-relevant data that is entered in the system in a Local Data Entry context. The Local Data Entry context implies that a user, operating locally, enters or accept data so that the system can associate the data with the user and list the user in the audit log with the accepted data (this data entry could take the form of a user vouching for information that has already been entered into the computer by clicking on an "accept" button or by otherwise indicating acceptance of the information).*
- *All security-relevant messages that are received by the system in a Remote Data Entry context. The Remote Data Entry context implies that related data could be received over a network in such a way that it can be bound to the identity of the sender of the data (or to the identity of some other user).*
- *All successful and unsuccessful requests for confidential and security-relevant information in a Data Export and Output context.*
- *Whenever the TSF requests generation of a cryptographic key (not mandatory for single session or one-time use symmetric keys).*
- *The loading of Component private keys.*
- *All access to certificate subject private keys retained within the TOE for key recovery purposes.*
- *All changes to the trusted public keys, including additions and deletions.*
- *The manual entry of secret keys used for authentication.*
- *The export of private and secret keys (keys used for a single session or message are excluded).*
- *All certificate requests.*
- *All requests to change the status of a certificate.*
- *Any security-relevant changes to the configuration of the TSF.*
- *All changes to the certificate profile.*
- *All changes to the revocation profile.*
- *All changes to the certificate revocation list profile.*
- *All changes to the OCSP profile.]*

FAU_GEN.1.2

The TSF shall record within each audit record at least the following information:

- a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
- b) For each audit event type, [the following information:



- *Digital signature, keyed hash, or authentication code shall be included in the audit log. This information will be recorded in the register of the audit log signing event.*
- *The identity of the data entry individual if the entered data is linked to any other data (e.g., clicking an “accept” button). This shall be included with the accepted data. This information will be recorded in the register of all security-relevant data that is entered in the system.*
- *The public key component of any asymmetric key pair generated. This information will be recorded in the register of the TSF requests generation of a cryptographic key*
- *The public key and all information associated with the key (in operations of changes, additions and deletions of trusted public keys).*
- *The copy of the related certificate when a certificate request is accepted, and the reason for rejection when a certificate request is rejected.*
- *Whether a request to change the status of a certificate was accepted or rejected.*
- *The changes made to the profile, when a change in the certificate profile is requested.*
- *The changes made to the profile, when a change in the revocation profile is requested.*
- *The changes made to the profile, when a change in the certificate revocation list profile is requested.*
- *The changes made to the profile, when a change in the OCSP profile is requested.]*

Additionally, the audit shall not include plaintext private or secret keys or other critical security parameters.

FAU_GEN.2 User Identity Association (iteration 2)

The TSF shall associate auditable events to individual user identities.

FAU_GEN.2.1

The TSF shall be able to associate each auditable event with the identity of the user that caused the event.

FAU_SEL – Security Audit Event Selection

This family defines requirements to select the events to be audited during TOE operation. It defines requirements to include or exclude events from the set of auditable events.

FAU_SEL.1 Selective Audit (iteration 2)

Selective Audit, requires the ability to include or exclude events from the set of audited events based upon attributes to be specified by the PP/ST author.

FAU_SEL.1.1

The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

- c) [selection: *event type*]
- d) [assignment: *no additional attributes*]

FAU_STG – Security Audit Event Storage

This family defines the requirements for the TSF to be able to create and maintain a secure audit trail.

FAU_STG.1 Protected audit trail storage (iteration 2)

Protected audit trail storage, requirements are placed on the audit trail. It will be protected from unauthorised deletion and/or modification.

FAU_STG.1.1

The TSF shall protect the stored audit records from unauthorized deletion.

FAU_STG.1.2

The TSF shall be able to [*detect*] unauthorized modifications to the audit records in the audit trail.

FAU_STG.4 Prevention of audit data loss (iteration 2)

FAU_STG.4 Prevention of audit data loss specifies actions in case the audit trail is full.

FAU_STG.4.1

The TSF shall [*prevent auditable events, except those taken by Auditor*] and [assignment: *shuts down the system*] if the audit trail is full.

5.1.1.2 FPT – Protection of the TSF

This class contains families of functional requirements that relate to the integrity and management of the mechanisms that provide the TSF (independent of TSP specifics) and to the integrity of TSF data (independent of the specific contents of the TSP data).

FPT_STM – Time stamps

This family addresses requirements for a reliable time stamp function within a TOE.



FPT_STM.1 Reliable time stamps (iteration 2)

This component requires that the TSF provide reliable time stamps for TSF functions.

FPT_STM.1.1

The TSF shall be able to provide reliable time stamps for its own use.

5.1.1.3 FMT – Security Management

This class is intended to specify the management of several aspects of the TSF: security attributes, TSF data and functions. The different management roles and their interaction, such as separation of capability, can be specified.

FMT_MOF – Management of functions in TSF

This family allows authorized users control over the management of functions in the TSF. Examples of functions in the TSF include the audit functions and the multiple authentication functions.

FMT_MOF.1 Management of security functions behavior (iteration 2)

This component allows the authorized users (roles) to manage the behavior of functions in the TSF that use rules or have specified conditions that may be manageable.

FMT_MOF.1.1

The TSF shall restrict the ability to [*modify the behaviour of*] the functions [*list of functions listed in the table below*] to [*the authorised roles as specified in the table below*]

Section/Function	Component	Function/Authorized Role
Security Audit		The capability to configure the audit parameters shall be restricted to Administrators.
Backup and Recovery		The capability to configure the backup parameters shall be restricted to Administrators. The capability to initiate the backup or recovery function shall be restricted to [assignment: <i>Administrator</i>]
Certificate Registration		The capability to approve fields or extensions to be included in a certificate shall be restricted to Officers. If an automated process is used to approve fields or extensions to be included in a certificate, the capability to configure that

		process shall be restricted to Officers.
Data Export and Output		The export of KTS private keys shall require the authorization of at least two Administrators or one Administrator and one Officer, Auditor, or Operator.
Certificate Status Change Approval		<p>Only Officers shall configure the automated process used to approve the revocation of a certificate or information about the revocation of a certificate.</p> <p>Only Officers shall configure the automated process used to approve the placing of a certificate on hold or information about the hold status of a certificate.</p>
KTS Configuration		The capability to configure any TSF functionality shall be restricted to Administrators. (This requirement applies to all configuration parameters unless the ability to configure that aspect of the TSF functionality has been assigned to a different role elsewhere in this document).
Certificate Management Profile	<p>FMT_MOF_CIMC.2 Certificate management profile</p> <p>FMT_MOF_CIMC.3 Extended certificate profile management</p>	The capability to modify the certificate profile shall be restricted to Administrators.
Revocation Management Profile		The capability to modify the revocation profile shall be restricted to Administrators.
Certificate Revocation List Profile Management	<p>FMT_MOF_CIMC.4 Certificate revocation list profile management</p> <p>FMT_MOF_CIMC.5 Extended certificate revocation list profile management</p>	The capability to modify the certificate revocation list profile shall be restricted to Administrators.
Online Certificate Status Protocol (OCSP) Profile Management	FMT_MOF_CIMC.6 OCSP profile management	The capability to modify the OCSP profile shall be restricted to Administrators.

Table 5-1. Authorized Roles for Management of Security Functions Behavior



5.1.1.4 FDP – User Data Protection

This class contains families specifying requirements for TOE security functions and TOE security function policies related to protecting user data. FDP is split into four groups of families (listed below) that address user data within a TOE, during import, export, and storage as well as security attributes directly related to user data.

FDP_ACC – Access control policy

This family identifies the access control SFPs (by name) and defines the scope of control of the policies that form the identified access control portion of the TSP. This scope of control is characterized by three sets: the subjects under control of the policy, the objects under control of the policy, and the operations among controlled subjects and controlled objects that are covered by the policy. The criteria allows multiple policies to exist, each having a unique name. This is accomplished by iterating components from this family once for each named access control policy.

The rules that define the functionality of an access control SFP will be defined by other families such as FDP_ACF and FDP_SDI. The names of the access control SFPs identified here in FDP_ACC are meant to be used throughout the remainder of the functional components that have an operation that calls for an assignment or selection of an "access control SFP."

FDP_ACC.1 Subset access control (iteration 2)

This component requires that each identified access control SFP be in place for a subset of the possible operations on a subset of the objects in the TOE.

FDP_ACC.1.1

The TSF shall enforce the [*CIMC TOE Access Control Policy specified in chapter 2 of the SPM*] on [assignment: *All users of the application successfully identified and authenticated, configuration data, operations and function code , and access and code execution that can be assigned to the application roles*].

FDP_ACF – Access control functions

This family describes the rules for the specific functions that can implement an access control policy named in FDP_ACC. FDP_ACC specifies the scope of control of the policy.

This family addresses security attribute usage and characteristics of policies. The component within this family is meant to be used to describe the rules for the function that implements the SFP as identified in FDP_ACC. The PP/ST author may also iterate this component to address multiple policies in the TOE.

FDP_ACF.1 Security attribute based access control (iteration 2)

This component allows the TSF to enforce access based upon security attributes and named groups of attributes. Furthermore, the TSF may have the ability to explicitly authorize or deny access to an object based upon security attributes.

FDP_ACF.1.1

The TSF shall enforce the [CIMC TOE Access Control Policy specified in chapter 2 of the SPM] to objects based on the following: [the identity of the subject and the set of roles that the subject is authorized to assume].

FDP_ACF.1.2

The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [rules specified in the table below].

Section/Function	Component	Function/Authorized Role
Certificate Request Remote and Local Data Entry		The entry of certificate request data shall be restricted to Officers and the subject of the requested certificate.
Certificate Revocation Request Remote and Local Data Entry		The entry of certificate revocation request data shall be restricted to Officers and the subject of the certificate to be revoked.
Data Export and Output		The export or output of confidential and security-relevant data shall only be at the request of authorized users
Key Generation	FCS_CKM.1 Cryptographic Key Generation	The capability to request the generation of Component keys (used to protect data in more than a single session or message) shall be restricted to Administrators.
Private Key Load		The capability to request the loading of Component private keys into cryptographic modules shall be restricted to Administrators.
Private Key Storage		The capability to request the decryption of certificate subject private keys shall be restricted to Officers. The TSF shall no provide a capability to decrypt certificate subject private keys that may be used to generate digital signatures. At least two Officers or one Officer and an Administrator, Auditor, or Operator shall be required to request the decryption of a certificate subject private key.
Trusted Public Key Entry, Deletion, and Storage		The capability to change (add, revise, delete) the trusted public keys shall be restricted to Administrators.
Secret Key Storage		The capability to request the loading of KTS secret keys into cryptographic modules shall



		be restricted to Administrators.
Private and Secret Key Destruction		The capability to zeroize KTS plaintext private and secret keys shall be restricted to Administrators, Auditors, Officers, and Operators.
Private and Secret Key Export		The capability to export a component private key shall be restricted to Administrators. The capability to export certificate subject private keys shall be restricted to Officers. The export of a certificate subject private key shall require the authorization of at least two Officers or one Officer and an Administrator, Auditor, or Operator.
Certificate Status Change Approval		Only Officers and the subject of the certificate shall be capable of requesting that a certificate be placed on hold. Only Officers shall be capable of removing a certificate from on hold status. Only Officers shall be capable of approving the placing of a certificate on hold. Only Officers and the subject of the certificate shall be capable of requesting the revocation of a certificate. Only Officers shall be capable of approving the revocation of a certificate and all information about the revocation of a certificate.

Table 5-2. Access Controls

FDP_ACF.1.3

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: *none*].

FDP_ACF.1.4

The TSF shall explicitly deny access of subjects to objects based on the [assignment: *none*].

FDP_ITT – Internal TOE transfer

This family provides requirements that address protection of user data when it is transferred between parts of a TOE across an internal channel. This may be contrasted with the FDP_UCT and FDP_UIT families, which provide protection for user

data when it is transferred between distinct TSFs across an external channel, and FDP_ETC and FDP_ITC, which address transfer of data to or from outside the TSF's control.

FDP_ITT.1 Basic internal transfer protection (iteration 3)

This component requires that user data be protected when transmitted between parts of the TOE.

FDP_ITT.1.1

The TSF shall enforce the [CIMC TOE Access Control Policy specified in chapter 2 of the SPM] to prevent the [modification] of user data when it is transmitted between physically-separated parts of the TOE.

FDP_ITT.1 Basic internal transfer protection (iteration 4)

This component requires that user data be protected when transmitted between parts of the TOE.

FDP_ITT.1.1

The TSF shall enforce the [CIMC TOE Access Control Policy specified in chapter 2 of the SPM] to prevent the [disclosure] of user data when it is transmitted between physically-separated parts of the TOE.

FDP_UCT – Inter-TSF user data confidentiality transfer protection

This family defines the requirements for ensuring the confidentiality of user data when it is transferred using an external channel between distinct TOEs or users on distinct TOEs.

FDP_UCT.1 Basic data exchange confidentiality (iteration 2)

In this component, the goal is to provide protection from disclosure of user data while in transit.

FDP_UCT.1.1

The TSF shall enforce the [CIMC TOE Access Control Policy specified in chapter 2 of the SPM] to be able to [transmit] objects in a manner protected from unauthorised disclosure.

5.1.1.5 FPT – Protection of the TSF

This class contains families of functional requirements that relate to the integrity and management of the mechanisms that provide the TSF (independent of TSP specifics) and to the integrity of TSF data (independent of the specific contents of the TSP data). In some sense, families in this class may appear to duplicate components in the FDP (User data protection) class; they may even be implemented using the same mechanisms. However, FDP focuses on user data protection, while FPT focuses on TSF data protection. In fact, components from the FPT class are necessary to provide requirements that the SFPs in the TOE cannot be tampered with or bypassed.



FPT_RVM – Reference mediation

The requirements of this family address the “always invoked” aspect of a traditional reference monitor. The goal of this family is to ensure, with respect to a given SFP, that all actions requiring policy enforcement are validated by the TSF against the SFP. If the portion of the TSF that enforces the SFP also meets the requirements of appropriate components from FPT_SEP (Domain separation) and ADV_INT (TSF internals), then that portion of the TSF provides a “reference monitor” for that SFP.

A TSF that implements a SFP provides effective protection against unauthorized operation if and only if all enforceable actions (e.g. accesses to objects) requested by untrusted subjects with respect to any or all of that SFP are validated by the TSF before succeeding. If an action that could be enforceable by the TSF, is incorrectly enforced or incorrectly bypassed, the overall enforcement of the SFP could be compromised. Subjects could then bypass the SFP in a variety of unauthorised ways (e.g. circumvent access checks for some subjects or objects, bypass checks for objects whose protection was assumed by applications, retain access rights beyond their intended lifetime, bypass auditing of audited actions, or bypass authentication). Note that some subjects, the so called “trusted subjects” with respect to a specific SFP, might be trusted to enforce the SFP by themselves, and bypass the mediation of the SFP.

FPT_RVM.1 Non-bypassability of the TSP (iteration 2)

This component requires non-bypassability for all SFPs in the TSP.

FPT_RVM.1.1

The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

FPT_ITC – Confidentiality of exported TSF data

This family defines the rules for the protection from unauthorised disclosure of TSF data during transmission between the TSF and a remote trusted IT product. This data could, for example, be TSF critical data such as passwords, keys, audit data, or TSF executable code.

FPT_ITC.1 Inter-TSF confidentiality during transmission (iteration 2)

This component requires that the TSF ensure that data transmitted between the TSF and a remote trusted IT product is protected from disclosure while in transit.

FPT_ITC.1.1

The TSF shall protect all TSF data transmitted from the TSF to a remote trusted IT product from unauthorised disclosure during transmission.

FPT_ITT – Internal TOE TSF data transfer

This family provides requirements that address protection of TSF data when it is transferred between separate parts of a TOE across an internal channel.

FPT_IIT.1 Basic internal TSF data transfer protection (iteration 3)

This component requires that TSF data be protected when transmitted between separate parts of the TOE.

FPT_IIT.1.1

The TSF shall protect TSF data from [*modification*] when it is transmitted between separate parts of the TOE.

FPT_IIT.1 Basic internal TSF data transfer protection (iteration 4)

This component requires that TSF data be protected when transmitted between separate parts of the TOE.

FPT_IIT.1.1

The TSF shall protect TSF data from [*disclosure*] when it is transmitted between separate parts of the TOE.

5.1.1.6 FIA – Identification and Authentication

Families in this class address the requirements for functions to establish and verify a claimed user identity.

Identification and Authentication is required to ensure that users are associated with the proper security attributes (e.g. identity, groups, roles, security or integrity levels).

The unambiguous identification of authorized users and the correct association of security attributes with users and subjects is critical to the enforcement of the intended security policies. The families in this class deal with determining and verifying the identity of users, determining their authority to interact with the TOE, and with the correct association of security attributes for each authorized user. Other classes of requirements (e.g. User Data Protection, Security Audit) are dependent upon correct identification and authentication of users in order to be effective.

FIA_UAU – User Authentication

This family defines the types of user authentication mechanisms supported by the TSF. This family also defines the required attributes on which the user authentication mechanisms must be based.

FIA_UAU.1 Timing of authentication (iteration 2)

This component allows a user to perform certain actions prior to the authentication of the user's identity.

FIA_UAU.1.1

The TSF shall allow [*assignment: indicate the authentication mode, introduce the authentication data, cancel the login procedure*] on behalf of the user to be performed before the user is authenticated.



FIA_UAU.1.2

The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

FIA_UID – User Identification

This family defines the conditions under which users shall be required to identify themselves before performing any other actions that are to be mediated by the TSF and which require user identification.

FIA_UID.1 Timing of identification (iteration 2)

This component allows users to perform certain actions before being identified by the TSF.

FIA_UID.1.1

The TSF shall allow [assignment: *indicate the identification mode, introduce the identification data, cancel the login procedure*] on behalf of the user to be performed before the user is identified.

FIA_UID.1.2

The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

FIA_USB – User-subject binding

An authenticated user, in order to use the TOE, typically activates a subject. The user's security attributes are associated (totally or partially) with this subject. This family defines requirements to create and maintain the association of the user's security attributes to a subject acting on the user's behalf.

FIA_USB.1 User-subject binding (iteration 2)

This component requires the maintenance of an association between the user's security attributes and a subject acting on the user's behalf.

FIA_USB.1.1

The TSF shall associate the appropriate user security attributes with subjects acting on behalf of that user.

5.1.2 Requerimientos Funcionales de Seguridad Extendidos del TOE

Esta clase especifica los requisitos funcionales para el TOE de KTS. Estos requisitos funcionales extendidos se extraen del documento [CIMC].

5.1.2.1 Requerimientos Funcionales de Seguridad Extendida del CIMC

FPT – Protección del TSF

Esta clase contiene familias de requisitos funcionales que relacionan la integridad y la gestión de los mecanismos que proporciona el TSF (independiente de las especificaciones TSF) y la integridad de datos de TSF (independiente del contenido específico de los datos de TSF). En algún sentido, las familias en esta clase pueden aparecer para duplicar los componentes de la clase FDP (la protección de datos de usuario); pueden incluso estar implementadas utilizando los mismos mecanismos. Sin embargo, FDP centra la atención en la protección de datos de usuario, mientras FPT lo hace en la protección de datos de TSF. De hecho, los componentes de la clase FPT son necesarios para proporcionar los requisitos que el SFPs en el TOE no se puede manipular ni evitar.

FPT_CIMC_TSP.1 Evento firmado de *log* de auditoría

FPT_CIMC_TSP.1.1

El TSF creará periódicamente un evento de *log* de auditoría firmado en el que se computa una firma digital, un *hash* con clave, o código de autenticación sobre las entradas en el *log* de auditoría.

FPT_CIMC_TSP.1.2

La firma digital, *hash* con clave, o código de autenticación se computará, por lo menos, sobre cada entrada que se haya añadido al *log* de auditoría desde que el evento firmado de *log* de auditoría previo y la firma digital, *hash* con clave, o código de autenticación del evento de *log* firmado de auditoría previo.

FPT_CIMC_TSP.1.3

La frecuencia especificada en que el evento de *log* de auditoría firmado ocurre debe ser configurable.

FPT_CIMC_TSP.1.4

La firma digital, *hash* con clave, o código de autenticación del evento firmado de *log* de auditoría se incluirá en el *log* de la auditoría.

FDP – Protección de datos de usuario

Esta clase contiene familias que especifican los requisitos para funciones de seguridad de TOE y políticas de función de seguridad de TOE relacionadas con la protección datos de usuario.

FDP_ACF_CIMC.2 Protección confidencial de la clave privada de usuario

Las claves privadas pueden ser utilizadas por el KTS para muchos propósitos diferentes y almacenados durante períodos largos. KTS puede almacenar claves de Componente, claves de personal de KTS, y, por propósitos recuperación de claves, las claves privadas de sujetos de certificados.



FDP_ACF_CIMC.2.1

Las claves privadas del personal de KTS se almacenarán en un módulo criptográfico validado FIPS 140-1 o almacenados de forma cifrada. Si las claves del personal de KTS se almacenan en la forma cifrada, la codificación será realizada por el validado módulo criptográfico FIPS 140-1.

FDP_ACF_CIMC.2.2

Si las claves privadas del sujeto del certificado se almacenan en el TOE, serán cifrados utilizando una Clave de Protección de Claves Privadas de Largo. La codificación será realizada por el módulo cifrado validado FIPS 140-1.

FDP_ACF_CIMC.3 Protección confidencial de las claves secretas de usuario

Las claves secretas (simétricas) se pueden utilizar para varios propósitos en el KTS. Pueden ser utilizadas para cifrar otro clave privada o secreta cuando son almacenados dentro o exportados del KTS. Pueden ser utilizados también autenticar titulares (usuarios) y KTSs. Las llaves secretas se deben proteger contra la modificación y la revelación no autorizadas.

Los solicitantes de certificados pueden ser dados de un PIN o contraseña de autenticación. El proceso para generar y entregar estos autenticadores a los solicitantes está fuera del alcance de este documento.

FDP_ACF_CIMC.3.1

Las claves secretas de usuario almacenadas dentro del KTS, pero no dentro de un módulo cifrado validado FIPS 140-1, se almacenará de forma cifrada. El cifrado será realizado por el módulo cifrado validado FIPS 140-1.

FDP_SDI_CIMC.3 Acción y monitorización de las integridad de las claves publicas almacenadas

Los requisitos de seguridad son diseñados para detectar la modificación no autorizada de claves públicas almacenadas en el KTS.

FDP_SDI_CIMC.3.1

Las claves públicas almacenado dentro del KTS, pero no dentro de un módulo cifrado validado FIPS 140-1, se protegerán contra la modificación no detectada con el uso de firmas digitales, *hashes* con clave, o códigos de autenticación.

FDP_SDI_CIMC.3.2

La firma digital, *hashes* con clave, o código de autenticación utilizados para proteger una clave pública se verificará a cada acceso a la clave. Si la comprobación falla, el TSF hará [tarea: genera un informe de error y prohíbe el uso de la clave pública].

FDP_ETC_CIMC.5 Extended user private and secret key export

Las llaves se pueden exportar de los módulos criptográficos por una variedad de razones, inclusive la copia de seguridad de la clave, la réplica, y transmisión de claves privadas de usuario generadas en el KTS.

FDP_ETC_CIMC.5.1

Las llaves privadas y secretas sólo se exportaran del TOE de forma cifrada o utilizando los procedimientos de conocimiento de división. Las claves secretas y privadas electrónicamente distribuidas sólo serán exportadas del TOE de forma cifrada.

FDP_CIMC_BKP.1 Copias de seguridad y recuperación CIMC**FDP_CIMC_BKP.1.1**

El TSF incluirá una función de copia de seguridad.

FDP_CIMC_BKP.1.2

El TSF proporcionará la capacidad de invocar la función de copia de seguridad bajo demanda.

FDP_CIMC_BKP.1.3

Los datos almacenados en la copia de seguridad del sistema serán suficientes para reproducir el estado del sistema en el instante en que la copia de seguridad se creó usando sólo:

- a) una copia de la misma versión del KTS que se utilizó para crear los datos de copia de seguridad;
- b) una copia almacenada de los datos de copia de seguridad;
- c) la clave(s) criptográfica, si hay alguna, necesariaa para verificar la firma digital, *hash* con clave, o código de autenticación protegiendo la copia de seguridad; y
- d) la llave(s) criptográfica, si hay alguna, necesitada para descifrar cualquier parámetro crítico cifrado de seguridad.

FDP_CIMC_BKP.1.4

El TSF incluirá una función de recuperación que es capaz de restaurar el estado del sistema de una copia de seguridad. Al restaurar el estado del sistema, la función de recuperación es sólo requerida al crear un "equivalente" estado del sistema en el que la información acerca de todas las transacciones relevantes de KTS se han mantenido.

FDP_CIMC_BKP.2 Copias de seguridad y recuperación CIMC extendida**FDP_CIMC_BKP.2.1**

Los datos de copia de seguridad serán protegidos contra modificaciones con el uso de firmas digitales, *hashes* con claves, o código de autenticación.

FDP_CIMC_BKP.2.2

Los parámetros críticos de seguridad y otra información confidencial será almacenada sólo de forma cifrada.



FDP_CIMC_CSE.1 Exportación de estado de Certificados

El KTS debe ser capaz de exportar información del estado de los certificados. Cualquier mensaje enviado por el KTS que contiene información de estado de los certificados debe cumplir los requisitos para la Exportación de Estado de Certificados además de los requisitos para Exportación de Datos especificados en las clases FCO y la clase de FPT.

Los requisitos siguientes se aplican a la Exportación de Estado de Certificados.

FDP_CIMC_CSE.1.1

La información de estado de los certificados se exportará del TOE en mensajes cuyo formato cumple con [tarea: el estándar X. 509 para CRLs, el estándar de OCSP como se define por RFC 2560].

FDP_CIMC_CER.1 Generación de Certificados

Las funciones en esta sección dirigen la validación, aprobación, y firma de certificados claves públicos. X. 509. Los certificados de clave pública emitidos por el KTS deben cumplir con la estándar X. 509. Cualquiera campo o extensión a ser incluida en un certificado X. 509 será o bien creado por el KTS según las reglas del estándar X. 509 o validado por el KTS para asegurar la conformidad.

Los datos introducidos en cada campo y la extensión a ser incluidos en un certificado debe ser aprobado. Generalmente, un campo de certificado o valor de extensión se pueden aprobar de una de cuatro maneras:

- 1 Los datos pueden ser aprobados manualmente por un Oficial.
- 2 Un proceso automatizado se puede utilizar para revisar y aprobar los datos An automated process may be used to review and approve the data.
- 3 El valor para un campo o extensión puede ser generado automáticamente por el KTS.
- 4 El valor para un campo o extensión se puede tomar del perfil de certificado.

FDP_CIMC_CER.1.1

El TSF generará sólo certificados cuyo formato cumple con [tarea: el estándar X. 509 para certificados de clave pública].

FDP_CIMC_CER.1.2

El TSF generará sólo certificados que son consistentes con el perfil de certificado actualmente definido.

FDP_CIMC_CER.1.3

El TSF verificará que el potencial sujeto del certificado posee la clave privada que corresponde a la clave pública de la petición de certificado antes de emitir un certificado, a menos que el par clave público/privado sea generado por el TSF, siempre que la clave privada se puede utilizar para generar firmas digitales.

FDP_CIMC_CER.1.4

Si el TSF genera certificados de clave pública X. 509, sólo se generarán certificados que cumplan con los requisitos para certificados especificados en la Recomendación ITU-T X. 509. Como mínimo, el TSF asegurará eso:

- a) El campo de la versión contendrá el entero 0, 1, o 2.
- b) Si el certificado contiene un `issuerUniqueID` o `subjectUniqueID` entonces el campo de versión contendrá el entero 1 o 2.
- c) Si el certificado contiene `extensions` entonces el campo `version` contendrá el entero 2.
- d) El `serialNumber` será único con respecto a la Autoridad de Certificación emisora.
- e) El campo `validity` especificará un valor de `notBefore` que no precede el instante actual y un valor `notAfter` valora que no precede el valor especificado en el `notBefore`.
- f) Si el campo `issuer` contiene un `Name` nulo (por ejemplo, una sucesión de cero nombres distintivos relativos), entonces el certificado contendrá una extensión crítica `issuerAltName`.
- g) Si el campo `subject` contiene un `Name` nulo (por ejemplo, una sucesión de cero nombres distintivos relativos), entonces el certificado contendrá una extensión crítica `subjectAltName`.
- h) El campo `signature` y el `algorithm` en el campo `subjectPublicKeyInfo` contendrán el OID de un algoritmo aprobado o recomendado.

FDP_CIMC_CRL.1 Validación de listas de revocación de Certificados

Las funciones en estos requisitos se dirigen a la validación y aprobación de información de revocación de certificados

Las listas de revocación de certificados (CRLs) emitidas por el KTS cumplirá con el estándar X. 509. Cualquiera campo o extensiones a ser incluidas en un CRL será creado por el KTS según el estándar X. 509.

FDP_CIMC_CRL.1.1

Un TSF que emite CRLs verificará que todos campos obligatorios en cualquier CRL emitida contiene los valores de acuerdo con la Recomendación ITU-T X. 509. Como mínimo, los detalles siguientes se validarán:

- 1 Si el campo `version` está presente, entonces contendrá un 1.
- 2 Si el CRL contiene alguna extensión crítica, entonces el campo de versión será presente y contendrá el entero 1 *If the CRL contains any critical extensions, then the `version` field shall be present and contain the integer 1.*
- 3 Si el campo `issuer` contiene un `Name` nulo (por ejemplo, una sucesión de cero nombres distintivos relativos), entonces la CRL contendrá una extensión crítica `issuerAltName`.



- 4 Los campos `signature` y `signatureAlgorithm` contendrán el OID para un algoritmo digital de firma aprobado FIPS.
- 5 El campo `thisUpdate` indicará la fecha de emisión de la CRL.
- 6 El tiempo especificado en el campo `nextUpdate` (si existe) no precederá el tiempo especificado en el campo `thisUpdate`.

FDP_CIMC_OCSP.1 Validación de respuesta OCSP básica

Las funciones en estos requisitos se dirigen a la validación y aprobación de información de revocación de certificado.

Las respuestas OCSP básicas emitidas por el KTS cumplirán con IETF RFC 2560. Cualquiera campo o extensiones a ser incluidas en una respuesta OCSP será creado por el KTS según IETF RFC 2560.

FDP_CIMC_OCSP.1.1

Si un TSF se configura para permitir las respuestas OCSP de tipo básico de respuesta, el TSF verificará que todos campos obligatorios en la respuesta OCSP básica contiene valores de acuerdo con IETF RFC 2560. como mínimo, los detalles siguientes se validarán:

- 1 El campo `version` contendrá un 0.
- 2 Si el campo `issuer` contiene un `Name` nulo (por ejemplo, una sucesión de cero nombres distintivos relativos), entonces la respuesta contendrá una extensión crítica `issuerAltName`.
- 3 El campo `signatureAlgorithm` contendrá el OID para un algoritmo digital de firma aprobado FIPS.
- 4 El campo `thisUpdate` indicará el instante en que el estado indicado se sabe que es correcto.
- 5 El campo `producedAt` indicará el instante en que el contestador OCSP firmó la respuesta.
- 6 El tiempo especificado en el campo `nextUpdate` (si existe) no precederá el tiempo especificado en el campo `thisUpdate`.

5.1.2.1.1 FCO – Comunicación

Esta clase proporciona dos familias específicamente preocupadas por asegurar la identidad de una parte participando en un intercambio de datos. Estas familias se encargan de asegurar la identidad del autor de la información transmitida (la prueba del origen) y asegurar la identidad del recipiente de la información transmitida (la prueba de recibo). Estas familias aseguran que un autor no pueda negar que mandó el mensaje, ni puede el recipiente negar que lo recibió.

Esta sección cubre los casos en los que los datos deberán ser asociados con un usuario que no actúa localmente. En la mayoría de los casos, esto implicará los datos que se ha recibido en un mensaje que se ha firmado o que contiene un código de autenticación o *hash* con claves permitiendo determinar la fuente del mensaje (en caso que los datos se pueden asociar con la fuente del mensaje). Los datos recibidos

sobre un canal seguro de comunicación (por ejemplo, SSL) podrían ser tratados de manera similar.

Los requisitos de seguridad de entrada remota de datos se aplican siempre que los datos se han recibido de una fuente remota que se considere segura (es decir, la fuente de la información se puede determinar). Estos requisitos se aplican también a comunicaciones entre partes físicamente distribuidas de un solo TOE sobre una red no fiable.

Esta sección especifica también los requisitos de seguridad asociados con la exportación de datos de TOEs. Los datos pueden ser distribuidos a un dispositivo que está fuera de la frontera de un TOE (local o remotamente). El dispositivo o computadora remotos no pueden ser conectados directamente al TOE. La exportación de datos se aplica también cuando los datos se envían entre subcomponentes físicamente distribuidos de un TOE (por ejemplo, los datos enviados entre una CA y RA) y los datos se transmiten sobre una red no fiable.

FCO_NRO_CIMC.3 Prueba impuesta de origen y verificación de origen

FCO_NRO_CIMC.3.1

El TSF impondrá la generación de la evidencia de origen para la información de estado de certificado y toda la otra información relevante a la seguridad en todo momento.

FCO_NRO_CIMC.3.2

El TSF será capaz de relacionar la identidad y [tarea: certificado de autor] del autor de la información, y de las porciones de la información relativas a la seguridad a que la evidencia se aplica.

FCO_NRO_CIMC.3.3

El TSF verificará la evidencia del origen de información para toda la información relativa a la seguridad.

FCO_NRO_CIMC.4 verificación avanzada de origen

FCO_NRO_CIMC.4.1

El TSF, para mensajes iniciales de registro de certificado enviados por el sujeto del certificado, sólo aceptará mensajes protegidos utilizando un código de autenticación, *hashes* con clave, o algoritmo digital de firma.

FCO_NRO_CIMC.4.2

El TSF, para toda la otra información relativa a la seguridad, sólo aceptará la información si se firmó utilizando un algoritmo digital de firma.

5.1.2.1.2 FMT – Gestión de seguridad

Esta clase pretende especificar la gestión de varios aspectos del TSF: los atributos de seguridad, los datos y funciones de TSF. Los diferentes *roles* de gestión y su interacción, tal como la separación de la capacidad, se puede especificar.



FMT_MTD_CIMC.4 TSF protección confidencial de clave privada

FMT_MTD_CIMC.4.1

Las claves privadas de KTS serán almacenados en un módulo criptográfico validados FIPS 140-1 o almacenados de forma cifrada. Si las claves privadas KTS se almacenan de forma cifrada, la codificación será realizada por el módulo criptográfico validado FIPS 140-1.

FMT_MTD_CIMC.5 TSF protección confidencial de la clave secreta

Las claves secretas (simétricas) se pueden utilizar para varios propósitos en el KTS. Pueden ser utilizadas para cifrar otra clave secreta o privadas cuando son almacenadas dentro o exportadas del KTS. Pueden ser utilizados también para autenticar titulares (usuarios). Las claves Secretas se deben proteger contra la modificación y la revelación no autorizadas.

Los solicitantes de certificados pueden ser dotados de un PIN o contraseña de autenticación. El proceso para generar y entregar estos autenticadores a los solicitantes está fuera del alcance de este documento.

FMT_MTD_CIMC.5.1

Las claves secretas almacenadas en TSF, pero no dentro de un módulo criptográfico validado FIPS 140-1, se almacenará de forma cifrada. La codificación será realizada por el módulo criptográfico validado FIPS 140-1.

FMT_MTD_CIMC.7 Exportación de claves secretas y privadas extendida

Las llaves se pueden exportar de módulos criptográficos para una variedad de razones, inclusive las copias de seguridad de claves, la réplica, y la transmisión de claves privadas de usuario generadas en el KTS.

FMT_MTD_CIMC.7.1

Las claves privadas y secretas sólo se exportarán del TOE de forma cifrada o utilizando los procedimientos de conocimiento partido. Las claves secretas y privadas electrónicamente distribuidas sólo serán exportados del TOE de forma cifrada.

FMT_MOF_CIMC.3 Gestión de perfil de certificado extendido

Un perfil de certificado define el conjunto de valores aceptables para campos y extensiones de un certificado. Ejemplos de información que se puede especificar en un perfil de certificado incluyen:

- Restricciones en la identificación clave de dueño (por ejemplo, el sujeto y/o `subjectAltName` en X.509);
- El conjunto de algoritmos admisibles para el par clave público/privado del sujeto;
- La identificación de emisor de certificado (por ejemplo, el emisor y/o `issuerAltName` en X.509);
- Las limitaciones sobre el periodo de tiempo en el que el certificado es válido;

- Información adicional que puede/debe ser incluida en un certificado (por ejemplo, que extensiones pueden ser incluidas en un certificado X. 509);
- Si el sujeto del certificado puede ser un CA;
- Los tipos de operaciones que se pueden realizar utilizando la clave privada que corresponde a la clave pública del certificado (por ejemplo, los valores posibles para el `keyUsage` y/o `extKeyUsage` en X.509);
- La política(s) bajo las que el certificado puede/debe ser emitido.

FMT_MOF_CIMC.3.1

El TSF aplicará un perfil de certificado y asegurará que los certificados emitidos sean consecuentes con ese perfil.

FMT_MOF_CIMC.3.2

El TSF requerirá del Administrador para especificar el conjunto de valores aceptables para los campos y extensiones siguientes:

- La identificación clave de dueño;
- La identificación del algoritmo para el par clave público/privado del sujeto;
- La identificación del emisor de certificado;
- El plazo de tiempo para que el certificado es válido;

FMT_MOF_CIMC.3.3

Si los certificados generados son certificados de clave pública X. 509, el TSF requerirá del Administrador para especificar el conjunto de valores aceptables para los campos y extensiones siguientes:

- `KeyUsage`;
- `BasicConstraints`;
- `CertificatePolicies`

FMT_MOF_CIMC.3.4

El Administrador especificará el conjunto aceptable de extensiones de certificado.

FMT_MOF_CIMC.5 Gestión de perfil de lista de revocación de certificados extendida

Un perfil de lista de revocación de certificados se utiliza para definir el conjunto de valores aceptables para campos y extensiones en un CRL. Ejemplos de valores que pueden ser cubiertos por un perfil de lista de revocación de certificados incluyen:

- `Extensions` – el conjunto de las extensiones que puede/debe ser incluido en un CRL y el valor de cada bit crítico de la extensión.
- `Issuer`, `issuerAltName` – el nombre del emisor de CRL.
- `NextUpdate` – el periodo de vida de un CRL.



FMT_MOF_CIMC.5.1

Si el TSF emite CRLs, el TSF debe aplicar un perfil de lista de revocación de certificados y asegurar que las CRLs emitidas cumplan con el perfil de lista de revocación de certificados.

FMT_MOF_CIMC.5.2

Si el TSF publica CRLs, el TSF requerirá del Administrador para especificar el conjunto de valores aceptables para los campos y extensiones siguientes:

- Issuer;
- IssuerAltName ;
- NextUpdate (i.e., periodo de vida de una CRL).

FMT_MOF_CIMC.5.3

Si el TSF emite CRLs, el Administrador especificará el conjunto aceptable de CRL y extensiones de entrada de CRL.

FMT_MOF_CIMC.6 Gestión de perfil OCSP

Un perfil de protocolo de estado de certificado "en línea" se utiliza para definir el conjunto de valores aceptables para los campos en una respuesta OCSP. El perfil de OCSP puede especificar el tipo(s) de las respuestas que el KTS puede generar (es decir, los valores aceptables para el `responseType`) así como el conjunto de valores aceptables para los campos dentro de los tipos aceptables de la respuesta. Un ejemplo de un valor que puede ser cubierto por un perfil de OCSP para el tipo básico de es `ResponderID`, la identificación del contestador de OCSP.

FMT_MOF_CIMC.6.1

Si el TSF emite respuestas OCSP, el TSF aplicará un perfil de OCSP y asegurará que las respuestas OCSP emitidas cumplan con el perfil de OCSP.

FMT_MOF_CIMC.6.2

Si el TSF emite respuestas OCSP, el TSF requerirá del Administrador para especificar el conjunto de valores aceptables para el campo `responseType` (a menos que el KTS pueda emitir sólo respuestas de tipo básico de respuesta).

FMT_MOF_CIMC.6.3

Si el TSF se configura para permitir las respuestas OCSP de tipo básico de respuesta, el TSF requerirá del Administrador para especificar el conjunto de valores aceptables para el campo de `ResponderID` dentro del tipo básico de respuesta.

5.1.2.1.3 FCS – Soporte Criptográfico

El TSF puede emplear la funcionalidad criptográfica para ayudar a satisfacer varios objetivos de alto nivel de la seguridad. Estos incluyen (pero no son limitados a) : identificación y autenticación, el no-repudio, ruta de confianza, canal seguro y la separación de datos.

Esta clase se utiliza cuando el TOE aplica las funciones criptográficas, la implementación de las cuales podría estar en el hardware, firmware y/o software.

FCS_CKM_CIMC.5 Destrucción (*zeroization*) CIMC de claves privadas y secretas

Estos requisitos de seguridad especifican los requisitos para la destrucción (*zeroization*) de claves privadas y secretas en claro almacenadas dentro del KTS.

FCS_CKM_CIMC.5.1

El TSF proporcionará la capacidad de zeroize las claves secretos y privadas en claro dentro del módulo criptográfico validado FIPS 140-1.

5.1.3 TOE Security Assurance Requirements

The assurance components chosen are those specified to comply with assurance level EAL4, as indicated in the following table:

Assurance Class	Assurance Component
Configuration Management	ACM AUT.1, ACM CAP.4, ACM SCP.2
Delivery and Operation	ADO DEL.2, ADO IGS.1
Development	ADV_FSP.2, ADV_HLD.2, ADV_IMP.1, ADV_LLD.1, ADV_RCR.1, ADV_SPM.1
Guidance Documents	AGD ADM.1, AGD USR.1
Life Cycle Support	ALC DVS.1, ALC LCD.1, ALC FLR.2, ALC TAT.1
Tests	ATE COV.2, ATE FUN.1, ATE IND.2, ATE DPT.1
Vulnerability Assessment	AVA SOF.1, AVA VLA.2, AVA MSU.2

Table 5-3. TOE Security Assurance Requirements

5.1.3.1 ACM – Configuration Management

ACM_CAP - CM capabilities

The capabilities of the CM system address the likelihood that accidental or unauthorised modifications of the configuration items will occur. The CM system should ensure the integrity of the TOE from the early design stages through all subsequent maintenance efforts.

ACM_CAP.4 Generation support and acceptance procedures

A unique reference is required to ensure that there is no ambiguity in terms of which instance of the TOE is being evaluated. Labelling the TOE with its reference ensures that users of the TOE can be aware of which instance of the TOE they are using.

Unique identification of the configuration items leads to a clearer understanding of the composition of the TOE, which in turn helps to determine those items which are subject to the evaluation requirements for the TOE.



Providing controls to ensure that unauthorised modifications are not made to the TOE, and ensuring proper functionality and use of the CM system, helps to maintain the integrity of the TOE.

The purpose of acceptance procedures is to confirm that any creation or modification of configuration items is authorised.

ACM_CAP.4.1D The developer shall provide a reference for the TOE.

ACM_CAP.4.2D The developer shall use a CM system.

ACM_CAP.4.3D The developer shall provide CM documentation.

ACM_CAP.4.1C The reference for the TOE shall be unique to each version of the TOE.

ACM_CAP.4.2C The TOE shall be labelled with its reference.

ACM_CAP.4.3C The CM documentation shall include a configuration list, a CM plan, and an acceptance plan.

ACM_CAP.4.4C The configuration list shall uniquely identify all configuration items that comprise the TOE.

ACM_CAP.4.5C The configuration list shall describe the configuration items that comprise the TOE.

ACM_CAP.4.6C The CM documentation shall describe the method used to uniquely identify the configuration items.

ACM_CAP.4.7C The CM system shall uniquely identify all configuration items.

ACM_CAP.4.8C The CM plan shall describe how the CM system is used.

ACM_CAP.4.9C The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.

ACM_CAP.4.10C The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.

ACM_CAP.4.11C The CM system shall provide measures such that only authorised changes are made to the configuration items.

ACM_CAP.4.12C The CM system shall support the generation of the TOE.

ACM_CAP.4.13C The acceptance plan shall describe the procedures used to accept modified or newly created configuration items as part of the TOE.

ACM_AUT - CM automation

The objective of introducing automated CM tools is to increase the effectiveness of the CM system. While both automated and manual CM systems can be bypassed, ignored, or prove insufficient to prevent unauthorised modification, automated systems are less susceptible to human error or negligence.

ACM_AUT.1 Partial CM automation

In development environments where the implementation representation is complex or is being developed by multiple developers, it is difficult to control changes without the support of automated tools. In particular, these automated tools need to be able to support the numerous changes that occur during development and ensure that those changes are authorised. It is the objective of this component to ensure that the implementation representation is controlled through automated means.

ACM_AUT.1.1D The developer shall use a CM system.

ACM_AUT.1.2D The developer shall provide a CM plan.

ACM_AUT.1.1C The CM system shall provide an automated means by which only authorised changes are made to the TOE implementation representation.

ACM_AUT.1.2C The CM system shall provide an automated means to support the generation of the TOE.

ACM_AUT.1.3C The CM plan shall describe the automated tools used in the CM system.

ACM_AUT.1.4C The CM plan shall describe how the automated tools are used in the CM system.

ACM_SCP - CM scope

The objective of this family is to require items to be included as configuration items and hence placed under the CM requirements of CM capabilities (ACM_CAP). Applying configuration management to these additional items provides additional assurance that the integrity of TOE is maintained.

ACM_SCP.2 Problem tracking CM coverage

A CM system can control changes only to those items that have been placed under CM (i.e., the configuration items identified in the configuration item list). Placing the TOE implementation and the evaluation evidence required by the other assurance components in the ST under CM provides assurance that they have been modified in a controlled manner with proper authorisations.

Placing security flaws under CM ensures that security flaw reports are not lost or forgotten, and allows a developer to track security flaws to their resolution.

ACM_SCP.2.1D The developer shall provide a list of configuration items for the TOE.

ACM_SCP.2.1C The list of configuration items shall include the following: implementation representation; security flaws; and the evaluation evidence required by the assurance components in the ST.

5.1.3.2 ADO – Delivery and Operation

ADO_DEL - Delivery



The requirements for delivery call for system control and distribution facilities and procedures that detail the measures necessary to provide assurance that the security of the TOE is maintained during distribution of the TOE. For a valid distribution of the TOE, the procedures used for the distribution of the TOE address the threats identified in the PP/ST relating to the security of the TOE during delivery.

ADO_DEL.2 Detection of modification

ADO_DEL.2.1D The developer shall document procedures for delivery of the TOE or parts of it to the user.

ADO_DEL.2.2D The developer shall use the delivery procedures.

ADO_DEL.2.1C The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site.

ADO_DEL.2.2C The delivery documentation shall describe how the various procedures and technical measures provide for the detection of modifications, or any discrepancy between the developer's master copy and the version received at the user site.

ADO_DEL.2.3C The delivery documentation shall describe how the various procedures allow detection of attempts to masquerade as the developer, even in cases in which the developer has sent nothing to the user's site.

ADO_IGS – Installation, generation and start-up

Installation, generation, and start-up procedures are useful for ensuring that the TOE has been installed, generated, and started up in a secure manner as intended by the developer. The requirements for installation, generation and start-up call for a secure transition from the TOE's implementation representation being under configuration control to its initial operation in the user environment.

ADO_IGS.1 Installation, generation and start-up procedures

ADO_IGS.1.1D The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE.

ADO_IGS.1.1C The installation, generation and start-up documentation shall describe all the steps necessary for secure installation, generation and start-up of the TOE.

5.1.3.3 ADV – Development

ADV_FSP – Functional Specification

The functional specification is a high-level description of the user-visible interface and behaviour of the TSF. It is an instantiation of the TOE security functional requirements. The functional specification has to show that all the TOE security functional requirements are addressed.

ADV_FSP.2 Fully defined external interfaces

ADV_FSP.2.1D The developer shall provide a functional specification.

ADV_FSP.2.1C The functional specification shall describe the TSF and its external interfaces using an informal style.

ADV_FSP.2.2C The functional specification shall be internally consistent.

ADV_FSP.2.3C The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing complete details of all effects, exceptions and error messages.

ADV_FSP.2.4C The functional specification shall completely represent the TSF.

ADV_FSP.2.5C The functional specification shall include rationale that the TSF is completely represented.

ADV_HLD – High-level design

The high-level design of a TOE provides a description of the TSF in terms of major structural units (i.e. subsystems) and relates these units to the functions that they provide. The high-level design requirements are intended to provide assurance that the TOE provides architecture appropriate to implement the TOE security functional requirements.

The high-level design refines the functional specification into subsystems. For each subsystem of the TSF, the high-level design describes its purpose and function, and identifies the security functions contained in the subsystem. The interrelationships of all subsystems are also defined in the high-level design. These interrelationships will be represented as external interfaces for data flow, control flow, etc., as appropriate.

ADV_HLD.2 Security enforcing high-level design

ADV_HLD.2.1D The developer shall provide the high-level design of the TSF.

ADV_HLD.2.1C The presentation of the high-level design shall be informal.

ADV_HLD.2.2C The high-level design shall be internally consistent.

ADV_HLD.2.3C The high-level design shall describe the structure of the TSF in terms of subsystems.

ADV_HLD.2.4C The high-level design shall describe the security functionality provided by each subsystem of the TSF.

ADV_HLD.2.5C The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software.

ADV_HLD.2.6C The high-level design shall identify all interfaces to the subsystems of the TSF.

ADV_HLD.2.7C The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.



ADV_HLD.2.8C The high-level design shall describe the purpose and method of use of all interfaces to the subsystems of the TSF, providing details of effects, exceptions and error messages, as appropriate.

ADV_HLD.2.9C The high-level design shall describe the separation of the TOE into TSP enforcing and other subsystems.

ADV_IMP – Implementation representation

The description of the implementation representation in the form of source code, firmware, hardware drawings, etc. captures the detailed internal workings of the TSF in support of analysis.

ADV_IMP.1 Subset of the implementation of the TSF

ADV_IMP.1.1D The developer shall provide the implementation representation for a selected subset of the TSF.

ADV_IMP.1.1C The implementation representation shall unambiguously define the TSF to a level of detail such that the TSF can be generated without further design decisions.

ADV_IMP.1.2C The implementation representation shall be internally consistent.

ADV_LLD – Low-level design

The low-level design of a TOE provides a description of the internal workings of the TSF in terms of modules and their interrelationships and dependencies. The low-level design provides assurance that the TSF subsystems have been correctly and effectively refined.

For each module of the TSF, the low-level design describes its purpose, function, interfaces, dependencies, and the implementation of any TSP enforcing functions.

ADV_LLD.1 Descriptive low level design

ADV_LLD.1.1D The developer shall provide the low-level design of the TSF.

ADV_LLD.1.1C The presentation of the low-level design shall be informal.

ADV_LLD.1.2C The low-level design shall be internally consistent.

ADV_LLD.1.3C The low-level design shall describe the TSF in terms of modules.

ADV_LLD.1.4C The low-level design shall describe the purpose of each module.

ADV_LLD.1.5C The low-level design shall define the interrelationships between the modules in terms of provided security functionality and dependencies on other modules.

ADV_LLD.1.6C The low-level design shall describe how each TSP-enforcing function is provided.

ADV_LLD.1.7C The low-level design shall identify all interfaces to the modules of the TSF.

ADV_LLD.1.8C The low-level design shall identify which of the interfaces to the modules of the TSF are externally visible.

ADV_LLD.1.9C The low-level design shall describe the purpose and method of use of all interfaces to the modules of the TSF, providing details of effects, exceptions and error messages, as appropriate.

ADV_LLD.1.10C The low-level design shall describe the separation of the TOE into TSP enforcing and other modules.

ADV_RCR – Representation correspondence

The correspondence between the various TSF representations (i.e. TOE summary specification, functional specification, high-level design, low-level design, implementation representation) addresses the correct and complete instantiation of the requirements to the least abstract TSF representation provided. This conclusion is achieved by step-wise refinement and the cumulative results of correspondence determinations between all adjacent abstractions of representation.

ADV_RCR.1 Informal correspondence demonstration

ADV_RCR.1.1D The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.

ADV_RCR.1.1C For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.

ADV_SPM – Security policy modeling

It is the objective of this family to provide additional assurance that the security functions in the functional specification enforce the policies in the TSP. This is accomplished via the development of a security policy model that is based on a subset of the policies of the TSP, and establishing a correspondence between the functional specification, the security policy model, and these policies of the TSP.

ADV_SPM.1 Informal TOE security policy model

ADV_SPM.1.1D The developer shall provide a TSP model.

ADV_SPM.1.2D The developer shall demonstrate correspondence between the functional specification and the TSP model.

ADV_SPM.1.1C The TSP model shall be informal.

ADV_SPM.1.2C The TSP model shall describe the rules and characteristics of all policies of the TSP that can be modelled.

ADV_SPM.1.3C The TSP model shall include a rationale that demonstrates that it is consistent and complete with respect to all policies of the TSP that can be modelled.



ADV_SPM.1.4C The demonstration of correspondence between the TSP model and the functional specification shall show that all of the security functions in the functional specification are consistent and complete with respect to the TSP model.

5.1.3.4 AGD – Guidance documents

AGD_ADM – Administrator guidance

Administrator guidance refers to written material that is intended to be used by those persons responsible for configuring, maintaining, and administering the TOE in a correct manner for maximum security. Because the secure operation of the TOE is dependent upon the correct performance of the TSF, persons responsible for performing these functions are trusted by the TSF.

Administrator guidance is intended to help administrators understand the security functions provided by the TOE, including both those functions that require the administrator to perform security-critical actions and those functions that provide security-critical information.

AGD_ADM.1 Administrator guidance

AGD_ADM.1.1D The developer shall provide administrator guidance addressed to system administrative personnel.

AGD_ADM.1.1C The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE.

AGD_ADM.1.2C The administrator guidance shall describe how to administer the TOE in a secure manner.

AGD_ADM.1.3C The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment.

AGD_ADM.1.4C The administrator guidance shall describe all assumptions regarding user behaviour that are relevant to secure operation of the TOE.

AGD_ADM.1.5C The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate.

AGD_ADM.1.6C The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_ADM.1.7C The administrator guidance shall be consistent with all other documentation supplied for evaluation.

AGD_ADM.1.8C The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator.

AGD_USR – User guidance

User guidance refers to material that is intended to be used by nonadministrative human users of the TOE, and by others (e.g. programmers) using the TOE's external

interfaces. User guidance describes the security functions provided by the TSF and provides instructions and guidelines, including warnings, for its secure use.

The user guidance provides a basis for assumptions about the use of the TOE and a measure of confidence that non-malicious users, application providers and others exercising the external interfaces of the TOE will understand the secure operation of the TOE and will use it as intended.

AGD_USR.1 User guidance

AGD_USR.1.1D The developer shall provide user guidance.

AGD_USR.1.1C The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE.

AGD_USR.1.2C The user guidance shall describe the use of user-accessible security functions provided by the TOE.

AGD_USR.1.3C The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment.

AGD_USR.1.4C The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behaviour found in the statement of TOE security environment.

AGD_USR.1.5C The user guidance shall be consistent with all other documentation supplied for evaluation.

AGD_USR.1.6C The user guidance shall describe all security requirements for the IT environment that are relevant to the user.

5.1.3.5 ATE – Tests

ATE_COV – Coverage

This family addresses those aspects of testing that deal with completeness of test coverage. That is, it addresses the extent to which the TSF is tested, and whether or not the testing is sufficiently extensive to demonstrate that the TSF operates as specified.

ATE_COV.2 Analysis of coverage

In this component, the objective is to establish that the TSF has been tested against its functional specification in a systematic manner. This is to be achieved through an examination of developer analysis of correspondence.

ATE_COV.2.1D The developer shall provide an analysis of the test coverage.

ATE_COV.2.1C The analysis of the test coverage shall demonstrate the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification.



ATE_COV.2.2C The analysis of the test coverage shall demonstrate that the correspondence between the TSF as described in the functional specification and the tests identified in the test documentation is complete.

ATE_FUN – Functional tests

Functional testing performed by the developer establishes that the TSF exhibits the properties necessary to satisfy the functional requirements of its PP/ST. Such functional testing provides assurance that the TSF satisfies at least the security functional requirements, although it cannot establish that the TSF does no more than what was specified. The family “Functional tests” is focused on the type and amount of documentation or support tools required, and what is to be demonstrated through developer testing. Functional testing is not limited to positive confirmation that the required security functions are provided, but may also include negative testing to check for the absence of particular undesired behaviour (often based on the inversion of functional requirements).

This family contributes to providing assurance that the likelihood of undiscovered flaws is relatively small.

The families Coverage (ATE_COV), Depth (ATE_DPT) and Functional tests (ATE_FUN) are used in combination to define the evidence of testing to be supplied by a developer. Independent functional testing by the evaluator is specified by Independent testing (ATE_IND).

ATE_FUN.1 Functional testing

The objective is for the developer to demonstrate that all security functions perform as specified. The developer is required to perform testing and to provide test documentation.

ATE_FUN.1.1D The developer shall test the TSF and document the results.

ATE_FUN.1.2D The developer shall provide test documentation.

ATE_FUN.1.1C The test documentation shall consist of test plans, test procedure descriptions, expected test results and actual test results.

ATE_FUN.1.2C The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed.

ATE_FUN.1.3C The test procedure descriptions shall identify the tests to be performed and describe the scenarios for testing each security function. These scenarios shall include any ordering dependencies on the results of other tests.

ATE_FUN.1.4C The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE_FUN.1.5C The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified.

ATE_IND – Independent testing

One objective is to demonstrate that the security functions perform as specified.

An additional objective is to counter the risk of an incorrect assessment of the test outcomes on the part of the developer that results in the incorrect implementation of the specifications, or overlooks code that is non-compliant with the specifications.

ATE_IND.2 Independent testing - sample

The objective is to demonstrate that the security functions perform as specified. Evaluator testing includes selecting and repeating a sample of the developer tests.

ATE_IND.2.1D The developer shall provide the TOE for testing.

ATE_IND.2.1C The TOE shall be suitable for testing.

ATE_IND.2.2C The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

ATE_DPT – Depth

The components in this family deal with the level of detail to which the TSF is tested. Testing of security functions is based upon increasing depth of information derived from analysis of the representations.

The objective is to counter the risk of missing an error in the development of the TOE. Additionally, the components of this family, especially as testing is more concerned with the internal structure of the TSF, are more likely to discover any malicious code that has been inserted.

Testing that exercises specific internal interfaces can provide assurance not only that the TSF exhibits the desired external security behaviour, but also that this behaviour stems from correctly operating internal mechanisms.

ATE_DPT.1 Testing: high level design

The subsystems of a TSF provide a high-level description of the internal workings of the TSF. Testing at the level of the subsystems, in order to demonstrate the presence of any flaws, provides assurance that the TSF subsystems have been correctly realised.

ATE_DPT.1.1D The developer shall provide the analysis of the depth of testing.

ATE_DPT.1.1C The depth analysis shall demonstrate that the tests identified in the test documentation are sufficient to demonstrate that the TSF operates in accordance with its high-level design.

5.1.3.6 ALC – Life Cycle Support

ALC_DVS – Development security

Development security is concerned with physical, procedural, personnel, and other security measures that may be used in the development environment to protect the TOE. It includes the physical security of the development location and any procedures used to select development staff.



ALC_DVS.1 Identification of security measures

ALC_DVS.1.1D The developer shall produce development security documentation.

ALC_DVS.1.1C The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.

ALC_DVS.1.2C The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE.

ALC_LCD – Life cycle definition

Poorly controlled development and maintenance of the TOE can result in a flawed implementation of a TOE (or a TOE that does not meet all of its security requirements). This, in turn, results in security violations. Therefore, it is important that a model for the development and maintenance of a TOE be established as early as possible in the TOE's life-cycle.

Using a model for the development and maintenance of a TOE does not guarantee that the TOE will be free of flaws, nor does it guarantee that the TOE will meet all of its security functional requirements. It is possible that the model chosen will be insufficient or inadequate and therefore no benefits in the quality of the TOE can be observed. Using a life-cycle model that has been approved by some group of experts (e.g. academic experts, standards bodies) improves the chances that the development and maintenance models will contribute to the overall quality of the TOE.

ALC_LCD.1 Developer defined life cycle model

ALC_LCD.1.1D The developer shall establish a life-cycle model to be used in the development and maintenance of the TOE.

ALC_LCD.1.2D The developer shall provide life-cycle definition documentation.

ALC_LCD.1.1C The life-cycle definition documentation shall describe the model used to develop and maintain the TOE.

ALC_LCD.1.2C The life-cycle model shall provide for the necessary control over the development and maintenance of the TOE.

ALC_FLR – Flaw remediation

Flaw remediation requires that discovered security flaws be tracked and corrected by the developer. Although future compliance with flaw remediation procedures cannot be determined at the time of the TOE valuation, it is possible to evaluate the policies and procedures that a developer has in place to track and correct flaws, and to distribute the flaw information and corrections.

ALC_FLR.2 Flaw reporting procedures

In order for the developer to be able to act appropriately upon security flaw reports from TOE users, and to know to whom to send corrective fixes, TOE users need to understand how to submit security flaw reports to the developer. Flaw remediation guidance from the developer to the TOE user ensures that TOE users are aware of this important information.

ALC_FLR.2.1D The developer shall provide flaw remediation procedures addressed to TOE developers.

ALC_FLR.2.2D The developer shall establish a procedure for accepting and acting upon all reports of security flaws and requests for corrections to those flaws.

ALC_FLR.2.3D The developer shall provide flaw remediation guidance addressed to TOE users.

ALC_FLR.2.1C The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.

ALC_FLR.2.2C The flaw remediation procedures shall require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.

ALC_FLR.2.3C The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.

ALC_FLR.2.4C The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users.

ALC_FLR.2.5C The flaw remediation procedures shall describe a means by which the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.

ALC_FLR.2.6C The procedures for processing reported security flaws shall ensure that any reported flaws are corrected and the correction issued to TOE users.

ALC_FLR.2.7C The procedures for processing reported security flaws shall provide safeguards that any corrections to these security flaws do not introduce any new flaws.

ALC_FLR.2.8C The flaw remediation guidance shall describe a means by which TOE users report to the developer any suspected security flaws in the TOE.

ALC_TAT – Tools and techniques

Tools and techniques is an aspect of selecting tools that are used to develop, analyse and implement the TOE. It includes requirements to prevent ill-defined, inconsistent or incorrect development tools from being used to develop the TOE. This includes, but is not limited to, programming languages, documentation, implementation standards, and other parts of the TOE such as supporting runtime libraries.



ALC_TAT.1 Well-defined development tools

ALC_TAT.1.1D The developer shall identify the development tools being used for the TOE.

ALC_TAT.1.2D The developer shall document the selected implementation-dependent options of the development tools.

ALC_TAT.1.1C All development tools used for implementation shall be well-defined.

ALC_TAT.1.2C The documentation of the development tools shall unambiguously define the meaning of all statements used in the implementation.

ALC_TAT.1.3C The documentation of the development tools shall unambiguously define the meaning of all implementation-dependent options.

5.1.3.7 AVA – Vulnerability assessment

AVA_MSU – Misuse

Misuse investigates whether the TOE can be configured or used in a manner that is insecure but that an administrator or user of the TOE would reasonably believe to be secure.

The objectives are:

- a) to minimise the probability of configuring or installing the TOE in a way that is insecure, without the user or administrator being able to detect it;
- b) to minimise the risk of human or other errors in operation that may deactivate, disable, or fail to activate security functions, resulting in an undetected insecure state.

AVA_MSU.2 Validation of analysis

The objective is to ensure that misleading, unreasonable and conflicting guidance is absent from the guidance documentation, and that secure procedures for all modes of operation have been addressed. Insecure states should be easy to detect. In this component, an analysis of the guidance documentation by the developer is required to provide additional assurance that the objective has been met.

AVA_MSU.2.1D The developer shall provide guidance documentation.

AVA_MSU.2.2D The developer shall document an analysis of the guidance documentation.

AVA_MSU.2.1C The guidance documentation shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AVA_MSU.2.2C The guidance documentation shall be complete, clear, consistent and reasonable.

AVA_MSU.2.3C The guidance documentation shall list all assumptions about the intended environment.

AVA_MSU.2.4C The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).

AVA_MSU.2.5C The analysis documentation shall demonstrate that the guidance documentation is complete.

AVA_SOF – Strength of TOE security functions

Even if a TOE security function cannot be bypassed, deactivated, or corrupted, it may still be possible to defeat it because there is a vulnerability in the concept of its underlying security mechanisms. For those functions a qualification of their security behaviour can be made using the results of a quantitative or statistical analysis of the security behaviour of these mechanisms and the effort required to overcome them. The qualification is made in the form of a strength of TOE security function claim.

AVA_SOF.1 Strength of TOE security function evaluation

The objective is to ensure that misleading, unreasonable and conflicting guidance is absent from the guidance documentation, and that secure procedures for all modes of operation have been addressed. Insecure states should be easy to detect. In this component, an analysis of the guidance documentation by the developer is required to provide additional assurance that the objective has been met.

AVA_SOF.1.1D The developer shall perform a strength of TOE security function analysis for each mechanism identified in the ST as having a strength of TOE security function claim.

AVA_SOF.1.1C For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the PP/ST.

AVA_SOF.1.2C For each mechanism with a specific strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the specific strength of function metric defined in the PP/ST.

AVA_VLA – Vulnerability analysis

Vulnerability analysis is an assessment to determine whether vulnerabilities identified, during the evaluation of the construction and anticipated operation of the TOE or by other methods (e.g. by flaw hypotheses), could allow users to violate the TSP.

Vulnerability analysis deals with the threats that a user will be able to discover flaws that will allow unauthorised access to resources (e.g. data), allow the ability to interfere with or alter the TSF, or interfere with the authorised capabilities of other users.

AVA_VLA.2 Independent vulnerability analysis

A vulnerability analysis is performed by the developer to ascertain the presence of security vulnerabilities, and to confirm that they cannot be exploited in the intended environment for the TOE.



The evaluator performs independent penetration testing, supported by the evaluator's independent vulnerability analysis, to determine that the TOE is resistant to penetration attacks performed by attackers possessing a low attack potential.

AVA_VLA.2.1D The developer shall perform a vulnerability analysis.

AVA_VLA.2.2D The developer shall provide vulnerability analysis documentation.

AVA_VLA.2.1C The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for ways in which a user can violate the TSP.

AVA_VLA.2.2C The vulnerability analysis documentation shall describe the disposition of identified vulnerabilities.

AVA_VLA.2.3C The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

AVA_VLA.2.4C The vulnerability analysis documentation shall justify that the TOE, with the identified vulnerabilities, is resistant to obvious penetration attacks.

5.2 Security requirements for the IT environment

5.2.1 Security Functional Requirements for the IT environment

This section specifies the security functional requirements that are applicable to the IT environment. All these requirements have been extracted from the [CIMC] Protection Profile, except the FMT_SMF.1.1 requirement (FMT_SMF Specification of Management Functions) that has been included in order to accomplish dependencies between functional requirements.

Some of these requirements have been instantiated by means of the use of the operations mechanism offered by the Common Criteria. The following table lists all the security functional requirements for the IT environment, and the type of operation applied to them.

<i>Functional Requirement</i>	<i>Security Target Operation</i>
FAU_GEN.1.1 (FAU_GEN.1 iteration 1)	Selection, Assignment, Refinement
FAU_GEN.1.2 (FAU_GEN.1 iteration 1)	Refinement, Assignment
FAU_GEN.2.1 (FAU_GEN.2 iteration 1)	Refinement
FAU_SAR.1.1	Assignment, Refinement
FAU_SAR.1.2	Refinement
FAU_SAR.3.1	Selection, Assignment, Refinement
FAU_SEL.1.1 (FAU_SEL.1 iteration 1)	Selection, Assignment, Refinement

FAU_STG.1.1 (FAU_STG.1 iteration 1)	Refinement
FAU_STG.1.2 (FAU_STG.1 iteration 1)	Selection, Refinement
FAU_STG.4.1 (FAU_STG.4 iteration 1)	Selection, Assignment, Refinement
FPT_STM.1.1 (FPT_STM.1 iteration 1)	Refinement
FPT_SEP.1.1	Refinement
FPT_SEP.1.2	Refinement
FPT_RVM.1.1 (FPT_RVM.1 iteration 1)	Refinement
FPT_ITC.1.1 (FPT_ITC.1 iteration 1)	Refinement
FPT_ITT.1.1 (FPT_ITT.1 iteration 1)	Selection, Refinement
FPT_ITT.1.1 (FPT_ITT.1 iteration 2)	Selection, Refinement
FPT_AMT.1.1	Selection, Refinement
FMT_SMR.2.1	Assignment, Refinement
FMT_SMR.2.2	Refinement
FMT_SMR.2.3	Assignment, Refinement
FMT_MOF.1.1 (FMT_MOF.1 iteration 1)	Selection, Assignment, Refinement
FMT_MSA.1.1	Selection, Assignment, Refinement
FMT_MSA.2.1	Refinement
FMT_MSA.3.1	Selection, Assignment, Refinement
FMT_MSA.3.2	Assignment, Refinement
FMT_MTD.1.1	Assignment, Selection, Refinement
FMT_SMF.1.1	Assignment, Refinement
FDP_ACC.1.1 (FDP_ACC.1 iteration 1)	Assignment, Refinement
FDP_ACF.1.1 (FDP_ACF.1 iteration 1)	Assignment, Refinement
FDP_ACF.1.2 (FDP_ACF.1 iteration 1)	Assignment, Refinement
FDP_ACF.1.3 (FDP_ACF.1 iteration 1)	Assignment, Refinement
FDP_ACF.1.4 (FDP_ACF.1 iteration 1)	Assignment, Refinement
FDP_ITT.1.1 (FDP_ITT.1 iteration 1)	Assignment, Selectionn, Refinement
FDP_ITT.1.1 (FDP_ITT.1 iteration 2)	Assignment, Selectionn, Refinement
FDP_UCT.1.1 (FDP_UCT.1 iteration 1)	Assignment, Selection, Refinement
FIA_ATD.1.1	Assignment, Refinement
FIA_UAU.1.1 (FIA_UAU.1 iteration 1)	Assignment, Refinement
FIA_UAU.1.2 (FIA_UAU.1 iteration 1)	Refinement



FIA_UID.1.1 (FIA_UID.1 iteration 1)	Assignment, Refinement
FIA_UID.1.2 (FIA_UID.1 iteration 1)	Refinement
FIA_USB.1.1 (FIA_USB.1 iteration 1)	Refinement
FIA_AFL.1.1	Refinement, Selection, Assignment
FIA_AFL.1.2	Assignment, Refinement
FTP_TRP.1.1	Selection, Refinement
FTP_TRP.1.2	Selection, Refinement
FTP_TRP.1.3	Assignment, Selection, Refinement
FCS_CKM.1.1	Assignment, Refinement
FCS_CKM.4.1	Assignment, Refinement
FCS_COP.1.1	Assignment, Refinement
FPT_TST_CIMC.2.1	None
FPT_TST_CIMC.2.2	Assignment
FPT_TST_CIMC.3.1	None
FPT_TST_CIMC.3.2	Assignment

Table 5-4. Functional Requirements for the TOE Environment

5.2.1.1 FAU – Security audit

Security auditing involves recognizing, recording, storing and analyzing information related to security relevant activities (i.e. activities controlled by the TSP). The resulting audit records can be examined to determine which security relevant activities took place and whom (which user) is responsible for them.

Audit includes a chronological recording of events that occur in a system. The objective is to track what occurs to enable the reconstruction and examination of a sequence of events and/or changes in an event. This is useful in ensuring that the system is operated securely and in providing evidence when a suspected or actual security compromise has occurred. Audit also provides for reconstructing a specific state of a system. The objective in a PKI system is to enable an appropriate authority to determine whether a signature should have been accepted as valid.

The audit will be used to reconstruct important events that were performed by the TOE, such as issuance of a CA certificate, and the user or event (e.g., a signed certificate request) that caused them. The audit will be used to arbitrate future disputes by establishing the validity of a signature at a particular time.

The audit log records the security-relevant events that were performed by the TOE and the users or events (e.g., a signed certificate request) that caused them. This subsection specifies the security requirements for maintaining and protecting the integrity of the audit logs.

FAU_GEN – Security Audit Data Generation

FAU_GEN.1 Audit Data Generation (iteration 1)

Audit data generation defines the level of auditable events, and specifies the list of data that shall be recorded in each record.

FAU_GEN.1.1

The [IT environment] shall be able to generate an audit record of the following auditable events:

- a) Start-up and shutdown of the audit functions.
- b) All auditable events for the [minimum] level of audit; and
- c) [
 - Any changes to the audit parameters, e.g., audit frequency, type of event audited. Any attempt to delete the audit log.
 - Successful and unsuccessful attempts to assume a role.
 - The maximum authentication attempts is changed.
 - Maximum authentication attempts unsuccessful authentication attempts occur during user login.
 - An Administrator unlocks an account that has been locked as a result of unsuccessful authentication attempts.
 - An Administrator changes the type of authenticator, e.g., from password to biometrics.
 - Roles and users are added or deleted.
 - The access control privileges of a user account or a role are modified.]

FAU_GEN.1.2

The [IT environment] shall record within each audit record at least the following information:

- a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
- b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [none]

[Additionally, the audit shall not include plaintext private or secret keys or other critical security parameters.]

FAU_GEN.2 User Identity Association (iteration 1)

The IT environment shall associate auditable events to individual user identities.



FAU_GEN.2.1

The [IT environment] shall be able to associate each auditable event with the identity of the user that caused the event.

FAU_SAR – Security Audit Review

FAU_SAR.1 Audit review

Audit review provides the capability to read information from the audit records.

FAU_SAR.1.1

The [IT environment] shall provide [assignment: Auditors] with the capability to read [all information] from the audit records.

FAU_SAR.1.2

The [IT environment] shall provide the audit records in a manner suitable for the user to interpret the information.

FAU_SAR.3 Selectable audit review

Audit review provides the capability to read information from the audit records.

FAU_SAR.3.1

The [IT environment] shall provide the ability to perform [searches] of audit data based on [the type of event, the user responsible for causing the event and as specified in Table below].

Section/Function	Search Criteria
Certificate Request Remote and Local Data Entry	Identity of ghe subject of the certificate being requested
Certificate Revocation Request Remote and Local Data Entry	Identity of the subject of the certificate to be revoked

Table 5-5. Audit Search Criteria

FAU_SEL – Security Audit Event Selection

FAU_SEL.1 Selective Audit (iteration 1)

Selective Audit, requires the ability to include or exclude events from the set of audited events based upon attributes to be specified by the PP/ST author.

FAU_SEL.1.1

The [IT environment] shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

[selection: *event type*]

i) [assignment: *none*]

FAU_STG – Security Audit Event Storage

FAU_STG.1 Protected audit trail storage (iteration 1)

Protected audit trail storage, requirements are placed on the audit trail. It will be protected from unauthorised deletion and/or modification.

FAU_STG.1.1

The [*IT environment*] shall protect the stored audit records from unauthorized deletion.

FAU_STG.1.2

The [*IT environment*] shall be able to [*detect*] unauthorized modifications to the audit records in the audit trail.

FAU_STG.4 Prevention of audit data loss (iteration 1)

FAU_STG.4 Prevention of audit data loss specifies actions in case the audit trail is full.

FAU_STG.4.1

The [*IT environment*] shall [*prevent auditable events*] except those taken by the [*Auditor*], if the audit trail is full.

5.2.1.2 FPT – Protection of the IT environment

This class contains families of functional requirements that relate to the integrity and management of the mechanisms that provide the IT environment (independent of TSP specifics) and to the integrity of IT environment data (independent of the specific contents of the TSP data).

FPT_STM – Time stamps

FPT_STM.1 Reliable time stamps (iteration 1)

This component requires that the IT environment provide reliable time stamps for IT environment functions.

FPT_STM.1.1

The [*IT environment*] shall be able to provide reliable time stamps for its own use.

FPT_SEP – Domain separation



FPT_SEP.1 TSF domain separation

This component provides a distinct protected domain for the IT environment and provides separation between subjects within the TSC.

FPT_SEP.1.1

[Each operating system in the IT environment] shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

FPT_SEP.1.2

[Each operating system in the IT environment] shall enforce separation between the security domains of subjects in *[its scope of control]*.

FPT_RVM – Reference mediation

FPT_RVM.1 Non-bypassability of the TSP (iteration 1)

This component requires non-bypassability for all SFPs in the TSP.

FPT_RVM.1.1

[Each operating system in the IT environment] shall ensure that *[its policy]* enforcement functions are invoked and succeed before each function within *[its scope of control]* is allowed to proceed.

FPT_ITC – Confidentiality of exported TSF data

FPT_ITC.1 Inter-TSF confidentiality during transmission (iteration 1)

This component requires that the IT environment ensure that data transmitted between the IT environment and a remote trusted IT product is protected from disclosure while in transit.

FPT_ITC1.1

The *[IT environment]* shall protect *[confidential IT environment]* data transmitted from the *[IT environment]* to a remote trusted IT product from unauthorized disclosure during transmission.

FPT_ITT – Internal TOE TSF data transfer

FPT_ITT.1 Basic internal TSF data transfer protection (iteration 1)

This component requires that IT environment data be protected when transmitted between separate parts of the TOE Environment IT.

FPT_ITT.1.1

The *[IT environment]* shall protect *[security-relevant IT environment]* data from *[modification]* when it is transmitted between separate parts of the *[IT environment]*.

FPT_IIT.1 Basic internal TSF data transfer protection (iteration 2)

This component requires that IT environment data be protected when transmitted between separate parts of the TOE Environment IT.

FPT_IIT.1.1

The *[IT environment]* shall protect *[confidential IT environment]* data from *[disclosure]* when it is transmitted between separate parts of the *[IT environment]*.

FPT_AMT – Underlying abstract machine test

FPT_AMT.1 Abstract machine test

This component provides for testing of the underlying abstract machine.

FPT_AMT.1.1

The *[IT environment]* shall run a suite of tests *[selection: during initial start-up]* to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the *[IT environment]*.

5.2.1.3 FMT – Security Management

This class is intended to specify the management of several aspects of the IT environment: security attributes, IT environment data and functions. The different management roles and their interaction, such as separation of capability, can be specified.

FMT_SMR – Security management roles

FMT_SMR.2 Restrictions on security roles

This component specifies the roles with respect to security that the IT environment recognises.

FMT_SMR.2.1

The *[IT environment]* shall maintain the roles *[Administrator, Auditor, and Officer]*.

FMT_SMR.2.2

The *[IT environment]* shall be able to associate users with roles.

FMT_SMR.2.3

The *[IT environment]* shall ensure that *[a) no identity is authorized to assume both an Administrator and an Officer role; b) no identity is authorized to assume both an Auditor and a Officer role; and c) no identity is authorized to assume both an Administrator and an Auditor role]*.

FMT_MOF – Management of functions in TSF



FMT_MOF.1 Management of security functions behavior (iteration 1)

This component allows the authorized users (roles) to manage the behavior of functions in the IT environment that use rules or have specified conditions that may be manageable.

FMT_MOF.1.1

The [IT environment] shall restrict the ability to [modify the behaviour of] the functions [list of functions listed in the table below] to [the authorised roles as specified in the table below]

Section/Function	Component	Function/Authorized Role
Security Audit		The capability to configure the audit parameters shall be restricted to Administrators.
Identification and Authentication		The capability to specify or change maximum authentication attempts shall be restricted to Administrators. The capability to change authentication mechanisms shall be restricted to Administrators.
Account Administrators		The capability to create user accounts and roles shall be restricted to Administrators. The capability to assign privileges to those accounts and roles shall be restricted to Administrators.

Table 5-6. Authorized Roles for Management of Security Functions Behavior

FMT_MSA – Management of security attributes

FMT_MSA.1 Management of security attributes

This component allows authorised users (roles) to manage the specified security attributes.

FMT_MSA.1.1

The [IT environment] shall enforce the [CIMC IT Environment Access Control Policy specified in chapter 2 of the SPM] to restrict the ability to [modify] the security attributes [assignment: user definitions, roles] to [Administrators].

FMT_MSA.2 Secure security attributes

This component ensures that values assigned to security attributes are valid with respect to the secure state.

FMT_MSA.2.1

The [IT environment] shall ensure that only secure values are accepted for security attributes.

FMT_MSA.3 Static attributes initialization

This component ensures that the default values of security attributes are appropriately either permissive or restrictive in nature.

FMT_MSA.3.1

The [IT environment] shall enforce the [CIMC IT Environment Access Control Policy specified in chapter 2 of the SPM] to provide [selection: choose one of: *permissive*] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2

The [IT environment] shall allow the [Administrator] to specify alternative initial values to override the default values when an object or information is created.

FMT_MTD – Management of TSF data

FMT_MTD.1 Management of TSF data

This component allows authorised users to manage IT environment data.

FMT_MTD.1.1

The [IT environment] shall restrict the ability to [view (read) or delete] the [audit logs] to [Auditors].

FMT_SMF – Specification of Management Functions

FMT_SMF.1 Specification of Management Functions

This component requires that the environment provide specific management functions.

FMT_SMF.1.1

The [IT environment] shall be capable of performing the following security management functions: [assignment: *management of users and permissions of access on the part of the users, administration of users authentication*]

5.2.1.4 FDP – User Data Protection

This class contains families specifying requirements for TOE security functions and TOE security function policies related to protecting user data. FDP is split into four groups of families (listed below) that address user data within a TOE, during import, export, and storage as well as security attributes directly related to user data.



FDP_ACC – Access control policy

FDP_ACC.1 Subset access control (iteration 1)

This component requires that each identified access control SFP be in place for a subset of the possible operations on a subset of the objects in the TOE.

FDP_ACC.1.1

The *[IT environment]* shall enforce the *[CIMC IT Environment Access Control Policy specified in chapter 2 of the SPM]* on *[assignment: all users, files and other structures containing sensitive information and all operations among users and objects covered by the CIMC IT Environment Access Control Policy]*

FDP_ACF – Access control functions

FDP_ACF.1 Security attribute based access control (iteration 1)

This component allows the IT environment to enforce access based upon security attributes and named groups of attributes. Furthermore, the IT environment may have the ability to explicitly authorize or deny access to an object based upon security attributes.

FDP_ACF.1.1

The *[IT environment]* shall enforce the *[CIMC IT Environment Access Control Policy specified in chapter 2 of the SPM]* to objects based on the following: *[the identity of the subject and the set of roles that the subject is authorized to assume]*.

FDP_ACF.1.2

The *[IT environment]* shall enforce the following *[rule]* to determine if an operation among controlled subjects and controlled objects is allowed: *[the capability to zeroize plaintext private and secret keys shall be restricted to Administrators, Auditors, Officers, and Operators]*.

FDP_ACF.1.3

The *[IT environment]* shall explicitly authorize access of subjects to objects based on the following additional rules: *[assignment: none]*.

FDP_ACF.1.4

The *[IT environment]* shall explicitly deny access of subjects to objects based on the *[assignment: none]*.

FDP_ITT – Internal TOE transfer

FDP_ITT.1 Basic internal transfer protection (iteration 1)

This component requires that user data be protected when transmitted between parts of the TOE Environment IT.

FDP_ITT.1.1

The [IT environment] shall enforce the [CIMC IT Environment Access Control Policy specified in chapter 2 of the SPM] to prevent the [modifications of security-relevant] of user data when it is transmitted between physically-separated parts of the [IT environment].

FDP_ITT.1 Basic internal transfer protection (iteration 2)

This component requires that user data be protected when transmitted between parts of the TOE.

FDP_ITT.1.1

The [IT environment] shall enforce the [CIMC IT Environment Access Control Policy specified in chapter 2 of the SPM] to prevent the [disclosure of confidential] of user data when it is transmitted between physically-separated parts of the [IT environment].

FDP_UCT – Inter-TSF user data confidentiality transfer protection

FDP_UCT.1 Basic data exchange confidentiality (iteration 1)

In this component, the goal is to provide protection from disclosure of user data while in transit.

FDP_UCT.1.1

The [IT environment] shall enforce the [CIMC IT Environment Access Control Policy specified in chapter 2 of the SPM] to be able to [transmit] objects in a manner protected from unauthorised disclosure.

5.2.1.5 FIA – Identification and Authentication

Families in this class address the requirements for functions to establish and verify a claimed user identity.

Identification and Authentication is required to ensure that users are associated with the proper security attributes (e.g. identity, groups, roles, security or integrity levels).

The unambiguous identification of authorised users and the correct association of security attributes with users and subjects is critical to the enforcement of the intended security policies. The families in this class deal with determining and verifying the identity of users, determining their authority to interact with the TOE, and with the correct association of security attributes for each authorised user. Other classes of requirements (e.g. User Data Protection, Security Audit) are dependent upon correct identification and authentication of users in order to be effective.

FIA_ATD – User attribute definition



FIA_ATD.1 User attribute definition

This component allows user security attributes for each user to be maintained individually.

FIA_ATD.1.1

The *[IT environment]* shall maintain the following list of security attributes belonging to individual users: *[the set of roles that the user is authorized to assume, [assignment: no other security attributes]]*.

FIA_UAU – User Authentication

FIA_UAU.1 Timing of authentication (iteration 1)

This component allows a user to perform certain actions prior to the authentication of the user's identity.

FIA_UAU.1.1

The *[IT environment]* shall allow *[assignment: request for username and password]* on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2

The *[IT environment]* shall require each user to be successfully authenticated before allowing any other *[IT environment]* -mediated actions on behalf of that user.

FIA_UID – User Identification

FIA_UID.1 Timing of identification (iteration 1)

This component allows users to perform certain actions before being identified by the IT environment.

FIA_UID.1.1

The *[IT environment]* shall allow *[assignment: request for username and password]* on behalf of the user to be performed before the user is identified.

FIA_UID.1.2

The *[IT environment]* shall require each user to be successfully identified before allowing any other *[IT environment]* -mediated actions on behalf of that user.

FIA_USB – User-subject binding

FIA_USB.1 User-subject binding (iteration 1)

This component requires the maintenance of an association between the user's security attributes and a subject acting on the user's behalf.

FIA_USB.1.1

The [IT environment] shall associate the appropriate user security attributes with subjects acting on behalf of that user.

FIA_AFL – Authentication failures

FIA_AFL.1 Authentication failure handling

This component requires that the IT environment be able to terminate the session establishment process after a specified number of unsuccessful user authentication attempts. It also requires that, after termination of the session establishment process, the IT environment be able to disable the user account or the point of entry (e.g. workstation) from which the attempts were made until an administrator-defined condition occurs.

FIA_AFL.1.1

[If authentication is not performed in a cryptographic module that has been FIPS 140-1 validated to an overall Level of 2 or higher with Level 3 or higher for Roles and Services], the [IT environment] shall detect when an [Administrator] [configurable maximum authentication attempts] unsuccessful authentication attempts have occurred [since the last successful authentication for the indicated user identity].

FIA_AFL.1.2

When the defined number of unsuccessful authentication attempts has been met or surpassed, the [IT environment] shall [assignment: record a log related to the authentication failure].

5.2.1.6 FTP – Trusted path/channels

Families in this class provide requirements for a trusted communication path between users and the IT environment, and for a trusted communication channel between the IT environment and other trusted IT products.

FTP_TRP – Trusted path

FTP_TRP.1 Trusted path

This component requires that a trusted path between the IT environment and a user be provided for a set of events defined by a PP/ST author. The user and/or the IT environment may have the ability to initiate the trusted path.

FTP_TRP.1.1

The [IT environment] shall provide a communication path between itself and [selection: local, remote] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure.



FTP_TRP.1.2

The [IT environment] shall permit [selection: *the IT environment, local users, remote users*] to initiate communication via the trusted path.

FTP_TRP.1.3

The [IT environment] shall require the use of the trusted path for [initial user authentication], [assignment: *no other services*]

5.2.1.7 FCS – Cryptographic support

The IT environment may employ cryptographic functionality to help satisfy several high-level security objectives. These include (but are not limited to): identification and authentication, non-repudiation, trusted path, trusted channel and data separation. This class is used when the TOE implements cryptographic functions, the implementation of which could be in hardware, firmware and/or software.

FCS_CKM – Cryptographic key management

FCS_CKM.1 Cryptographic key generation

This component requires cryptographic keys to be generated in accordance with a specified algorithm and key sizes which can be based on an assigned standard.

FCS_CKM.1.1

The [FIPS 140-1 validated cryptographic module] shall generate cryptographic keys in accordance with [assignment: *3DES, DES, AES, RSA, DSA*] that meet the following: [assignment: *FIPS 46-3 Data Encryption Standard (DES, 3DES), FIPS PUB 186-2 (DSA and RSA), FIPS PUB 197 (AES)*]

FCS_CKM.4 Cryptographic key destruction

This component requires cryptographic keys to be destroyed in accordance with a specified destruction method which can be based on an assigned standard.

FCS_CKM.4.1

The [IT environment] shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: *any FIPS approved or recommended key destruction method*] that meets the following: [assignment: *FIPS 140-2*]

FCS_COP – Cryptographic operation

FCS_COP.1 Cryptographic operation

This component requires a cryptographic operation to be performed in accordance with a specified algorithm and with a cryptographic key of specified sizes. The specified algorithm and cryptographic key sizes can be based on an assigned standard.

FCS_COP.1.1

The [FIPS 140-1 validated cryptographic module] shall perform [assignment: encryption, decryption, signature generation, signature verification, hash generation, hash verification] in accordance with [assignment: FIPS 46-3 Data Encryption Standard -DES, 3DES- (encryption, decryption), FIPS PUB 186-2 -DSA, RSA- (signature generation, signature verification), FIPS PUB 197 -AES- (encryption, decryption), FIPS PUB 180-2 -SHA1, SHA-256, SHA-512, SHA-384- (hash generation, hash verification)].

5.2.2 Requisitos de Seguridad Extendidos Proprietarios para el entorno IT

Esta sección especifica requisitos de seguridad extendidos propietarios para el entorno IT.

5.2.2.1 FPT – Protección del TSF

Esta clase contiene familias de requisitos que relacionan la integridad y la gestión de mecanismos que proveen al TSF y la integridad de los datos del TSF.

Esta clase además contiene requisitos que están asociados con los mecanismos que están relacionados con los mecanismos de control de acceso.

FPT_ACC – Control de Acceso

Esta familia define requisitos sobre control de acceso a las herramientas y programas que pueden estar disponibles por el TOE.

FPT_ACC.1 Control de Acceso al Software

Este componente requiere medidas de control de acceso a ser aplicadas a aquel software que puede estar disponible por el TOE.

FPT_ACC.1.1

El entorno no debe tener instalado ningún programa de base de datos (ejemplo: telnet, import, export, ...) que acceda a la base de datos usada por el TOE.

5.2.3 Requisitos No IT de Seguridad Extendidos Proprietarios para el entorno

Esta sección especifica requisitos no IT de seguridad extendidos propietarios para el entorno.



5.2.3.1 FPT – Protección del TSF

Esta clase contiene familias de requisitos que relacionan la integridad y la gestión de mecanismos que proveen al TSF y la integridad de los datos del TSF.

Esta clase además contiene requisitos que están asociados con los mecanismos que están relacionados con los mecanismos de control de acceso.

FPT_ACC – Control de Acceso

Esta familia define requisitos sobre control de acceso a las herramientas y programas que pueden estar disponibles por el TOE.

FPT_ACC.1 Control de Acceso al Software

Este componente requiere medidas de control de acceso a ser aplicadas a aquel software que puede estar disponible por el TOE.

FPT_ACC.1.2

Si se utilizan programas que acceden a la base de datos, entonces este acceso debe estar controlado y supervisado por el Auditor.

5.2.4 Requisitos Funcionales de Seguridad Extendidos CIMC

Estos requisitos funcionales extendidos han sido extraídos de los documentos [CEN01c] y [CIMC].

FPT – Protección del TSF

Esta clase contiene familias de requisitos funcionales que relacionan la integridad y la gestión de los mecanismos que proveen el TSF (independientes de los específicos TSP) y la integridad de los datos del TSF (independientes de los contenidos específicos de los datos del TSP). En algunos casos, familias en esta clase aparecen para duplicar componentes en la clase FDP (protección de datos de usuario); éstos pueden incluso estar implementados usando los mismos mecanismos. Sin embargo, FDP se centra en la protección de datos de usuario, mientras que FPT se centra en la protección de datos de usuario. De hecho, los componentes desde la clase FPT son necesarios para proveer requisitos de que los SFPs en el TOE no pueden ser *bypassed*.

FPT_TST_CIMC.2 Test de integridad Software/firmware

FPT_TST_CIMC.2.1

Un código de detección de errores (EDC) o técnica de autenticación recomendada o aprobada FIPS (ejemplo, el cálculo y verificación de un código de autenticación, hash con clave o algoritmo de firma digital) debe ser aplicada a todo el software



relevante de seguridad y el firmware residiendo dentro del KTS (por ejemplo dentro de EEPROM y RAM). El EDC debe tener al menos 16 bits de longitud.

FPT_TST_CIMC.2.2

El código de detección de error, código de autenticación, hash con clave, o firma digital debe verificarse al arrancar y bajo demanda. Si la verificación falla, el entorno IT debe [asignación: *reportar el fallo del test*].

FPT_TST_CIMC.3 Test de carga Software/firmware

FPT_TST_CIMC.3.1

Un mecanismo criptográfico utilizando una técnica de autenticación recomendada o aprobada FIPS (por ejemplo, un código de autenticación, hash con clave, o algoritmo de firma digital) debe ser aplicada a todo el software y firmware relevante de seguridad que puede ser externamente cargado dentro del KTS.

FPT_TST_CIMC.3.2

El entorno IT debe verificar el código de autenticación, hash con clave, o firma digital siempre que el software o firmware sea cargado externamente dentro del KTS. Si la verificación falla, el entorno IT debe [asignación: *no permitir la ejecución del componente donde el test ha fallado*].

6 Especificación resumida del TOE

6.1 Funciones de seguridad del TOE

Esta sección proporciona una descripción de las funciones de seguridad que satisfacen los requisitos de seguridad del TOE. Es decir, esta sección describe cómo se cumplen los requisitos de seguridad por medio de las funciones de seguridad.

El TOE proporciona las siguientes familias de funciones de seguridad:

- Gestión de datos de auditoría
- Base de datos segura
- Gestión del Control de Acceso
- Identificación y
- Comunicaciones seguras
- Gestión de certificados
- Almacén privado seguro
- Gestión del archivo de claves
- Copia de seguridad y recuperación

6.1.1 Gestión de datos de auditoría

KeyOne 3.0 guarda información sobre las operaciones realizadas a través del mantenimiento de un registro de eventos. Las operaciones registradas incluyen las realizadas por los administradores u otros usuarios que utilizan las aplicaciones KeyOne. Algunos ejemplos de operaciones registradas son la aprobación de una petición de certificado, la revocación de un certificado, el procesamiento de un lote, la generación de CRLs. Por medio de una opción de configuración de KeyOne Console es posible configurar la lista de eventos que registrar, de forma que los eventos pueden ser incluidos o excluidos de la lista de eventos registrados por el sistema KeyOne.

Las operaciones se dividen en eventos, de forma que la información sobre uno o más eventos es almacenada para cada operación relevante (por ejemplo la generación



de los diferentes certificados cuando KeyOne CA procesa un lote). Se registran tanto los eventos informativos como los eventos de error. Un evento informativo se almacena en la base de datos del producto KeyOne, en una tabla específica de *log*. Esta tabla podría residir en una base de datos distinta de la del resto de tablas KeyOne, pero siempre se trata de una tabla i3D y, por consiguiente, el registro de eventos se beneficia de todos los mecanismos de seguridad que proporciona la tecnología KeyOne (integridad, autenticación, no repudio).

6.1.1.1 Requerimientos funcionales satisfechos por las funciones de seguridad

Los servicios de gestión de los datos de auditoría, está formado por las siguientes funciones de seguridad:

- Función de selección de *logs* (FUNC_SELL). Esta funcionalidad permite configurar los eventos que auditar. La aplicación KeyOne Console tiene una opción por medio de la que el administrador puede seleccionar los tipos de eventos que incluir/excluir, a partir de la lista completa de eventos que la función de generación de datos de auditoría podría registrar.
- Función de generación de datos de auditoría (FUNC_SADG). Este servicio se encarga de registrar en la tabla de *logs* la información referente a los eventos que ocurren en el sistema.

Estos servicios satisfacen los siguientes requisitos:

6.1.1.1.1 FAU_GEN.1.1 (iteración 2)

La función de generación de datos de auditoría del TOE es capaz de generar un registro de auditoría para los siguientes eventos auditables:

- j) Inicio y finalización de las funciones de auditoría. Las funciones de auditoría siempre se inician/detienen cuando el motor de KeyOne Server se inicia/detiene. No es posible que las aplicaciones KeyOne arranquen las funciones de auditoría sin arrancar el servidor KeyOne, ni tampoco detenerlas sin detener el servidor KeyOne. Cuando el servidor KeyOne arranca se genera un registro de auditoría en la tabla i3D de *logs* y también cuando el servidor KeyOne se detiene se genera entonces un registro de auditoría indicando que el servidor KeyOne ha sido parado.
- k) Los siguientes eventos auditables (que corresponden al nivel mínimo de auditoría del requisito FAU_GEN.1.1):
 - a) Todas las modificaciones de la configuración de auditoría que ocurren mientras las funciones de auditoría están operando (dependencia FAU_SEL.1). Todos los cambios relativos a la configuración de la auditoría serán registrados en un registro de auditoría: modificaciones en la lista de eventos que deben ser auditados (función de selección de *logs*), cambios en los parámetros de configuración de la tabla de *logs* y de la base de datos que contiene dicha tabla (cambio de la tabla de *logs* cambio del driver de conexión con la base de datos, cambio del servicio de la base de datos, cambio del nombre de usuario y de la contraseña de autenticación en la base de datos).

- a) Respecto a los cambios en la hora (dependencia FPT_STM.1) el TOE, depende del reloj del sistema. Los cambios en el reloj del sistema quedan fuera de la funcionalidad que ofrecen los componentes KeyOne y están fuera del control de KeyOne. Por lo tanto, el registro de los cambios de la hora son responsabilidad del control de acceso al entorno IT.
- b) Uso fallido del mecanismo de autenticación de usuarios, incluyendo la identidad de usuario provista (dependencia FIA_UID.1). La función de seguridad de generación de datos de auditoría registra todos los intentos de acceso al sistema KeyOne; estos intentos implican el uso de un mecanismo de identificación y, por consiguiente, este evento es registrado. La identidad provista en los intentos de identificación es también incluida en el *log* registrado (el nombre del usuario o el titular del certificado, dependiendo del tipo de identificación).
- b) Modificaciones de los roles asignados a los grupos de usuarios (dependencia FMT_SMR.1). Los usuarios de las aplicaciones KeyOne pertenecen a uno o varios grupos que se definen para todo el sistema. A cada grupo de usuarios se les puede asignar uno o varios roles, que son específicos de cada aplicación. Estos roles son parte de la configuración de KeyOne Console y se inician a partir de los valores que define la política de seguridad que se selecciona durante la puesta en marcha. Todas las modificaciones sobre las relaciones entre los grupos de usuarios y los roles son registradas en registros de auditoría de la tabla de *logs*.
- c) Peticiones exitosas para realizar una operación sobre un objeto cubierto por la SFP (dependencia FDP_ACF.1). Todas las operaciones sobre objetos que están contemplados por la política funcional de seguridad (roles, claves) se registran en un registro de auditoría. Los siguientes eventos son registrados por la función de seguridad de generación de datos de auditoría:
- Crear, eliminar y modificar usuarios
 - Suspende y habilita usuarios
 - Modificar las propiedades de los usuarios
 - Crear, eliminar y modificar grupos
 - Modificar las propiedades de los grupos
 - Modificar las restricciones sobre las contraseñas
 - Modificar la lista de certificados de CA del sistema
 - Modificar las incompatibilidades entre los roles
 - Modificar los roles asignados a los grupos
 - Crear la tabla de *logs*
 - Renombrar la tabla de *logs*
 - Modificar las conexiones con bases de datos configuradas
- d) El uso fallido del mecanismo de autenticación (dependencia FIA_UAU.1). La función de seguridad de generación de datos de auditoría registra todos los



intentos de acceder al sistema KeyOne; estos intentos implican la utilización del mecanismo de autenticación de usuarios (contraseña de usuario, o desafío-respuesta).

- c) Asignación fallida de atributos de seguridad a los usuario (e.g. creación de un usuario) (dependencia FIA_USB.1). Todos los usuarios disponen de uno o más roles. Estos roles son parte de la configuración de KeyOne Console y se inicializan durante la puesta en marcha, a partir de los valores que define la política de seguridad. No es posible asignar directamente roles a los usuarios, sino que los roles se asignan a los grupos. De esta manera, los usuarios disponen de aquellos roles que dispongan los grupos a los que pertenecen. Queda registrada cualquier intento de modificación de la relaciones entre usuarios y grupos o entre grupos y roles.
- e) Transferencia exitosa de datos de usuario, incluyendo la identificación del método de protección utilizado (dependencia FDP_ITT.1). Este evento afecta a las transferencias de datos de usuario por medio de un canal interno. Desde el punto de vista de KeyOne estas transferencias corresponden a la comunicación entre KeyOne LRA y KeyOne CA y a la comunicación entre KeyOne CA y KeyOne CA. El protocolo de transporte utilizado en estas comunicaciones es SSL/TLS. Se genera un registro de *log* cuando se inicia el servicio KeyOne; este registro contiene los parámetros de conexión SSL/TLS (algoritmos, versión del protocolo, ...) utilizado en cada conexión SSL/TLS (es necesario detener el servidor para cambiar estos parámetros) y, por lo tanto, incluye la identificación del método de protección utilizado.

En la comunicación entre KeyOne LRA y KeyOne CA, los datos de usuario son incluidos en un lote KeyOne. Este lote KeyOne incluye la firma digital de todos los datos que contiene y también los identificadores de los algoritmos utilizados para generar dicha firma digital.

En la comunicación entre KeyOne CA y KeyOne CA, los datos de usuario se incluyen en un mensaje NDCCP. Este mensaje incluye la firma digital de todos los datos que contiene y también los identificadores de los algoritmos utilizados para generar dicha firma digital.

- f) La identidad de cualquier usuario o sujeto que utilice los mecanismos de intercambio de datos (dependencia FDP_UCT.1). Este evento afecta a la transferencia de datos de usuario por medio de un canal externo. Desde el punto de vista del sistema KeyOne, estas transferencias corresponden a las comunicaciones siguientes:
- Comunicaciones entre un usuario y KeyOne VA, utilizando el protocolo OCSP. Si la identidad del solicitante está incluida en la petición OCSP, entonces ésta se incluirá en el registro de *log* que registra la llegada de la petición OCSP (campo `requestorName` de la petición OCSP). Si el la petición OCSP no incluye el campo `requestorName` (petición no firmada), entonces si el cliente OCSP utiliza el protocolo TLS/SSL con autenticación de cliente) los campo emisor y número de serie se guardan en el registro de *log*. Si para comunicarse con el servidor OCSP

el cliente no utiliza TLS/SSL con autenticación de cliente, entonces se guardará la dirección IP del cliente en el registro de log⁵.

Cuando el servidor de KeyOne VA genera la respuesta OCSP y la envía al usuario, entonces se genera un registro de *log* que contiene la identidad KeyOne OCSP (esta identificación se indica por medio del registro de <dirección IP del servidor><nombre del servicio> en el campo *author* de la tabla de registro.

- Comunicación entre las aplicaciones KeyOne y la base de datos. Estas comunicaciones implican la creación de un registro en la base de datos, por lo que estas comunicaciones quedan registradas.
 - Comunicaciones entre las aplicaciones KeyOne y el módulo de seguridad *hardware*. Las comunicaciones que involucran datos de usuario son registradas en una entrada de *log* por la función de generación de datos de auditoría (e.g creación de certificados de usuario).
 - Comunicaciones entre las aplicaciones KeyOne y el dispositivo de creación de firmas. Las comunicaciones que involucran datos de usuario son registradas en una entrada de *log* por la función de generación de datos de auditoría (e.g creación de certificados de usuario).
- g) Éxitos y fallos de la actividad de destrucción de claves (dependencia FCS_CKM.4). Cuando el sistema KeyOne elimina las claves de la aplicación (claves de infraestructura y de control y claves para la emisión de certificados y CRLs) se genera una entrada de *log* en la tabla de *logs*.
- h) Utilización de las funciones de administración (dependencia FMT_SMF.1). La función de seguridad de generación de datos de auditoría registra los eventos referentes a las funciones que invoca un administrador que opera sobre aspectos asociados con la seguridad del TOE, como atributos que protegen datos, atributos que protegen el TOE, atributos de auditoría y atributos de identificación/autenticación. Los siguientes eventos son registrados por la función de seguridad de generación de datos de auditoría:
- Crear, eliminar y modificar usuarios
 - Suspender y habilitar usuarios
 - Modificar las propiedades de los usuarios
 - Crear, eliminar y modificar grupos
 - Modificar las propiedades de los grupos
 - Modificar las restricciones sobre las contraseñas
 - Modificar la lista de certificados de CA del sistema
 - Seleccionar certificados de usuario

⁵ The user identification is registered in the "observation" field of the log table.



- Modificar las incompatibilidades entre los roles
 - Modificar los roles asignados a los grupos
 - Seleccionar la conexión con la base de datos
 - Crear la tabla de *logs*
 - Renombrar la tabla de *logs*
 - Modificar las conexiones con bases de datos configuradas
 - Seleccionar la lista de eventos que auditar
- i) Todos los datos de atributo de seguridad que se han intentado y han sido rechazados (dependencia FMT_MSA.2). Cuando se intenta asignar un valor a un atributo de seguridad (por ejemplo la operación de asignar una contraseña o un certificado a un usuario), entonces el registro de *log correspondiente* contendrá este valor inicial del atributo; si el valor es rechazado, entonces este valor rechazado también se incluye en el registro de la operación fracasada.
- j) Éxito y fracaso de la generación de claves criptográficas y de la actividad de distribución de claves (dependencia FCS_CKM.1, FCS_CKM.2).

Cuando se generan claves criptográficas propias, se genera un registro de *log* que contiene información sobre el evento de generación de la clave.

Con respecto a la actividad de distribución de claves:

- La generación de un certificado implica la generación de un registro de *log* que contiene información sobre el evento de generación del certificado. La generación de un certificado implica la generación de un archivo de registro con información sobre el evento de generación de certificado.
- l) Los eventos auditables siguientes:
- a) Eventos de auditoría de seguridad. La función de seguridad de generación de datos de auditoría, registra los eventos relativos a los cambios en los parámetros de auditoría. Los siguientes eventos se registran en la tabla de *logs*:
- Modificaciones en la lista de eventos que debe auditarse
 - Cambios en los parámetros de configuración de la tabla de *logs* y de la base de datos en la que se almacena dicha tabla (cambio de la tabla de *logs*, cambio del driver de conexión con la base de datos, cambio del servicio de la base de datos, cambio del nombre de usuario y de la contraseña de autenticación en la base de datos).

La aplicación del TOE no tiene ninguna funcionalidad para borrar *logs* de auditoría; puesto que con una aplicación KeyOne no se permite el borrado de logs, es por esto que no existe ningún registro de *log* relativo a este evento.

- b) Todos los datos relevantes para la seguridad que son entrados en el sistema (entrada local de datos). Todas las operaciones que reciben localmente datos relevantes para la seguridad conllevan la generación de una entrada de log. Los siguientes eventos son registrados por la función de seguridad de generación de datos de auditoría:
- Crear, eliminar y modificar usuarios
 - Suspender y habilitar usuarios
 - Modificar las propiedades de los usuarios
 - Crear, eliminar y modificar grupos
 - Modificar las propiedades de los grupos
 - Modificar las restricciones sobre las contraseñas
 - Modificar la lista de certificados de CA del sistema
 - Seleccionar certificados de usuario
 - Modificar las incompatibilidades entre los roles
 - Modificar los roles asignados a los grupos
 - Seleccionar la conexión con la base de datos
 - Crear la tabla de *logs*
 - Renombrar la tabla de *logs*
 - Modificar las conexiones con bases de datos configuradas
 - Seleccionar la lista de eventos que auditar
- c) Todos los mensajes relevantes para la seguridad que el sistema recibe (entrada remota de datos). Este evento está relacionado con la entrada de datos que son recibidos de procedencia remota y para los que es posible identificar y autenticar a su remitente. En el contexto del sistema KeyOne estos eventos son los aquellos que se refieren a la recepción en el servidor de KeyOne VA de una petición firmada (puesto que se requiere autenticación) procedente de un cliente OCSP. Toda la información relativa a las peticiones/respuestas OCSP recibidas/enviadas por el servidor KeyOne VA es incluida en una entrada de *log*. La información que se registra en el campo *observation* de la tabla de *log* es la siguiente:
- Si la petición OCSP está firmada, se registra el campo *requestorName*. Si el OCSP no está firmado y el mecanismo de transporte utilizado es el protocolo TLS/SSL con autenticación de cliente, entonces el emisor y el número de serie del certificado de cliente que se utiliza en el protocolo TLS/SSL se registra en el campo *observations* de la tabla de *log*. Si la petición OCSP no está firmada y la comunicación no utiliza TLS/SSL con autenticación de cliente, entonces se guarda la dirección IP del cliente en el campo *observations*.
 - La identificación del servidor (<IP de la máquina servidor><nombre del servidor>) se registra en el campo *author*.



- Si el content-type de la respuesta es "application/ocsp-request", entonces se registra la siguiente información: a) se registra el status del mensaje de respuesta; b) si el estado es distinto de malformedRequest, entonces se registra la identificación de los certificados (posición del certificado en la petición OCSP, algoritmo de *hash*, *hash* de la clave pública, *hash* del nombre del emisor, número de serie); c) si el estado es successful, entonces se registra la siguiente información: estos de los certificados (estado y razón de la revocación, si el estado es revoked) y los datos de la firma de la respuesta (campo producedAt de la respuesta OCSP); e) información sobre el error (si se produjo).
 - Si el content-type de la respuesta no es "application/ocsp-request", entonces se registra todo el mensaje.
- d) Todas las peticiones exitosas y fallidas de información confidencial y relevante para la seguridad (exportación y salida de datos)
- Respecto a la exportación de datos locales se registran todas las peticiones que supongan la exportación de datos locales, todas las peticiones que impliquen la exportación de datos confidenciales (e.g. peticiones PKCS #12).
 - Respecto a la salida de datos con destinos remotos, se registran todas las peticiones de procedencia remota que impliquen tráfico confidencial (e.g. peticiones de certificación de la RA a la CA).
- e) Todas las peticiones para generar claves criptográficas (la generación de claves simétricas para usar en una única sesión no se incluyen en este evento). La función de seguridad de generación de datos de auditoría registra todas las peticiones para generar claves simétricas y asimétricas. En la fase de arranque del sistema se genera un registro relativo a la creación del sistema; puesto que la creación del sistema implica la generación de claves, en este caso el registro relativo a la creación de las claves está implícitamente contenido en la entrada de *log* que corresponde a la creación del sistema.
- f) La carga de claves privadas de componente (carga de claves privadas). Puesto que la funcionalidad de KeyOne no proporciona medios para cargar claves privadas de componente, no se genera ningún *log* para este evento. Todas las claves privadas se generan y se mantienen en módulos criptográficos y estos componentes están fuera del TOE y pertenecen al entorno IT.
- g) Todos los accesos a claves privadas de usuario que están almacenadas en el TOE con el propósito de posibilitar su recuperación (almacenamiento de claves privadas). La operación de recuperación que proporciona el componente KeyOne KeyArchive (componente localizado en el producto KeyOne CA) es registrada por la función de seguridad de generación de datos de auditoría.
- h) La entrada manual de claves secretas que se utilizan para la autenticación (almacenamiento de claves secretas). Puesto que las aplicaciones KeyOne no permiten la entrada manual de claves secretas, no se genera ninguna entrada de *log* que esté relacionada con este evento.

- i) La exportación de claves secretas y privadas (claves para una sola sesión o mensaje son excluidas de este evento). La función de seguridad de generación de datos de auditoría registra los siguientes eventos:
- Respecto la exportación de claves privadas de usuario, la operación de exportación de PKCS #12 conlleva la generación de una entrada de *log*.
 - Con respecto a la exportación de las claves privadas/secretas de TSF éstas son exportadas al módulo de seguridad hardware cuando el sistema arranca. En este caso, cuando el sistema arranca, se genera una entrada de *log* indicando que el sistema está funcionando. El *log* relativo a la exportación de claves privadas/secretas está implícitamente contenido en la entrada de log que se genera en la fase de arranque del sistema.
- j) Todas las peticiones de certificación (FDP_CIMC_CER.1). Todas las peticiones de certificado que se generan desde KeyOne LRA y desde KeyOne CA son registradas en *logs* de auditoría por la función de generación de datos de auditoría.
- k) Todas las peticiones para cambiar el estado de un certificado (aprobación de cambio de estado de certificado). Todas las peticiones para cambiar el estado de un certificado desde KeyOne LRA y desde KeyOne CA son registradas en *logs* de auditoría por la función de generación de datos de auditoría.
- Cualesquiera cambios en la configuración del TSF que sean relevantes para la seguridad (configuración CIMC). Todos los cambios de la configuración son registrados por la función de generación de datos de auditoría. Estos servicios generan un *log* para cada uno de los eventos siguientes:
 - Crear, eliminar y modificar usuarios
 - Suspender y habilitar usuarios
 - Modificar las propiedades de los usuarios
 - Crear, eliminar y modificar grupos
 - Modificar las propiedades de los grupos
 - Modificar las restricciones sobre las contraseñas
 - Modificar la lista de certificados de CA del sistema
 - Seleccionar certificados de usuario
 - Modificar las incompatibilidades entre los roles
 - Modificar los roles asignados a los grupos
 - Seleccionar la conexión con la base de datos
 - Crear la tabla de *logs*
 - Renombrar la tabla de *logs*



- Modificar las conexiones con bases de datos configuradas
 - Seleccionar la lista de eventos que auditar
- l) Todos los cambios que se realicen en perfiles de certificación (FMT_MOF_CIMC.2, FMT_MOF_CIMC.3). La función de generación de datos de auditoría genera una entrada de *log* para cada cambio que se realiza sobre un perfil de certificación.
- m) Todos los cambios que se realicen sobre perfiles de listas de revocación de certificados (FMT_MOF_CIMC.4, FMT_MOF_CIMC.5). La función de generación de datos de auditoría genera una entrada de *log* para cada cambio que se realiza sobre un perfil de revocación.
- n) Todos los cambios que se realicen sobre un perfil de OCSP (FMT_MOF_CIMC.6). La función de generación de datos de auditoría genera una entrada de *log* para cada cambio que se realice sobre un perfil de OCSP.

6.1.1.1.2 FAU_GEN.1.2 (iteración 2)

La función de seguridad del TOE para la generación de datos de auditoría incluye en cada entrada de *log* la siguiente información:

- Fecha y hora en la que ocurrió el evento (campo *timeLog*). La fecha y la hora se representan en formato numérico (*time_t*).
- Identificación de la entidad que generó el evento (campo *author*).
- Una cadena de caracteres indicando el tipo de entidad que generó el evento (campo *role*).
- Un número que indica el tipo de evento (campo *evtype*).
- Un número que identifica unívocamente al evento entre el conjunto de eventos del mismo tipo y que hayan sido generados por el mismo módulo (campo *event*).
- Un número que identifica el módulo que ha generado el evento (campo *modu*). Esta columna contiene el valor nulo para los eventos de tipo MARK.
- Un número que indica la importancia del evento (campo *evlevel*). Los *logs* se clasifican en las siguientes categorías de acuerdo a su importancia:
 - Informativo: los eventos de esta categoría proporcionan información sobre operaciones que se realizaron con éxito. Esta categoría implica una operación culminada con éxito.
 - Marca: se registra un evento de esta categoría cuando comienza y finaliza una sesión de administración. Esta categoría implica una operación culminada con éxito.
 - Advertencia: indica que se detectó una condición inusual durante la realización de una operación pero que esto no causó que la operación fracasara. Esta categoría implica un fallo en una operación.

- Error: indica que una operación falló debido a un error predecible. Esta categoría implica una operación fallida.
- Error fatal: indica que una circunstancia excepcional e impredecible ocurrió mientras se realizaba una operación. Esta categoría implica una operación fallida.
- Una cadena de caracteres que describe el evento. Para algunos eventos, la descripción va seguida de una lista de parámetros (separados mediante caracteres de nueva línea) cuyo valor varía dependiendo de los datos sobre los que se ejecutó la operación (campo `obser`).

Adicionalmente, se registra la siguiente información:

- En el evento correspondiente a la firma del *log* de auditoría: la firma digital, el *hash* con clave y el código de autenticación se incluyen en el *log* de auditoría (FPT_CIMC_TSP.1).

Los registros de inicio y de final de sesión se firman asimétricamente con el certificado de firma digital del usuario (campo *signature*). Además, los registros de la tabla de sesión se enlazan de tal manera que permite detectar cualquier inserción o borrado fraudulento de sesiones intermedias, cuando se verifica la integridad de la base de datos. Este enlazamiento se realiza del siguiente modo:

- La firma asimétrica del registro de inicio de sesión incluye el valor del campo *signature* del registro de inicio de sesión precedente.
- La firma asimétrica del registro de fin de sesión incluye el valor del campo *signature* del registro de inicio de la misma sesión.

Cuando se cierra una sesión i3D, se modifica el registro de final de sesión que fue insertado en la tabla de sesión cuando la sesión comenzó. Concretamente, se añade a dicho registro el *hash* acumulado de todos los registros históricos generados durante la sesión (campo *hashchain*). Si durante la sesión sólo se realizaron operaciones de consulta, entonces el campo permanecerá vacío. Una vez actualizado, se firma de nuevo el registro de fin de sesión con el certificado de firma digital del usuario y la sesión se da por cerrada.

Adicionalmente, cuando se añade un registro histórico (inserción, actualización o borrado de un registro lógico), se calcula la firma simétrica de este registro histórico y se añade a la tabla de históricos y también en la tabla de consultas asociada.

- Para los eventos relativos a la entrada de datos en el sistema que son relevantes para la seguridad, debe incluirse la siguiente información en el registro de *log*: la identidad de la entrada de datos individual si los datos introducidos están relacionados con cualquier otro dato; esto se incluye con los datos aceptados (entrada local de datos). La función de seguridad para la generación de datos de auditoría incluye en el registro de *log* la identidad de la entidad responsable del evento, los datos introducidos en el sistema y las operaciones realizadas por la entidad relacionada con el evento.
- Para los eventos relativos a la generación de peticiones de claves criptográficas (no se incluyen aquí las claves simétricas de sesión o de un único uso): se incluye en la entrada de *log* la parte pública del par de claves asimétricas (FCS_CKM.1).



La función de seguridad para la generación de datos de auditoría incluye este componente en las siguientes operaciones:

- Pedir la generación de un par de claves asimétricas.
- Pedir la generación de un PKCS #12.
- Pedir la generación de un certificado.
- Para los eventos relativos a cambios en las claves públicas de confianza, incluyendo adición y borrado: la clave pública y toda la información asociada con la clave se incluye en el registro de *log* (entrada, almacenamiento y borrado de claves públicas de confianza). Cuando ocurre cualquier operación que involucra claves públicas de confianza (certificados de CA raíz), se genera un registro de *log* que contiene la clave pública que está implicada en la operación.
- Para cada petición de certificado se genera una entrada de *log* que contiene la siguiente información (FDP_CIMC_CER.1):
 - Si la petición es aceptada, entonces se incluye una copia del certificado en la tabla de certificados. La entrada que se genera en esta tabla está unívocamente enlazada con la entrada de *log* que contiene la petición relacionada (por medio de la única clave pública que está contenida tanto en la petición como en el certificado).
 - Si la petición es rechazada, entonces la razón del rechazo se incluye en la entrada de *log*.
- Para las peticiones de cambio de estado de certificados: se incluye en la entrada de *log* la información sobre si la petición fue aceptada o rechazada (aprobación de cambio de estado de certificado).
- Para los cambios en los perfiles de certificación: se registran en la entrada de *log* los cambios realizados al perfil (FMT_MOF_CIMC.2, FMT_MOF_CIMC.3).
- Para los cambios en los perfiles de revocación: se registran en la entrada de *log* los cambios realizados al perfil (gestión de perfiles de revocación)
- Para los cambios en los perfiles de listas de revocación: se registran en la entrada de *log* los cambios realizados al perfil (FMT_MOF_CIMC.4, FMT_MOF_CIMC.5).

La función de generación de datos de auditoría nunca incluye en claro, dentro de la entrada de *log*, ni claves secretas, ni privadas, ni tampoco ningún parámetro de seguridad que sea crítico.

6.1.1.1.3 FAU_GEN.2.1 (iteración 2)

Puesto que la función de seguridad para la generación de datos de auditoría siempre registra la identidad de quien generó el evento, siempre queda constancia de la asociación que existe entre cada evento que se audita y el usuario que lo produjo.

6.1.1.1.4 FAU_SEL.1.1 (iteración 2)

La función de selección de *logs* permite poner y quitar eventos auditables de la lista de eventos que se van a auditar, en base al tipo de evento. Esta función proporciona la funcionalidad que permite configurar, desde la aplicación KeyOne Console, los eventos que efectivamente se registrarán en la tabla de *logs*. Esta aplicación dispone de una opción que muestra gráficamente los eventos que se están auditando; utilizando esta opción, la aplicación permite poner y quitar eventos de la lista.

6.1.2 Base de datos segura

El sistema KeyOne utiliza bases de datos i3D. La tecnología i3D tiene las siguientes propiedades:

- Permite verificar la integridad de la base de datos, es decir, detectar posibles manipulaciones fraudulentas de los datos.
- Mediante firmas digitales, garantiza el no repudio ante los autores de operaciones realizadas sobre los datos.
- Mantiene un histórico de las actualizaciones de datos, es decir, guarda sucesivas versiones de cada registro (cada versión es el resultado de diversas operaciones realizadas sobre el registro). De esta manera se mantiene un histórico de las operaciones realizadas y, al actualizar datos, se evita perder firmas digitales previamente realizadas por otros usuarios.
- Permite que varios usuarios accedan concurrentemente a las mismas tablas de base de datos.
- Funciona sobre cualquier sistema de gestión de base de datos SQL. La funcionalidad de i3D reside completamente en el sistema del cliente, sin necesidad de servidores intermedios.

Las operaciones se agrupan en sesiones (sesiones i3D). Por ello, para consultar o modificar una tabla, el usuario debe abrir primero una sesión en esa tabla. Tras concluir las operaciones deseadas, el usuario debe cerrar la sesión. Las sesiones realizadas por varios usuarios sobre una misma tabla se identifican mediante un número secuencial llamado identificador de sesión.

Las entidades que acceden a una base de datos i3D se clasifican en los siguientes tipos:

- Usuarios o entidades que realizan operaciones sobre los datos. Por ejemplo, lectura, inserción actualización o eliminación de registros en una tabla de la base de datos tabla. Cada usuario de tener un certificado propio de firma que se usará para firmar los datos que añada, actualice o borre el usuario, durante la sesión i3D.
- Algunas entidades pueden realizar operaciones especiales en la base de datos (entidades master). Dichas entidades deben tener un certificado de firma digital y cifrado de datos. Las entidades habilitadas como master tienen las siguientes funciones:



- Verificar y cerrar sesiones i3D que los usuarios no cerraron (por ejemplo, a causa de una catástrofe).
- Firmar de nuevo sesiones i3D ya cerradas para forzar la recuperación de la integridad de los datos.

Cuando se empieza una sesión de administración, se inicia una sesión i3D en cada una de las tablas del producto. Las diversas operaciones realizadas por los usuarios de la aplicación (aprobación de peticiones de certificado, revocación de certificado, generación de CRL, proceso de lotes, etc.) provocan la inserción de nuevos registros de histórico en las tablas i3D. De esta forma, la base de datos mantiene internamente sucesivas actualizaciones de cada registro lógico. Cuando los contenidos de una tabla se consultan desde una tabla de la aplicación de administración, siempre se muestra la última versión de los registros.

Las sesiones i3D se cierran automáticamente cuando se acaba la sesión de administración; esto es, cuando el servidor se cierra mediante las opciones de aplicación. Conviene que la sesión de administración siempre se cierre de esta manera. De otro modo, las sesiones i3D permanecen abiertas y sólo un usuario master puede cerrarlas.

6.1.2.1 Estructura interna de una base de datos i3D

La tecnología i3D se basa en el uso de firmas digitales y otras técnicas criptográficas para asegurar la integridad y el no repudio en la base de datos. En esta sección se da una visión general de ciertos aspectos de la estructura interna y el funcionamiento de una base de datos i3D (para cumplir los requisitos de seguridad incluidos en el documento).

Desde este punto de vista, el término "tabla lógica" se utilizará para referirse al conjunto de registros en los que el usuario de la base de datos realiza operaciones de lectura, inserción, actualización o eliminación. Asimismo, el término "registros lógicos" se utilizará para referirse a una tabla lógica.

6.1.2.1.1 Tablas i3D

En una base de datos i3D no hay relación uno a uno entre las tablas lógicas y físicas (las que realmente residen en el sistema gestor de la base de datos). Por el contrario, cada tabla lógica tiene tres tablas físicas:

- Tabla de sesión: Contiene información sobre todas las sesiones i3D (cerradas o no) realizadas sobre la tabla lógica.
- Tabla de registros históricos: Contiene todas las actualizaciones de cada registro de la tabla lógica.
- Tabla de browsing: Contiene información duplicada sobre la última versión de cada registro de la tabla lógica. Permite consultas SQL.

6.1.2.1.2 Inicio de una sesión i3D

Cada vez que un usuario inicia una sesión i3D en una tabla lógica, se añaden dos registros a la tabla de sesión: la entrada de inicio de sesión y la entrada de final de sesión. Dichos registros contienen diversos campos de control entre los que se

encuentra el identificador de sesión (campo `sessionid`). Dicho identificador es diferente en cada sesión i3D iniciada por un usuario en la tabla lógica.

Además, al iniciar la sesión i3D, se genera aleatoriamente una clave simétrica 3DES. Esta clave, denominada "clave de sesión", se guarda en el registro de inicio de sesión, cifrada asimétricamente y reservada para los usuarios master de la base de datos (sólo ellos pueden conocerla).

Los registros de inicio y final de sesión se firman asimétricamente con el certificado de firma digital del usuario (campo `signature`). Además, todos los registros de la tabla de sesión se vinculan de tal modo que cualquier posible inserción o eliminación fraudulenta en una sesión intermedia será detectada al verificar la integridad de la base de datos. Los registros se vinculan del siguiente modo:

- La firma asimétrica del registro de inicio de sesión incluye el valor del campo `signature` del registro de inicio de la anterior sesión.
- La firma asimétrica del registro de final de sesión incluye el valor del campo `signature` del registro de inicio de la misma sesión.

6.1.2.1.3 Operaciones sobre la tabla lógica

De una sesión i3D iniciada se dice que es una sesión activa. Es decir, que el usuario puede realizar operaciones SQL sobre los registros de la tabla lógica; las operaciones realizadas se asociarán a las de dicha sesión. A continuación se describe como estas operaciones sobre la tabla lógica afectan a la tabla de registros históricos y a la tabla de browsing.

Inserción de un registro lógico

Provoca la inserción de un registro en la tabla de registros históricos. Dicha tabla contiene datos codificados en DER (campo `info`) y otros campos de control. El nuevo registro incluye información sobre el identificador y la sesión activa (campo `sessionid`). El registro completo se firma simétricamente con la clave de sesión (campo `hmac`).

Además, se añade un registro a la tabla de browsing asociada al registro histórico (mediante el campo `hmac`). Dicho registro contiene parte de los datos del registro lógico guardados en campos no codificados.

Actualización del registro lógico

Provoca la inserción de un registro histórico que contiene los nuevos datos del registro lógico. El nuevo registro está relacionado con el anterior registro histórico del mismo registro lógico. Además, el nuevo registro incluye información sobre el identificador de la sesión activa (campo `sidcurrent`). El registro completo se firma con la clave de sesión (campo `hmac`).

La tabla de browsing asociada al registro histórico anterior se actualiza con los nuevos datos y se asocia al nuevo registro histórico (mediante el campo `hmac`).

Selección y obtención de un registro lógico

Las consultas SQL requeridas por el usuario se realizan sobre las columnas de la tabla de browsing. Cada registro de esta tabla corresponde a un registro lógico.



Tras seleccionar el registro deseado en la tabla de browsing, se accede a la tabla de registro históricos (mediante el valor del campo `hmac`) para recuperar el último registro histórico asociado al registro lógico. El actual valor del registro lógico se obtiene descodificando los datos guardados en la columna `info` del registro histórico.

Esta operación no provoca la inserción o modificación de datos en tabla alguna.

Supresión de un registro lógico

Provoca la inserción de un registro histórico marcado de una manera especial para indicar que el registro lógico ha sido eliminado y por tanto no se le pueden asociar más registros históricos (campo `deleted`). El nuevo registro histórico incluye información sobre el identificador de la sesión activa (campo `sidcurrent`). El registro completo se firma con la clave de sesión (campo `hmac`).

Además, la entrada correspondiente al registro lógico que se ha borrado se elimina de la tabla de browsing, de tal modo que sólo queda rastro de su existencia en la tabla de registros históricos.

6.1.2.1.4 Cierre normal de una sesión i3D

Tras realizar un cierto número de operaciones sobre la tabla lógica, el usuario debe cerrar finalmente la sesión i3D activa. Este modo de cerrar la sesión recibe el nombre de cierre normal.

Cuando se cierra una sesión i3D, se modifica el registro de final de sesión que se insertó en la tabla al iniciar la sesión. En concreto, al registro se añade el hash acumulado de todos los registros de histórico generados durante la sesión (campo `hashchain`). Si la sesión sólo ha consistido en operaciones de consulta, el campo permanece vacío.

El valor del campo `entrytype` también se modifica como se indica a continuación. Una vez actualizado, el registro de final de sesión se firma de nuevo asimétricamente con el certificado de firma digital del usuario y la sesión se considera cerrada.

La columna `entrytype` de la tabla de sesión permite distinguir el registro de inicio de sesión del registro de final de sesión (y en este último registro indica el modo de cierre de la sesión).

6.1.2.2 Requisitos funcionales que cumplen las Funciones de Seguridad

Los servicios de base de datos segura se componen de las siguientes funciones de seguridad:

- Función de verificación de la integridad de la base de datos (FUNC_DBIV). Esta funcionalidad detecta modificaciones en los registros de la base de datos KeyOne (registros que contienen certificados, CRLs, peticiones, logs de auditoría, lotes KeyOne, etc.). Esta verificación se basa en el mecanismo i3D de KeyOne que garantiza los servicios de integridad y verificación de integridad.

Esta función consiste básicamente en la herramienta de línea de comandos `i3dverify.ws` que verifica mediante certificados y claves según el tipo de test

que deba realizarse. En concreto, esta función permite realizar los siguientes tests:

- Verificación de la integridad de una sesión

El test consiste en verificar los registros de histórico y la información de cabecera asociada a una sesión i3D. Debe realizarse cuando se sospecha o se tiene la certeza de que la sesión presenta inconsistencias.

- Verificación de la integridad de una sesión cerrada

Mediante este test se verifica la integridad de todos los registros de histórico generados en una cierta sesión i3D. También se verifica que ningún registro haya sido insertado o borrado fraudulentamente. La sesión debe estar cerrada (el campo `entrytype` del registro de final de sesión debe tener un valor superior a 1).

Bajo este supuesto, la integridad de la sesión puede verificarse sin conocer la clave de sesión. Por tanto, cualquier entidad puede ejecutar el test. Se requiere el certificado de firma digital del usuario que realizó la sesión. Si la sesión la cerró un usuario master, también es necesario el certificado de firma digital del usuario master.

- Verificación de la integridad de una sesión abierta

Los test de integridad también se pueden realizar sobre i3D sesiones abiertas. Para ello es necesario conocer la correspondiente clave de sesión. Por ello, sólo un usuario master puede realizar este test. También es necesario el certificado de firma digital del usuario que inició la sesión.

Es aconsejable realizar los tests de integridad cuando todos los usuarios de la base de datos están desconectados. De esta forma se garantiza que toda sesión abierta permanece inactiva. No obstante, este test también puede realizarse en una sesión activa.

Al realizar este test, la herramienta de verificación primero verifica la integridad de los registros de inicio y final de sesión. En concreto, la herramienta verifica la firma asimétrica de dichos registros (campo `signature`).

- Verificación de la integridad de la tabla de registros

Este test permite verificar todo el contenido de la tabla de registros históricos (mediante una verificación secuencial de todas las sesiones que contiene). También se puede indicar que cada sesión sea verificada exhaustivamente como se explica más arriba.

Mediante este test, se verifica la integridad de todas las sesiones i3D realizadas sobre una cierta tabla lógica, verificando también que ninguna sesión intermedia ha sido insertada o eliminada fraudulentamente. Es necesario conocer los certificados de firma digital de los usuarios que han realizado sesiones en la tabla. Además, si algunas sesiones fueron cerradas por usuarios master, también deben conocerse los certificados de firma digital de dichos usuarios.



- Verificar la capacidad de la base de datos (FUNC_CDBC). Este servicio permite gestionar los servicios KeyOne cuando la base de datos está al máximo de su capacidad.
- Gestión de la tabla de sesión (FUNC_I3DSESSION). Esta funcionalidad vincula todos los registros de la tabla de sesión mediante la firma asimétrica de los registros de inicio y final.
- Gestión de la tabla de histórico (FUNC_I3DHISTORIC). Esta funcionalidad aporta un mecanismo de integridad para las tablas de histórico y browsing. El mecanismo consiste en generar una firma simétrica del registro de histórico e incluir dicha firma en el correspondiente registro de la tabla de browsing.

Este servicio cumple los siguientes requisitos:

6.1.2.2.1 FAU_STG.1.1 (iteración 2)

El TSF impide que los registros de auditoría sean borrados de forma no autorizada porque el KTS carece de funcionalidad para borrar registros de la base de datos de auditoría. Desde las aplicaciones KeyOne no es posible borrar registro alguno de cualquiera de las bases de datos que utilizan dichas aplicaciones.

6.1.2.2.2 FAU_STG.1.2 (iteración 2)

Como se ha explicado en la sección Función de verificación de la integridad de la base de datos, página 114, la función FUNC_DBIV puede detectar modificaciones en los registros de la base de datos, y por tanto permite detectar modificaciones en los registros de auditoría.

6.1.2.2.3 FDP_SDI_CIMC.3.1

Este requisito obliga a garantizar el servicio de integridad a las claves públicas guardadas en el CIMC (mediante firmas digitales, hashes con clave o códigos de autenticación) pero no dentro de un módulo criptográfico validado FIPS 140-1.

La clave pública certificada se protege mediante la firma digital relacionada con el certificado. Si el certificado es un certificado raíz, porque los certificados de confianza se guardan en una base de datos i3D, los mecanismos de integridad i3D garantizan la integridad del servicio de seguridad.

En la comunicación entre la RA y la CA de una clave pública no certificada, la integridad de la clave se garantiza mediante el formato firmado usado para la comunicación (lote KeyOne, véase la sección FDP_SDI_CIMC.3.1, página 150) y por el mecanismo de integridad aportado por el protocolo SSL/TLS (véase la sección FDP_SDI_CIMC.3.1, página 150). Cuando la clave pública se guarda en la base de datos del sistema KeyOne, se protege mediante la integridad garantizada por la base de datos i3D.

La función de gestión de la tabla de sesión (FUNC_I3DSESSION) genera la firma digital asimétrica a partir del certificado de firma digital del usuario. Como se explica en la sección Inicio de una , página 112, una a posible modificación fraudulenta en una sesión intermedia puede detectarse al verificar la integridad de la base de datos. En concreto, se verifica el vínculo de todos los registros de la tabla de sesión: la firma asimétrica del registro de inicio de sesión incluye el valor del campo *signature* del anterior registro de inicio de sesión; la firma asimétrica del registro de final de sesión

incluye el valor del campo `signature` del correspondiente registro de inicio de sesión.

La función de gestión de la tabla de histórico (FUNC_I3DHISTORIC) genera, mediante la clave de sesión, una firma digital simétrica (HMAC) de cada registro de la tabla de registros históricos. Además, como se explica en la sección Operaciones sobre la , página 113, los registros añadidos a la tabla de browsing se asocian a la tabla de histórico mediante el campo `hmac` insertado en el registro histórico asociado.

6.1.2.2.4 FPT_STM.1.1 (iteración 2)

Este requisito obliga a garantizar sellos de tiempo fiables para uso propio.

La fiabilidad se garantiza mediante un reloj de sistema sincronizado con un cliente NTP instalado en la misma máquina que el servidor KeyOne. Dicho componente NTP se sincroniza con un reloj fiable que obtiene el Tiempo Universal Coordinado de una fuente segura. La comunicación se realiza mediante el protocolo NTP (Network Time Protocol).

La funcionalidad de sellado se cumple porque la base de datos (donde se guarda la relación entre los datos y el tiempo) aporta el servicio de integridad. La función de gestión de la tabla de sesión (FUNC_I3DSESSION) y la función de gestión de la tabla de histórico (FUNC_I3DHISTORIC) garantizan las funciones de integridad que sustentan la funcionalidad de integridad de la base de datos i3D.

6.1.2.2.5 FPT_CIMC_TSP.1.1

Este requisito obliga a garantizar la creación del siguiente evento de firma de registro de log:

- Debe calcularse una firma digital, hash con clave, o código de autenticación sobre todas las entradas del log de auditoría.
- La firma digital, hash con clave o código de autenticación debe calcularse al menos sobre cada entrada añadida al registro de auditoría desde el último evento de firma del registro de auditoría y sobre la firma digital, hash con clave o código de autenticación generado en el último evento de firma de registro.
- La firma digital, hash con clave, o código de autenticación generado en el evento de firma de registro de auditoría debe incluirse en el registro de auditoría.

Este requisito se cumple mediante la Función de Gestión de Tabla de Sesión (FUNC_I3DSESSION). Esta función genera una firma digital asimétrica firma digital mediante el certificado de firma digital del usuario. Como se explica en la sección Inicio de una , página 112, los registros de la tabla de sesión se vinculan de la siguiente manera:

- La firma asimétrica del registro de inicio de sesión incluye el valor del campo `signature` del anterior registro de inicio de sesión.
- La firma asimétrica del registro de final de sesión incluye el valor del campo `signature` del correspondiente registro de inicio de sesión.

Cuando se cierra una sesión i3D, se modifica el registro de final de sesión que fue insertado en la tabla sesión al iniciar la sesión. En concreto, el hash acumulado de los registros de histórico generados durante la sesión se añaden a este registro. Una vez



actualizado, el registro de final de sesión se firma de nuevo asimétricamente con el certificado de firma digital del usuario y la sesión se considera una sesión concluida.

6.1.2.2.6 FPT_CIMC_TSP.1.2

Este requisito obliga a garantizar la creación del siguiente evento de firma de registro de auditoría:

- Debe calcularse una firma digital , hash con clave o código de autenticación sobre las entradas del registro de auditoría.
- La firma digital, hash con clave o código de autenticación deben calcularse al menos sobre cada entrada añadida al registro de auditoría desde el último evento de firma de registro de auditoría, y sobre la firma digital, hash con clave o código de autenticación realizado en el último evento de firma de registro de auditoría.
- La firma digital, hash con clave o código de autenticación del evento de firma de registro de auditoría debe incluirse en el registro de auditoría.

Este requisito se cumple mediante la Función de Gestión de la Tabla de Sesión (FUNC_I3DSESSION). Dicha función genera una firma digital asimétrica a partir del certificado de firma digital del usuario. Como se explica en la sección Inicio de una , página 112, los registros de la tabla de sesión se vinculan de la siguiente manera:

- La firma asimétrica del registro de inicio de sesión incluye el valor del campo *signature* del anterior registro de inicio de sesión.
- La firma asimétrica del registro de final de sesión incluye el valor del campo *signature* del correspondiente registro de inicio de sesión.

Cuando se cierra una sesión i3D, el registro de final de sesión que fue insertado al iniciar la sesión es modificado. En concreto, el hash acumulado de todos los registro de histórico generados durante la sesión se añaden a este registro. Una vez actualizado, el registro de final de sesión se firma asimétricamente de nuevo con el certificado de firma digital del usuario y la sesión se considera una sesión concluida.

6.1.2.2.7 FPT_CIMC_TSP.1.4

Este requisito obliga a garantizar la creación del siguiente evento de firma de registro de auditoría:

- Debe calcularse una firma digital, hash con clave, o código de autenticación sobre todas las entradas del log de auditoría.
- La firma digital, hash con clave o código de autenticación debe calcularse al menos sobre cada entrada añadida al registro de auditoría desde el último evento de firma del registro de auditoría y sobre la firma digital, hash con clave o código de autenticación generado en el último evento de firma de registro.
- La firma digital, hash con clave, o código de autenticación generado en el evento de firma de registro de auditoría debe incluirse en el registro de auditoría.

Este requisito se cumple mediante la Función de Gestión de la Tabla de Sesión (FUNC_I3DSESSION). Esta función genera la firma digital asimétrica a partir del

certificado de firma digital. Como se explica en la sección Inicio de una , página 112, los registros de la tabla de sesión se vinculan del siguiente modo:

- La firma asimétrica del registro de inicio de sesión incluye el valor del campo `signature` del anterior registro de inicio de sesión.
- La firma asimétrica del registro de final de sesión incluye el valor del campo `signature` del correspondiente registro de inicio de sesión.

Cuando se cierra una sesión i3D, se modifica el registro de final de sesión que se insertó en la tabla de sesión al iniciar la sesión. En concreto, el hash acumulado de todo los registros de histórico generados durante la sesión se añaden a este registro. Una vez actualizado, el registro de final de sesión se firma asimétricamente de nuevo con el certificado de firma digital del usuario y la sesión se considera una sesión concluida.

6.1.2.2.8 FAU_STG.4.1

Este requisito obliga a que, cuando el registro de auditoría está lleno, sólo los eventos realizados por el auditor sean registrados.

Este requisito especifica el comportamiento del TOE cuando el registro de auditoría está lleno. En concreto, este requisito establece que, para cualquier instanciación del mismo, el usuario autorizado con derechos específicos para este efecto (Auditor), puede seguir generando eventos auditables (acciones). De otro modo, podría darse que un usuario autorizado ni siquiera puede reiniciar el sistema.

En este caso, si el registro de auditoría está lleno, la Función de Verificación de la Capacidad de la Base de Datos (FUN_CDBC) se hace cargo de la situación. En concreto, esta función se ocupa de las siguientes tareas:

- Generación de un log auxiliar blindado en disco.
- Roll-back de las acciones relacionadas con el evento.
- Detención del sistema.

Cuando un administrador intenta iniciar el sistema, se verifica la capacidad de la base de datos (sólo se iniciará con éxito cuando la base de datos tenga asignada la capacidad necesaria para generar las entradas de registro iniciales). Si los servidores KeyOne se detienen, no se generan más entradas de registro relacionadas con los eventos auditables.

El requisito establece que el usuario autorizado con derechos específicos (Auditor), puede seguir generando eventos auditables, para reiniciar el sistema o reparar el problema de capacidad de la base de datos. En ese caso, el usuario con rol de Auditor puede realizar las siguientes tareas:

- Examinar el registro auxiliar.
- Examinar la tabla de auditoría desde el sistema gestor de la base de datos.

De otro modo, al iniciar el servidor KeyOne, el usuario con rol Auditor no puede tener más información sobre la situación para reparar el problema.



6.1.2.2.9 FPT_CIMC_TSP.1.3

Este requisito obliga a que la frecuencia del evento de firma de registro de auditoría sea configurable.

El evento de firma de registro de auditoría es generado por la Función de Gestión de la Tabla de Sesión (FUNC_I3DSESSION). Dicha función genera la firma digital asimétrica a partir del certificado de firma digital del usuario. Como se explica en la sección Inicio de una , página 112, los registros de la tabla de sesión se vinculan del siguiente modo:

- La firma asimétrica del registro de inicio de sesión incluye el valor del campo *signature* del anterior registro de inicio de sesión.
- La firma asimétrica del registro de final de sesión incluye el valor del campo *signature* del correspondiente registro de inicio de sesión.

Cuando se cierra una sesión i3D, se modifica el registro de final de sesión que fue insertado en la tabla de sesión al iniciar la sesión. En concreto, el hash acumulado de todos los registros de histórico generados durante la sesión se añade a este registro. Una vez actualizado, el registro de final de sesión se firma asimétricamente con el certificado de firma digital del usuario y la sesión se considera una sesión concluida.

Además, cuando se añade un registro histórico (por inserción, actualización o supresión de un registro lógico), se genera una firma simétrica de dicho registro histórico que se añade a la tabla de histórico y al registro de browsing asociado. La firma simétrica de los registros de la base de datos es realizada por la función FUNC_I3DHISTORIC.

Para cada modificación en los registros de la base de datos (inserción, actualización o supresión), el mecanismo i3D genera una firma digital mecanismo que garantiza la integridad de la base de datos. Por ello, el sistema KeyOne funciona como si estuviese configurado para la máxima frecuencia, es decir, la frecuencia más segura (refinamiento del requisito FPT_CIMC_TSP.1.3).

6.1.2.2.10 FDP_CIMC_BKP.2.1

Este requisito obliga a proteger las copias de seguridad ante posibles modificaciones (mediante firma digital, hash con claves o códigos de autenticación).

El sistema KeyOne incluye una funcionalidad de copias de seguridad que hace una copia de seguridad de todo el sistema KeyOne necesario para reconstruir el estado actual a partir de esa copia de seguridad y el mismo software usado para la instalación inicial del sistema KeyOne. Los datos almacenados en la copia de seguridad del sistema para recuperar el estado del sistema en el momento de realizar la copia incluyen toda la información almacenada en la base de datos KeyOne. Dicha información está protegida mediante el mecanismo i3D de KeyOne y las funciones de seguridad FUNC_I3DSESSION y FUNC_I3DHISTORIC.

El evento de firma de registro de auditoría es generado por la Función de Gestión de la Tabla de Sesión (FUNC_I3DSESSION). Dicha función genera una firma digital asimétrica a partir del certificado de firma digital del usuario. Como se explica en la sección Inicio de una , página 112, los registros de la tabla de sesión se vinculan del siguiente modo:

- La firma asimétrica del registro de inicio de sesión incluye el valor del campo `signature` del anterior registro de inicio de sesión.
- La firma asimétrica del registro de final de sesión incluye el valor del campo `signature` del correspondiente registro de inicio de sesión.

Cuando se cierra una sesión i3D, se modifica el registro de final de sesión que fue insertado en la tabla de sesión al iniciar la sesión. En concreto, el hash acumulado de todos los registros de histórico generados durante la sesión se añade a este registro. Una vez actualizado, el registro de final de sesión se firma de nuevo asimétricamente con el certificado de firma digital del usuario y la sesión se considera una sesión concluida.

Además, cuando se añade un registro histórico (inserción, actualización o supresión de un registro lógico), se genera una firma simétrica del registro de dicho histórico que se añade a la tabla de histórico y al registro de browsing correspondiente. La firma digital del registro de la base de datos es realizada por la función `FUNC_I3DHISTORIC`.

Para cada modificación (inserción, modificación o supresión) de un registro de la base de, el mecanismo i3D genera una firma digital que garantiza la integridad de la base de datos. Por ello, mediante firmas digitales, la copia de seguridad del sistema guardada en la base de datos queda protegida de modificaciones.

6.1.3 Gestión del Control de Acceso

La tecnología KeyOne usa un control de acceso basado en gestión de roles cuando un usuario intenta acceder a las funciones del TOE que gestionan recursos KeyOne.

Dependiendo de la política de seguridad, los nombres de los roles KeyOne pueden ser diferentes. De este modo, si se utiliza la política de seguridad CIMC, el rol Administrador (Administrator) sería el rol Administrador de Sistemas (System Administrator) de la política CWA; el rol Oficial (Officer) del CIMC sería el rol Oficial de Registro (Registration Officer) de la política CWA; mientras que el rol Auditor de la política CIMC sería el rol Auditor del Sistema (System Auditor) de la política CWA.

El rol Operador del Sistema (System Operator) no es un rol KeyOne en el sentido estricto, ya que no hay ninguna funcionalidad de aplicación asignada a este rol. De hecho, cuando se habla de usuarios con el rol Operador del Sistema no nos referimos a los usuarios registrados como tal en el sistema KeyOne; es una manera de designar usuarios que pueden proporcionar secretos requeridos para hacer funcionar la aplicación. Desde este punto de vista, se considera Operadores del Sistema como meros recursos cuya presencia es necesaria para arrancar la aplicación.

6.1.3.1 Usuarios, grupos y roles

Los usuarios de las aplicaciones KeyOne pertenecen a uno o más grupos. Tanto los grupos como los usuarios se definen para todo el sistema KeyOne (i.e. para todas las aplicaciones que forman el sistema). A cada grupo de usuarios se le pueden asignar uno o más roles específicos de cada aplicación. No se pueden asignar directamente roles a usuarios individuales. Cada rol de una aplicación KeyOne representa un conjunto de permisos sobre funciones de esa aplicación. Este conjunto de permisos



se establece (i.e. otorga) al cargar la política de seguridad (seleccionada durante la puesta en marcha de KeyOne).

Todos los usuarios de aplicación y grupos de usuarios que componen el sistema KeyOne deben ser creados desde el a aplicación KeyOne Console. No obstante, KeyOne Console sólo puede asignar roles a los grupos definidos en KeyOne Console. Por ello, una vez se han creado los usuarios y grupos del sistema, la asignación de roles para cada aplicación debe hacerse desde la misma aplicación. Esto es, la asignación de roles para una instancia de KeyOne CA debe realizarse desde la misma instancia.

Todos los **usuarios** del sistema KeyOne aplicación deben crearse desde la aplicación KeyOne Console. De este modo, KeyOne Console se usa para registrar todos los usuarios de sistema y no sólo los que son específicos de KeyOne Console.

Un **grupo de usuarios** es un conjunto de usuarios a los que se pueden asignar roles en diferentes aplicaciones del sistema KeyOne. Los grupos de usuarios son pues un mecanismo para asignar roles a usuarios del sistema. De esta forma, un usuario posee en cada aplicación del sistema KeyOne aplicaciones, todos los roles que se hayan asignado al grupo al que pertenece.

Los grupos de usuario del sistema KeyOne deben crearse desde la aplicación KeyOne Console. De este modo, KeyOne Console se utiliza para registrar todos los grupos de usuario y no sólo los grupos específicos de KeyOne Console.

Algunos grupos de usuario se tratan de manera diferente:

- Grupos iniciales

Los grupos iniciales son grupos creados al cargar la política de seguridad durante la puesta en marcha de KeyOne Console. Más adelante, pueden añadirse grupos adicionales, salvo si la política lo prohíbe.

- Grupos principales pertenecientes a aplicaciones KeyOne

Los grupos principales de las aplicaciones KeyOne son subconjuntos de grupos iniciales para los que se han establecido las siguientes restricciones:

- Los grupos principales de KeyOne Console deben tener al menos un usuario habilitado.
- Ni en los grupos principales de KeyOne Console ni en los del resto de aplicaciones pueden eliminarse roles asignados por la política de seguridad. La política puede permitir asignar roles adicionales a los grupos principales. No obstante, el sistema no permite borrar ninguno de los roles asignados por la política.

La política de seguridad seleccionada durante la puesta en marcha de KeyOne Console define los grupos principales del sistema y cada una de las políticas. Cada política define al menos los principales grupos principales:

- Para KeyOne Console:
 - Grupo *Administradores*, usualmente llamado **System Administrators** (ADMIN_GROUP).

- Grupo Oficiales de Seguridad, usualmente llamado **Security Officers** (MAIN_GROUP).

Los usuarios que intervienen en la puesta en marcha de KeyOne Console (i.e. usuarios iniciales) son automáticamente asignados al resto de los grupos.

- Para el resto de aplicaciones KeyOne:
 - Grupo Administradores, usualmente llamado **System Administrators** (ADMIN_GROUP).
 - Grupo Oficiales de Seguridad, usualmente llamado **Security Officers** (MAIN_GROUP).

El usuario que crea una aplicación KeyOne que no sea KeyOne Console debe pertenecer al segundo grupo. El grupo principal de una aplicación KeyOne puede ser el mismo que el de KeyOne Console (i.e. llamarse igual).

Todos los usuarios de KeyOne Console usuarios poseen uno o más **roles**. Dichos roles son parte de la configuración de KeyOne Console y se inicializan a partir de los valores definidos en la política de seguridad seleccionada durante la puesta en marcha.

Los roles no pueden asignarse directamente a usuarios individuales sino a grupos. De esta manera, los usuarios poseen los roles asignados a los grupos a los que pertenecen.

Cada rol de KeyOne Console representa un conjunto de permisos específicos sobre las funciones de la aplicación. Dicho conjunto de permisos se establece cuando se carga la política de seguridad seleccionada durante la puesta en marcha de KeyOne Console, y no puede modificarse una vez establecida.

Un usuario de KeyOne Console puede tener varios roles (siempre que no sean incompatibles).

Cada uno de los roles tiene un objetivo específico definido en KeyOne Console y, por tanto, un conjunto específico de privilegios para ejecutar las funciones de KeyOne Console.

Pueden definirse incompatibilidades entre roles para evitar que un usuario acceda a todas las funciones de KeyOne Console.

6.1.3.2 Control del acceso a las funciones de KeyOne

El control de acceso realizado por las aplicaciones KeyOne se basa en la posibilidad de que el usuario realice una determinada operación.

La relación entre las soft-pages de KeyOne y las operaciones se mantiene en un solo archivo de configuración firmado, y la relación entre las operaciones y los roles se establece en la política de seguridad.

El sistema KeyOne mantiene ACLs (Access Control Lists) gestionadas por las aplicaciones KeyOne:



- Cuando se carga una aplicación, el objeto ACL contiene información sobre la misma como los roles, usuarios, relaciones entre operaciones y roles, etc.
- Durante el proceso de login, dicho objeto contiene información sobre el usuario como los roles que tiene asignados.

El motor WinScryptor asocia las operaciones con las soft-pages KeyOne, y carga la información en memoria.

La función `TestAction/CheckAction` (método perteneciente al objeto ACL) usa la información del objeto ACL y el motor WinScryptor, y determina si el usuario puede o no ejecutar una cierta soft-page KeyOne. Todas las funciones KeyOne a las que puede acceder un usuario deben ejecutar el método `TestAction/CheckAction`.

Respecto a las acciones del motor WinScryptor, siempre requieren una invocación previa al método `TestAction/CheckAction` (el motor WinScryptor siempre ejecuta el método `.runMethod` que invoca la función `testAllowed` que a su vez siempre invoca la función `TestAction/CheckAction`).

También las acciones configurables deben incorporar una invocación a la función `TestAction/CheckAction`. Para que un código personalizado se ejecute debidamente, el motor del servidor KeyOne exige la invocación de la función `TestAction/CheckAction`.

6.1.3.3 Requisitos Funcionales que cumplen las Funciones de Seguridad

Los Servicios de Control de Acceso se componen de las siguientes funciones de seguridad:

- Función de control de acceso (`FUNC_ACCESSCTRL`). Esta funcionalidad permite controlar del TOE a las funciones mediante roles asignados al usuario.

Este servicio cumple los siguientes requisitos:

6.1.3.3.1 FMT_MOF.1.1 (iteración 2)

La Función de Control de Acceso permite restringir la funcionalidad indicada en este requisito a los roles incluidos en la siguiente tabla:

Sección/Función	Componente	Función/Rol Autorizado
Auditoría de Seguridad		Sólo los Administradores deben poder configurar los parámetros de auditoría.
Copia de seguridad y recuperación		Sólo los Administradores deben poder configurar los parámetros de copia de seguridad. Sólo los Administradores deben poder iniciar la función de copia de seguridad o recuperación.

Registro de Certificados		<p>Sólo los Oficiales deben poder aprobar los campos o extensiones que se incluirán en un certificado.</p> <p>Si se utiliza un proceso automatizado para aprobar los campos o extensiones que se incluirán en un certificado, sólo los Oficiales deben poder configurar dicho proceso.</p>
Exportación y salida de datos		La exportación de las claves privadas CIMC puede requerir la autorización de al menos dos Administradores o un Administrador y un Oficial, Auditor u Operador.
Aprobación de cambio de estado de un certificado		<p>Sólo los Oficiales deben poder configurar el proceso automatizado que aprueba la revocación de un certificado o informa sobre la revocación de un certificado.</p> <p>Sólo los Oficiales pueden configurar el proceso automatizado que aprueba la suspensión de un certificado, o informa sobre el estado de suspensión de un certificado.</p>
Configuración CIMC		Solo los Administradores deben poder configurar cualquier funcionalidad TSF. (Este requisito se cumple para todos los parámetros de configuración salvo si la capacidad de configurar alguna funcionalidad del TSF ha sido asignada a otro rol descrito en este documento).
Gestión de certificados	<p>FMT_MOF_CIMC.2 Gestión de certificados</p> <p>FMT_MOF_CIMC.3 Gestión extendida de los perfiles de certificado</p>	Sólo los Administradores deben poder modificar un perfil de certificado.
Gestión de perfiles de revocación		Sólo los Administradores deben poder modificar un perfil de revocación.
Gestión de los perfiles de Lista de Revocación de Certificados	<p>FMT_MOF_CIMC.4 Gestión de los perfiles de Lista de Revocación de Certificados</p> <p>FMT_MOF_CIMC.5 Gestión extendida de los perfiles de Lista de Revocación de Certificados</p>	Sólo los Administradores deben poder modificar Listas de Revocación de Certificados.
Gestión de perfiles de OCSP (Online Certificate Status)	FMT_MOF_CIMC.6 Gestión de perfiles de OCSP	Sólo los Administradores deben poder modificar perfiles OCSP.



Protocol)		
-----------	--	--

Tabla 6-1. Roles autorizados a modificar el comportamiento de las funciones de seguridad.

Las relaciones entre operaciones y roles se establecen en el archivo de la política de seguridad. Las funciones descritas en la tabla anterior afectan a las siguientes operaciones/privilegios (el rol con acceso a esas operaciones puede configurarse en el archivo de la política de seguridad):

- Configurar los parámetros de auditoría: VIEW_DATABASETREE, CREATE_DATABASE_TABLES, RENAME_DATABASE_TABLES, CHANGE_DATABASE_CONNECTION, VIEW_REGISTEREDDATABAS, EDIT_REGISTEREDDATABAS, EDIT_LOGS_REGISTER.
- Configurar los parámetros de la copia de seguridad: EXECUTE_SYSTEM_BACKUP.
- Iniciar la función de copia de seguridad o recuperación: EXECUTE_SYSTEM_BACKUP.
- Aprobar los campos o extensiones que se incluirán en un certificado: MODIFY_PROFILES.
- Exportación de las claves privadas CIMC: ISSUE_CERTIFICATES, KEY_RECOVERY. Las claves se exportan al Módulo de Seguridad Hardware al arrancar el sistema KeyOne. En concreto, el operador que introduce las tarjetas de autenticación correctas puede arrancar el sistema y por tanto exportar claves privadas CIMC al HSM.
- Configurar el proceso automatizado que aprueba la revocación de certificados o informa sobre la revocación de certificados. Este privilegio no se aplica porque la tecnología KeyOne no incluye un proceso automatizado para tal fin.
- Configurar el proceso automatizado que aprueba la suspensión de un certificado o informa sobre el estado de suspensión de un certificado. Este privilegio no se aplica porque la tecnología KeyOne no incluye un proceso automatizado para tal fin.
- Configurar cualquier funcionalidad TSF: BROWSE_USER, CREATE_USERS, DELETE_USERS, ENABLE_USER, VIEW_USER_PROPERTIES, EDIT_USER_PROPERTIES, BROWSE_GROUPS, CREATE_GROUP, DELETE_GROUP, VIEW_GROUP_PROPERTIES, EDIT_GROUP_PROPERTIES, EDIT_PASSWORD_RULES, VIEW_PASSWORD_RULES, EDIT_SYSTEM_CERTIFICATES, VIEW_SYSTEM_CERTIFICATES, EDIT_LOGON_TIMEOUT, VIEW_LOGON_TIMEOUT, VIEW_APP_CERTS, VIEW_APP_KEYS, VIEW_APP_CRLS, INSTALL_APP_ROOT, INSTALL_APP_CERT, REMOVE_APP_CERT, EXPORT_APP_CERT, INSTALL_APP_CRL, REMOVE_APP_CRL, EXPORT_APP_CRL, GENERATE_KEY_PARAMS, VIEW_APP_KEY_DEFS, EDIT_APP_KEY_DEFS, RENEW_APP_KEYS, GENERATE_APP_KEYS, REMOVE_APP_KEYS, CREATE_APPLICATION, SELECT_APP_CERTS, VIEW_CERT, MODIFY_PROFILES, VIEW_PRIVILEGES, VIEW_ROLE_COMPATIBILITIES, EDIT_ROLE_COMPATIBILITIES, VIEW_USER_ROLES, EDIT_USER_ROLES.
- Modificar un perfil de certificado: MODIFY_PROFILES.



- Modificar un perfil de revocación: `MODIFY_PROFILES`.
- Modificar un perfil de Lista de Revocación de Certificado: `MODIFY_PROFILES`.
- Modificar un perfil de OCSP: `MODIFY_OCSP_PROFILES`.

Al modificar la política de seguridad KeyOne se pueden asignar estos privilegios a los roles adecuado para que pueden ejecutar la correspondiente operación.

Mediante el método `TestAction/CheckAction` y la asignación de roles, la Función de Control de Acceso puede controlar el acceso de usuarios a acciones.

6.1.3.3.2 FDP_ACC.1.1 (iteración 2)

Para reforzar la política de seguridad, el control de acceso al sistema KeyOne se basa en las siguientes relaciones seguras

- Las soft-pages KeyOne se relacionan con operaciones definidas en un archivo de configuración firmado (`pssmanager.actions`).
- Las Operaciones se relacionan con roles en la política de seguridad KeyOne (`policies`).

Para determinar el acceso de un usuario a una función, la función `TestAction/CheckAction` utiliza la información cargada en el objeto ACL (roles asignados al usuario actual y relaciones entre operaciones y roles). El TOE garantiza la política de control de acceso en las siguientes entidades y objetos:

- Usuarios de aplicaciones KeyOne.
- Recursos gestionados por el sistema.
- Privilegios definidos por el sistema y que pueden asignarse a los roles de la aplicación.

6.1.3.3.3 FDP_ACF.1.1 (iteración 2)

Para garantizar la política de seguridad, el control de acceso del sistema KeyOne se basa en los siguientes atributos de seguridad:

- Identidad del sujeto.
- Conjunto de roles que el sujeto puede desempeñar.

La Función de Control de Acceso se basa en el objeto ACL para determinar el acceso de un usuario a la ejecución de una función KeyOne. Durante el proceso de login, se carga un objeto ACL con la información de usuario necesaria (identificación de usuario y roles asignados al usuario).

6.1.3.3.4 FDP_ACF.1.2 (iteración 2)

El control de acceso puede configurarse en KeyOne para cumplir las reglas especificadas en la siguiente tabla:

Sección/Función	Componente	Función/Rol Autorizado
-----------------	------------	------------------------



Introducción remota y local de los datos de una petición de certificado		Sólo los Oficiales y el solicitante del certificado deben poder introducir los datos de una petición de certificado.
Introducción local y remota de peticiones de revocación		Sólo los Oficiales y el titular del certificado que se desea revocar deben poder introducir peticiones de revocación de certificado.
Exportación y salida de datos		Sólo usuarios autorizados deben poder exportar y extraer datos confidenciales y que afectan a la seguridad.
Generación de claves	FCS_CKM.1 Generación de claves criptográficas	Sólo los Administradores deben poder solicitar la generación de claves del componente (usadas para proteger datos en más de una sesión o mensaje).
Carga de clave privada		Sólo los Administradores deben poder solicitar la carga de claves privadas del componente en módulos criptográficos.
Almacenamiento de clave privada		Solo los Oficiales deben poder solicitar el descifrado de las clave privada del titular de un certificado. El TSF no debe permitir el descifrado de claves privadas de titular de certificado que puedan usarse para firma digital. Deben ser necesarios al menos dos Oficiales o un Oficial y un Administrador, Auditor o Operador para descifrar la clave privada del titular de un certificado.
Introducción, borrado o almacenamiento de una clave pública de confianza		Sólo los Administradores deben poder cambiar (añadir, revisar, borrar) las claves públicas de confianza.
Almacenamiento de Clave Secreta		Sólo los Administradores deben poder solicitar la carga de claves secretas CIMC en módulos criptográficos.
Destrucción de Claves Privadas y Secretas		Sólo los Administradores, Auditores, Oficiales, y Operadores deben poder poner a cero claves privadas y secretas CIMC guardadas como texto.
Exportación de Claves Privadas y Secretas		Sólo los Administradores deben poder exportar una clave privada del componente. Sólo los Oficiales deben poder exportar

		<p>claves privadas de titular de certificado.</p> <p>Deben ser necesarios al menos dos Oficiales o un Oficial y un Administrador, Auditor o Operador para exportar la clave privada de un titular de certificado.</p>
Aprobación de cambio de estado de un certificado		<p>Sólo los Oficiales y el titular del certificado deben poder solicitar la suspensión de un certificado.</p> <p>Sólo los Oficiales deben poder eliminar la suspensión de un certificado.</p> <p>Sólo los Oficiales deben poder aprobar la suspensión de un certificado.</p> <p>Sólo los Oficiales y el titular del certificado deben poder solicitar la revocación de un certificado.</p> <p>Sólo los deben poder aprobar la revocación de un certificado y la información sobre la revocación del certificado.</p>

Tabla 6-2. Controles de Acceso.

Las funciones descritas en la tabla anterior afectan a las siguientes operaciones/privilegios (el rol con acceso a esas operaciones puede configurarse en el archivo de la política de seguridad):

- Introducción de los datos de una petición de certificado: `ISSUE_CERTIFICATES`.
- Introducción de los datos de una petición de revocación de certificado: `BROWSE_CERTS_DB`, `REVOKE_CERTIFICATES`.
- Exportación o extracción de datos confidenciales que afectan a la seguridad: `ISSUE_CERTIFICATES`, `KEY_RECOVERY`. Las claves se exportan al Módulo de Seguridad Hardware al arrancar el sistema KeyOne. En concreto, el operador que introduce las tarjetas de autenticación correctas puede arrancar el sistema y por tanto exportar claves privadas CIMC al HSM.
- Solicitar la generación de claves del componente (usadas para proteger datos en más de una sesión o mensaje): `GENERATE_KEY_PARAMS`, `RENEW_APP_KEYS`, `GENERATE_APP_KEYS`.
- Solicitar la carga de Claves Privadas del Componente en módulos criptográficos. Las claves del Módulo de Seguridad Hardware se exportan cuando se arranca el sistema KeyOne. En concreto, el operador que introduce las tarjetas de autenticación correctas puede arrancar el sistema y exportar las claves privadas del componente a módulos criptográficos.
- Solicitar el descifrado de la clave privada del titular de un certificado: `KEY_RECOVERY`.



- El TSF no debe permitir el descifrado de una clave privada de titular que pueda usarse para generar firmas digitales. El componente KeyOne Key Archive (situado en el producto KeyOne CA) permite filtrar el tipo de certificado (descarta los certificados de firma digital) que pueden archivarse en la base de datos de Key Archive (para permitir una posterior recuperación de la clave privada del titular).
- Solicitar el descifrado de la clave privada del titular de un certificado: KEY_RECOVERY. La aplicación KeyOne Key Archive impone la presencia de usuarios de dos roles para recuperar la clave privada del titular.
- Cambiar las claves públicas de confianza: EDIT_SYSTEM_CERTIFICATES, VIEW_SYSTEM_CERTIFICATES.
- Solicitar la carga de claves secretas CIMC en módulos criptográficos: las claves se exportan al Módulo de Seguridad Hardware cuando se arranca el sistema KeyOne. En concreto, el operador que introduce las tarjetas de autenticación correctas puede arranca el sistema y por tanto cargar las claves privadas del componente en módulos criptográficos.
- Poner a cero CIMC claves privadas y secretas guardadas como texto: el TOE nunca tiene las claves privadas y secretas CIMC como texto; por tanto, la función que pone a cero las claves privadas y secretas guardadas como texto no es aplicable. El sistema KeyOne permite acceder al Módulo de Seguridad Hardware, y por tanto gestiona las claves privadas y secretas CIMC. El TOE utiliza el módulo validado FIPS 140-2 para generar claves, almacenar claves y poner a cero claves (con el fin de destruirlas). No se guarda ninguna clave privada ni secreta CIMC en el sistema KeyOne, y el sistema KeyOne accede al HSM para realizar operaciones con este tipo de claves.
- Exportar una clave privada de componente: ISSUE_CERTIFICATES, KEY_RECOVERY. Las claves se exportan al Módulo de Seguridad Hardware cuando se arranca el sistema. En concreto, el operador que introduce las tarjetas de autenticación correctas puede arrancar el sistema y exportar claves privadas CIMC al HSM.
- Exportar la clave privada del titular de un certificado: ISSUE_CERTIFICATES, KEY_RECOVERY. La exportación de la clave privada del titular de un certificado requiere la autorización de dos roles.
- Solicitar la suspensión de un certificado: BROWSE_CERTS_DB, REVOKE_CERTIFICATES.
- Eliminar la suspensión de un certificado: BROWSE_CERTS_DB, REVOKE_CERTIFICATES.
- Aprobar la suspensión de un certificado: BROWSE_CERTS_DB, REVOKE_CERTIFICATES.
- Solicitar la revocación de un certificado: BROWSE_CERTS_DB, REVOKE_CERTIFICATES.
- Aprobar la revocación de un certificado y toda la información sobre la revocación del certificado: BROWSE_CERTS_DB, REVOKE_CERTIFICATES.

Mediante el objeto ACL, la función de control de acceso determina el acceso a un usuario a una función KeyOne. Durante el proceso de login, se carga el objeto ACL con la información de usuario necesaria (identificación de usuario y roles asignados al usuario).

6.1.3.3.5 FMT_MOF_CIMC.3.2

El sistema KeyOne puede configurarse para determinar que un rol pueda asignar valores válidos a los campos y extensiones de un certificado. El privilegio `MODIFY_PROFILES` puede ser asignado a un rol específico en la política de seguridad KeyOne.

Cuando se invoca la función de configuración de plantillas de certificación, la Función de Control de Acceso verifica los roles concedidos al usuario y los roles asignados al privilegio `MODIFY_PROFILES`.

6.1.3.3.6 FMT_MOF_CIMC.3.3

El sistema KeyOne puede ser configurado para determinar que un rol pueda asignar valores válidos a los campos y extensiones de un certificado. El privilegio `MODIFY_PROFILES` puede ser asignado a un rol específico en la política de seguridad KeyOne.

Cuando se invoca la función de configuración de la plantilla de certificación, la Función de Control de Acceso verifica los roles concedidos al usuario y los roles asignados al privilegio `MODIFY_PROFILES`.

6.1.3.3.7 FMT_MOF_CIMC.3.4

El sistema KeyOne puede configurarse para determinar que un rol específico pueda asignar valores válidos a los campos y extensiones de un certificado. El privilegio `MODIFY_PROFILES` puede ser asignado a un rol específico en la política de seguridad KeyOne.

Cuando se invoca la función de configuración de la plantilla de certificación, la Función de Control de Acceso verifica los roles concedidos al usuario y los roles asignados al privilegio `MODIFY_PROFILES`.

6.1.3.3.8 FMT_MOF_CIMC.5.2

El sistema KeyOne puede configurarse para determinar que un rol específico pueda asignar valores válidos a los campos y extensiones de una CRL. El privilegio `MODIFY_PROFILES` puede ser asignado a un rol específico en la política de seguridad KeyOne.

Cuando se invoca la función de configuración de la plantilla de CRL, la Función de Control de Acceso verifica los roles concedidos al usuario y los roles asignados al privilegio `MODIFY_PROFILES`.

6.1.3.3.9 FMT_MOF_CIMC.5.3

El sistema KeyOne puede configurarse para determinar que un rol específico pueda asignar valores válidos a los campos y extensiones de una CRL. El privilegio



MODIFY_PROFILES puede ser asignado a un rol específico en la política de seguridad KeyOne.

Cuando se invoca la función de configuración de la plantilla de CRL, la Función de Control de Acceso verifica los roles concedidos al usuario y los roles asignados al privilegio MODIFY_PROFILES.

6.1.3.3.10 FMT_MOF_CIMC.6.2

El sistema KeyOne puede configurarse para determinar que un rol específico pueda asignar valores válidos a algunos campos de los mensajes de respuesta OCSP.

El privilegio EDIT_VA_CONFIG puede ser asignado a un rol específico en la política de seguridad KeyOne. Cuando se invoca la función de configuración del mensaje de confirmación OCSP, la Función de Control de Acceso verifica los roles concedidos al usuario, y los roles asignados al privilegio EDIT_VA_CONFIG.

6.1.3.3.11 FMT_MOF_CIMC.6.3

El sistema KeyOne puede configurarse para determinar que un rol específico pueda asignar valores válidos a algunos campos de los mensajes de respuesta OCSP.

El privilegio EDIT_VA_CONFIG puede ser asignado a un rol específico en la política de seguridad KeyOne. Cuando se invoca la función de configuración del mensaje de confirmación OCSP, la Función de Control de Acceso verifica los roles concedidos al usuario, y los roles asignados al privilegio EDIT_VA_CONFIG.

6.1.3.3.12 FPT_RVM.1.1 (iteración 2)

La función `TestAction/CheckAction` gestiona el servicio de control de acceso de KeyOne. Este método pertenece al objeto ACL y utiliza información del objeto mismo y del motor WinScryptor para determinar si el usuario puede ejecutar una determinada soft-page KeyOne. Todas las funciones KeyOne que puede acceder un usuario requieren una invocación del método `TestAction/CheckAction`.

Respecto a las acciones del motor WinScryptor, siempre requieren una invocación previa al método `TestAction/CheckAction` (el motor WinScryptor siempre ejecuta el método `.runMethod` que invoca la función `testAllowed` que a su vez siempre invoca la función `TestAction/CheckAction`).

También las acciones configurables deben incorporar una invocación a la función `TestAction/CheckAction`. Para que un código personalizado se ejecute debidamente, el motor del servidor KeyOne exige la invocación de la función `TestAction/CheckAction`.

6.1.3.3.13 FDP_ACF.1.3 (iteración 2)

Este requisito no implica el uso de ningún mecanismo de control, y por tanto no tiene función de seguridad asociada en el TOE.

6.1.3.3.14 FDP_ACF.1.4 (iteración 2)

Este requisito no implica el uso de ningún mecanismo de control, y por tanto no tiene función de seguridad asociada en el TOE.

6.1.4 Identificación y Autenticación

Antes de arrancar cualquier aplicación KeyOne son necesarios procesos de identificación y autenticación. El TOE tiene información relacionada con estos procesos (usuarios, contraseñas, certificados, etc.) en un repositorio seguro (Private Secure Store) que ofrece integridad y confidencialidad (para datos reservados).

El servidor KeyOne garantiza sus procesos de autenticación mediante los mecanismos de seguridad explicados en la sección Comunicaciones seguras, página 142.

6.1.4.1 Autenticación de los usuarios iniciales

La autenticación del Administrador y el Oficial de Seguridad iniciales se realiza durante la puesta en marcha del sistema.

Autenticación del sistema inicial del Administrador de KeyOne

Inicialmente, Administrador utiliza el asistente de instalación para introducir la configuración básica (módulo criptográfico del sistema, base de datos del sistema y lector de tarjetas del usuario). Al final del proceso, el asistente solicita una contraseña, y la configuración básica se guarda con dicha contraseña. Después, en la fase de inicialización, el Oficial de Seguridad solicita su contraseña al Administrador para cargar la configuración básica. Cuando concluye la fase de inicialización, el Administrador quedará establecido como usuario de sistema (autenticación mediante contraseña).

Autenticación del sistema inicial del Oficial de Seguridad de KeyOne

Esta autenticación puede realizarse mediante las siguientes dos opciones:

- m) El Oficial de Seguridad posee una tarjeta inteligente inicializada por Safelayer (entregada con el CD) o emitida por otra CA.
- n) El Oficial de Seguridad no tiene una tarjeta inteligente. La primera vez que la introduce en el sistema, debe introducir un nombre de usuario y una contraseña para autenticarse en adelante. Esta opción no es posible si la política de seguridad prohíbe el uso de contraseñas.

6.1.4.2 Grupos de usuarios especiales

Los siguientes grupos de usuarios se gestionan de una forma especial.

6.1.4.2.1 Grupos definidos por la política

La política de seguridad define un conjunto mínimo de grupos de usuarios que tendrá el sistema y cada una de las aplicaciones. La política permite definir más grupos pero no eliminar o renombrar los grupos definidos por la política.

6.1.4.2.2 Grupos Principales

La política de seguridad define los grupos principales para el sistema y cada aplicación. Dichos grupos son un subconjunto de los definidos por la política, y se



gestionan de una cierta manera. Como mínimo, la política define los siguientes grupos principales:

- Para el sistema: Administradores y Oficiales de Seguridad. Los dos usuarios implicados en la puesta en marcha del sistema son automáticamente asignados a estos grupos.
- Para cada aplicación: Administradores y Oficiales de Seguridad (pueden ser los mismos grupos que los grupos principales del sistema). El usuario que crea la aplicación debe pertenecer al segundo grupo.

Las siguientes restricciones se aplican a estos grupos:

- Los grupos principales del sistema siempre deben tener un usuario autorizado.

Esta restricción garantiza que siempre sea posible arrancar KeyOne Console para resolver precisamente algún problema de arranque. Esta restricción no es necesaria en el caso de los principales grupos de las aplicaciones porque siempre es posible entrar en KeyOne Console y asignar usuarios.

- No es posible reducir los roles asignados por la política a los grupos principales.

Esta restricción garantiza que los grupos principales siempre estén autorizados a realizar las tareas pertinentes en caso de problema de arranque. En dicho caso, la restricción también se aplica a los grupos principales de las aplicaciones, porque para resolver algunos problemas de arranque es necesario entrar en la aplicación en modo a prueba de fallos, y la asignación de roles a grupos debe realizarse en la misma aplicación.

6.1.4.3 Modos de Autenticación

Los siguientes modos de autenticación están definidos:

- Certificado (tarjeta inteligente)
- Nombre de usuario y contraseña
- Nombre de usuario y contraseña de seguridad. Este modo sólo puede usarlo un Oficial de Seguridad para arrancar KeyOne Console. Esta contraseña de seguridad se genera automáticamente y se exporta a un archivo durante la fase de arranque del sistema. Además, debe guardarse de forma segura mediante un procedimiento externo. Nadie puede conocer la contraseña hasta que se recupera para su uso; en ese momento, se crea otra contraseña que también debe guardarse de forma segura.

La política de seguridad seleccionada puede restringir los modos de autenticación permitidos a ciertos usuarios. Por esta razón, la política de seguridad especifica una de las siguientes configuraciones:

- a) Autenticación mediante contraseña permitida
- b) Autenticación mediante contraseña prohibida

6.1.4.3.1 Autenticación mediante contraseña permitida

Esta configuración permite que todos los usuarios se autenticquen mediante contraseña e impone las siguientes restricciones:

- Siempre debe haber un usuario perteneciente al grupo principal de administradores del sistema. Dicho usuario debe estar autorizado a autenticarse mediante contraseña (además de este tipo de autenticación, puede usarse la autenticación mediante certificado).

Esto garantiza que un administrador siempre podrá entrar en KeyOne Console para solucionar problemas (e.g. reconfigurar la tarjeta inteligente utilizada para autenticar usuarios).

- Siempre debe haber un usuario perteneciente al grupo principal de Oficiales de Seguridad del sistema. Dicho usuario debe estar autorizado a autenticarse mediante tarjeta (además de este tipo de autenticación, puede usarse la autenticación mediante certificado). Otra posibilidad es usar una contraseña de seguridad guardada de forma segura.

Así se garantiza que un Oficial de Seguridad siempre pueda entrar en KeyOne Console para solucionar problemas (e.g. pérdida de su tarjeta inteligente, expiración de su certificado, etc.).

- El resto de usuarios puede autenticarse tanto por certificado como por contraseña.

6.1.4.4 Requisitos funcionales que cumplen las Funciones de Seguridad

Los servicios de Identificación y Autenticación se componen de las siguientes funciones de seguridad:

- Identificación de Usuario y Función de Autenticación (FUNC_UIDAUT). Esta funcionalidad puede identificar y autenticar al usuario mediante un nombre de usuario y una contraseña o certificado previamente asignado al usuario.

Estos servicios cumplen los siguientes requisitos:

6.1.4.4.1 FIA_UAU.1.1 (iteración 2)

Según como haya sido configurado, el usuario podrá autenticarse por contraseña o por prueba de posesión (tarjeta criptográfica)⁶. El procedimiento de autenticación que se usará debe indicarse seleccionando el valor adecuado en la lista de modos de autenticación. El contenido de la ventana de login cambiará según el valor seleccionado.

Autenticación mediante contraseña

Para indicar esta opción, el usuario debe seleccionar el valor "contraseña" en la lista `Authentication mode`.

⁶ La autenticación mediante certificado sólo puede seleccionarse si un lector de tarjeta ha sido configurado como lector primario del sistema.



En este paso, el sistema solicitará la siguiente información de identificación y autenticación:

- a) Nombre de usuario (`User name`).
- b) Contraseña del usuario (campo obligatorio `Password`).

El sistema verificará el nombre de usuario (identificación) y la contraseña (autenticación) que se hayan introducida.

Si ambos son correctos:

- Si la aplicación es KeyOne Console, se mostrará la pantalla principal de la aplicación donde el usuario puede realizar todas las funciones que le permitan cada uno de sus roles.
- Para el resto de aplicaciones KeyOne:
 - Se mostrará la pantalla para seleccionar la instancia de aplicación donde iniciar una sesión.
 - Una vez seleccionada la instancia, se mostrará la pantalla principal, donde el usuario puede realizar todas las funciones que le permiten sus roles.

No obstante, si el nombre (error de identificación) o la contraseña (error de autenticación) son erróneos, se mostrará un mensaje de error en la pantalla. En ese caso, el usuario deberá reescribir el nombre y la contraseña (si el número de intentos de autenticación infructuosos iguala o supera el máximo número de intentos permitido, el TOE impedirá nuevos intentos de autenticación).

En el procedimiento de login, el usuario puede abortar el proceso antes de concluir la fase de autenticación.

Autenticación mediante certificado

Para indicar esta opción, el usuario debe seleccionar el valor "card" en la lista `Authentication mode`.

En este paso, el sistema solicitará la siguiente información de identificación y autenticación:

- a) Introducción de la tarjeta en el lector de tarjeta configurado como lector primario del sistema.
- b) PIN de la tarjeta (campo obligatorio `PIN`).

El sistema validará el certificado de usuario (identificación: el certificado introducido es un certificado que el sistema ha registrado como certificado de un usuario autorizado), y verificará la posesión de la clave privada asociada al certificado mediante el mecanismo de Prueba de Posesión (autenticación).

Si ambos son correctos:

- Si la aplicación es KeyOne Console, se mostrará la pantalla principal de la aplicación donde el usuario puede realizar todas las funciones que le permitan cada uno de sus roles.



- Para el resto de aplicaciones KeyOne:
 - Se mostrará la pantalla para seleccionar la instancia de aplicación donde iniciar una sesión.
 - Una vez seleccionada la instancia, se mostrará la pantalla principal, donde el usuario puede realizar todas las funciones que le permiten sus roles.

No obstante, si el certificado (error de identificación) o la Prueba de Posesión (error de autenticación) son erróneos, se mostrará un mensaje de error en pantalla. En ese caso, el usuario deberá introducir de nuevo el certificado y el correspondiente PIN (si el número de intentos de autenticación infructuosos iguala o supera el máximo número de intentos permitido, el TOE impedirá nuevos intentos de autenticación).

En el proceso de login, el usuario puede abortar el proceso antes de concluir la fase de autenticación.

6.1.4.4.2 FIA_UAU.1.2 (iteración 2)

Según como haya sido configurado, el usuario podrá autenticarse por contraseña o por prueba de posesión (tarjeta criptográfica)⁷. El procedimiento de autenticación que se usará debe indicarse seleccionando el valor adecuado en la lista de modos de autenticación. El contenido de la ventana de login cambiará según el valor seleccionado.

Autenticación mediante contraseña

Para indicar esta opción, el usuario debe seleccionar el valor "contraseña" en la lista `Authentication mode`.

En este paso, el sistema solicitará la siguiente información de identificación y autenticación:

- a) Nombre de usuario (`User name`).
- b) Contraseña del usuario (campo obligatorio `Password`).

El sistema verificará el nombre de usuario (identificación) y la contraseña (autenticación) que se hayan introducida.

Si ambos son correctos:

- Si la aplicación es KeyOne Console, se mostrará la pantalla principal de la aplicación donde el usuario puede realizar todas las funciones que le permitan cada uno de sus roles.
- Para el resto de aplicaciones KeyOne:
 - Se mostrará la pantalla para seleccionar la instancia de aplicación donde iniciar una sesión.
 - Una vez seleccionada la instancia, se mostrará la pantalla principal, donde el usuario puede realizar todas las funciones que le permiten sus roles.

⁷ La autenticación mediante certificado sólo puede seleccionarse si un lector de tarjeta ha sido configurado como lector primario del sistema.



No obstante, si el nombre (error de identificación) o la contraseña (error de autenticación) son erróneos, se mostrará un mensaje de error en la pantalla. En ese caso, el usuario deberá reescribir el nombre y la contraseña (si el número de intentos de autenticación infructuosos iguala o supera el máximo número de intentos permitido, el TOE impedirá nuevos intentos de autenticación).

En el procedimiento de login, el usuario puede abortar el proceso antes de concluir la fase de autenticación.

Autenticación mediante certificado

Para indicar esta opción, el usuario debe seleccionar el valor "card" en la lista `Authentication mode`.

En este paso, el sistema solicitará la siguiente información de identificación y autenticación:

- a) Introducción de la tarjeta en el lector de tarjeta configurado como lector de tarjeta primario.
- b) PIN de la tarjeta (campo obligatorio PIN).

El sistema validará el certificado de usuario (identificación: el certificado introducido es un certificado que el sistema ha registrado como certificado de un usuario autorizado), y verificará la posesión de la clave privada asociada al certificado mediante el mecanismo de Prueba de Posesión (autenticación).

Si ambos son correctos:

- Si la aplicación es KeyOne Console, se mostrará la pantalla principal de la aplicación donde el usuario puede realizar todas las funciones que le permitan cada uno de sus roles.
- Para el resto de aplicaciones KeyOne:
 - Se mostrará la pantalla para seleccionar la instancia de aplicación donde iniciar una sesión.
 - Una vez seleccionada la instancia, se mostrará la pantalla principal, donde el usuario puede realizar todas las funciones que le permiten sus roles.

No obstante, si el certificado (error de identificación) o la Prueba de Posesión (error de autenticación) son erróneos, se mostrará un mensaje de error en pantalla. En ese caso, el usuario deberá introducir de nuevo el certificado y el correspondiente PIN (si el número de intentos de autenticación infructuosos iguala o supera el máximo número de intentos permitido, el TOE impedirá nuevos intentos de autenticación).

En el proceso de login, el usuario puede abortar el proceso antes de concluir la fase de autenticación.

6.1.4.4.3 FIA_UID.1.1 (iteración 2)

Según como haya sido configurado, el usuario podrá autenticarse por contraseña o por prueba de posesión (tarjeta criptográfica)⁸. El procedimiento de autenticación que se usará debe indicarse seleccionando el valor adecuado en la lista de modos de autenticación. El contenido de la ventana de login cambiará según el valor seleccionado.

Identificación mediante nombre de usuario (autenticación mediante contraseña)

Para indicar esta opción, el usuario debe seleccionar el valor "password" en la lista `Authentication mode`.

En este paso, el sistema solicitará la siguiente información de identificación y autenticación:

- a) Nombre de usuario (`User name`).
- b) Contraseña del usuario (campo obligatorio `Password`).

El sistema verificará el nombre de usuario (identificación) y la contraseña (autenticación) que se hayan introducida.

Si ambos son correctos:

- Si la aplicación es KeyOne Console, se mostrará la pantalla principal de la aplicación donde el usuario puede realizar todas las funciones que le permitan cada uno de sus roles.
- Para el resto de aplicaciones KeyOne:
 - Se mostrará la pantalla para seleccionar la instancia de aplicación donde iniciar una sesión.
 - Una vez seleccionada la instancia, se mostrará la pantalla principal, donde el usuario puede realizar todas las funciones que le permiten sus roles.

No obstante, si el nombre (error de identificación) o la contraseña (error de autenticación) son erróneos, se mostrará un mensaje de error en la pantalla. En ese caso, el usuario deberá reescribir el nombre y la contraseña (si el número de intentos de autenticación infructuosos iguala o supera el máximo número de intentos permitido, el TOE impedirá nuevos intentos de autenticación).

En el procedimiento de login, el usuario puede abortar el proceso antes de concluir la fase de autenticación.

Identificación mediante certificado (autenticación mediante prueba de posesión)

Para indicar esta opción, el usuario debe seleccionar el valor "card" en la lista `Authentication mode`.

En este paso, el sistema solicitará la siguiente información de identificación y autenticación:

⁸ La autenticación mediante certificado sólo puede seleccionarse si un lector de tarjeta ha sido configurado como lector primario del sistema.



- a) Introducción de la tarjeta en el lector de tarjeta configurado como lector primario del sistema.
- b) PIN de la tarjeta (campo obligatorio PIN).

El sistema validará el certificado de usuario (identificación: el certificado introducido es un certificado que el sistema ha registrado como certificado de un usuario autorizado), y verificará la posesión de la clave privada asociada al certificado mediante el mecanismo de Prueba de Posesión (autenticación).

Si ambos son correctos:

- Si la aplicación es KeyOne Console, se mostrará la pantalla principal de la aplicación donde el usuario puede realizar todas las funciones que le permitan cada uno de sus roles.
- Para el resto de aplicaciones KeyOne:
 - Se mostrará la pantalla para seleccionar la instancia de aplicación donde iniciar una sesión.
 - Una vez seleccionada la instancia, se mostrará la pantalla principal, donde el usuario puede realizar todas las funciones que le permiten sus roles.

No obstante, si el certificado (error de identificación) o la Prueba de Posesión (error de autenticación) son erróneos, se mostrará un mensaje de error en pantalla. En ese caso, el usuario deberá introducir de nuevo el certificado y el correspondiente PIN (si el número de intentos de autenticación infructuosos iguala o supera el máximo número de intentos permitido, el TOE impedirá nuevos intentos de autenticación).

En el proceso de login, el usuario puede abortar el proceso antes de concluir la fase de autenticación.

6.1.4.4.4 FIA_UID.1.2 (iteration 2)

Dependiendo de cómo se haya configurado el usuario éste será capaz de identificarse mediante un nombre o un certificado (tarjeta criptográfica)⁹. El procedimiento de identificación que vaya a utilizar debe seleccionarlo en la lista de modo de identificación. El contenido de la pantalla de *login* cambiará en función del valor que sea seleccionado.

Identificación mediante nombre de usuario (autenticación mediante contraseña)

Para utilizar esta opción, el usuario debe seleccionar el valor "password" en la lista `Authentication mode`.

En este paso el sistema pedirá la siguiente información para la autenticación:

- a) Nombre del usuario (campo obligatorio `User name`).
- b) Contraseña del usuario (campo obligatorio `Password`).

⁹ *Authentication through certificate can only be selected if a card-reader has been configured as the system's primary card-reader.*



El sistema verificará el nombre del usuario (identificación) y la contraseña (autenticación) que se hayan introducido.

Si ambos son correctos:

- Si la aplicación es KeyOne Console, se muestra la pantalla principal de la aplicación y se inicia una sesión en la que el usuario podrá realizar todas las acciones que están permitidas a cada uno de sus roles.
- Para el resto de las aplicaciones KeyOne:
 - Se muestra la pantalla para seleccionar la instancia de aplicación sobre la que se realizará la sesión.
 - Se muestra la pantalla principal de la aplicación y se inicia una sesión en la que el usuario podrá realizar todas las acciones que están permitidas a cada uno de sus roles.

Sin embargo, si el nombre (fallo en la identificación) o la contraseña (fallo en la autenticación) son erróneos, se mostrará un mensaje de error en la pantalla. En este caso el usuario tendrá que volver a introducir el nombre y la contraseña (si el número de intentos de autenticación fallidos alcanza o supera el máximo número de intentos permitidos, el TOE no permitirá intentos de autenticación adicionales).

En el procedimiento de *login*, el usuario puede abortar el proceso antes de que se complete la fase de identificación.

Identificación mediante certificado (autenticación mediante prueba de posesión)

Para utilizar esta opción, el usuario debe seleccionar el valor "card" en la lista *Authentication mode*.

En este paso el sistema pedirá la siguiente información de identificación/autenticación:

- a) Introducción de la tarjeta en el lector de tarjetas que se haya configurado en el sistema como lector de tarjetas primario.
- b) El PIN de la tarjeta (campo obligatorio PIN).

El sistema validará el certificado del usuario (identificación: el certificado introducido es un certificado que el certificado ha registrado como perteneciente a un usuario autorizado), y verificará que posee la correspondiente clave privada mediante un mecanismo de Prueba de Posesión (autenticación).

Si ambos son correctos:

- Si la aplicación es KeyOne Console, se muestra la pantalla principal de la aplicación y se inicia una sesión en la que el usuario podrá realizar todas las acciones que están permitidas a cada uno de sus roles.
- Para el resto de las aplicaciones KeyOne:
 - Se muestra la pantalla para seleccionar la instancia de aplicación sobre la que se realizará la sesión.



- Se muestra la pantalla principal de la aplicación y se inicia una sesión en la que el usuario podrá realizar todas las acciones que están permitidas a cada uno de sus roles.

Sin embargo, si el certificado (fallo en la identificación) o la Prueba de Posesión (fallo en la autenticación) son erróneos, se mostrará un mensaje de error en la pantalla. En este caso el usuario tendrá que volver a presentar la tarjeta que contiene el certificado e introducir de nuevo el PIN asociado (si el número de intentos de autenticación fallidos alcanza o supera el máximo número de intentos permitidos, el TOE no permitirá intentos de autenticación adicionales).

En el procedimiento de *login*, el usuario puede abortar el proceso antes de que se complete la fase de identificación..

6.1.4.4.5 FIA_USB.1.1 (iteración 2)

La función de identificación y autenticación de usuario, después de identificar y autenticar al usuario asocia los atributos de seguridad (grupos y roles) a los sujetos que actúen en nombre de ese usuario.

Cuando una aplicación KeyOne arranca, se cargan las propiedades de esta aplicación en el objeto ACL. Los atributos de seguridad relativos a la aplicación, como son los grupos que están definidos en la aplicación y los roles asignados a los grupos definidos, se cargan en el objeto ACL.

Si la autenticación es mediante la utilización de nombre de usuario y contraseña, cuando el usuario introduce su nombre, la función de identificación y autenticación de usuarios indexa, por medio del *hash* SHA1 de <nombre de usuario><contraseña>, el almacén privado seguro en el que se guarda la información sensible del sistema. En este caso, la función recupera la información almacenada sobre las propiedades del usuario que está intentando el *login* y carga en el objeto ACL la información de seguridad sobre dicho usuario, como los grupos a los que pertenece (el objeto ACL ya guarda la relación entre grupos y roles, por lo que se puede obtener la relación entre usuarios y roles).

Si la autenticación es mediante certificado entonces el usuario proporciona el certificado y el PIN de la tarjeta en el que dicho certificado está almacenado, entonces la función de identificación y autenticación de usuarios indexa, por medio del *hash* SHA1 del certificado, el almacén privado seguro en el que se guarda la información sensible del sistema. Para autenticar al usuario, el sistema genera una cadena aleatoria (64 bytes) y plantea al usuario una prueba desafío-respuesta. Si la verificación de la firma que el usuario genera con su clave pública es satisfactoria, entonces la función recupera la información sobre las propiedades del usuario que está intentando el *login* y carga en el objeto ACL la información de seguridad sobre dicho usuario, como los grupos a los que pertenece (el objeto ACL ya guarda la relación entre grupos y roles, por lo que se puede obtener la relación entre usuarios y roles).

6.1.5 Comunicaciones seguras

La implantación de mecanismos seguros se requiere cuando ocurre una comunicación que afecta a algún componente del KeyOne TOE. El TOE protege la transferencia de datos tanto entre componentes KeyOne, como entre un

componente KeyOne y un componente externo al TOE. Esta protección se consigue mediante protocolos seguros estándar (e.g. protocolo SSL/TLS, OCSP, ...) o mediante el uso de protocolo propietarios de KeyOne (e.g. lotes KeyOne, protocolo NDCCP, ...)

6.1.5.1 Lotes KeyOne

Los servicios de generación y de revocación de certificados implican una comunicación entre KeyOne LRA y KeyOne CA. Los mensajes que se intercambian durante este proceso de comunicación reciben el nombre de lotes KeyOne y tienen una sintaxis que incluye una firma digital con la finalidad de proporcionar servicios de autenticación, integridad y no repudio.

Además, estos mensajes se transportan sobre conexiones SSL/TLS. Por lo tanto, queda garantizada la confidencialidad, autenticidad e integridad de las transacciones entre KeyOne LRA y KeyOne CA.

Los lotes se clasifican en dos categorías, según el tipo de peticiones de los que proceden:

- Lotes CR: Lotes que contienen una petición de certificación.
- Lotes RR: Lotes que contienen una petición de revocación, suspensión o rehabilitación.

Los lotes son firmados digitalmente por su emisor y son verificados en el lado receptor por el receptor del lote.

La aplicación KeyOne LRA envía peticiones de certificación o de revocación a la aplicación KeyOne CA utilizando la estructura del lote KeyOne. La aplicación KeyOne CA envía los certificados emitidos o los resultados de la revocación a la aplicación KeyOne LRA utilizando la estructura del lote KeyOne.

Estas son las etapas del ciclo de vida de un lote KeyOne::

- El lote es creado por la RA. La RA establece la información genérica del lote y añade las peticiones de certificación (o las peticiones de revocación) al lote.
- Por razones de seguridad, la RA firma el lote después de añadir los datos. La firma de la RA permite a la CA reconocer a la RA que envía el lote y asegura que el lote no haya sido modificado durante su transmisión.
- La RA envía el lote a la CA
- La CA recibe el lote, valida su firma y realiza las operaciones solicitadas
- La CA añade al lote los resultados de las operaciones y/o modifica los datos existentes. No se genera ningún lote nuevo. Para una petición de certificación, los certificados generados se añaden al lote. Para una petición de revocación, se añade el resultado de la revocación al lote. La cadena de certificación de la CA y las CRLs se añaden siempre al lote.
- Por razones de seguridad, la CA firma el lote después de añadir los datos. La firma de la CA permite a la RA reconocer la CA que envía el lote y asegura que el lote no haya sido modificado durante su transmisión .
- La CA envía el lote a la RA.



- La RA recibe el lote, valida su firma y comprueba los resultados de las operaciones solicitadas.
- Si el lote contenía peticiones de certificación, la RA extrae los certificados generados de la respuesta.

El ciclo de vida del lote KeyOne puede resumirse como sigue: la RA genera el lote y le añade peticiones, la CA procesa estas peticiones y añade las respuestas al lote. Por lo tanto, la estructura del lote puede considerarse como formada por datos aportados por la RA, datos aportados por la CA, información genérica del lote y extensiones del lote.

6.1.5.1.1 Datos relacionados con KeyOne LRA

- Información genérica del lote

La información genérica identifica el lote, el tipo de su contenido y su estado actual. Los siguientes campos conforman la información genérica del lote:

- **batchid**: identificador del lote. Este campo identifica el lote entre todos los lotes generados por una Autoridad de Registro.
- **batchtype**: Tipo de lote. Este campo identifica el tipo de peticiones del lote. Todas las peticiones de un lote tienen que ser del mismo tipo. Un lote puede ser:
 - **CR**: Lote de certificación: contiene peticiones de certificación.
 - **RR**: Lote de revocación: contiene peticiones de revocación.
- **status**: Estado del lote. Este campo corresponde más o menos con la etapa del ciclo de vida en la que se encuentra el lote.
- Datos relacionados con KeyOne LRA

La Autoridad de Registro genera el lote y le añade datos para que sean procesados por la CA. Estos datos incluyen peticiones de certificación o de revocación así como otros datos informativos que son intercambiados.

Después de que estos datos se hayan añadido, la RA firma el lote, para garantizar que la CA los recibe sin que sean modificados por un tercero.

La información que la RA añade al lote es la siguiente:

- Información general:
 - **timereg**: Fecha y hora en la que el lote fue generado por la RA.
 - **rasubject**: Nombre distintivo de la RA que generó el lote.
 - **policiesReq**: Lista de todas las políticas de certificación que se solicitan. En un lote de certificación (CR) esta lista contiene los nombres de las políticas de certificación que aparecen en las peticiones de certificación. Si el lote no es un lote CR, este campo permanece vacío.
- CSRs

El dato más importantes que la RA añade al lote es la lista de peticiones.

- `csrReportSeq`: Lista de peticiones. En un lote CR contendrá peticiones de certificación. En un lote RR contendrá peticiones de revocación. En otros tipos de lote este campo puede estar vacío.
- Argumentos
Los argumentos son datos adicionales que la RA le envía a la CA. Estos datos adicionales pueden contener información que la RA necesite recuperar una vez que la CA haya procesado el lote o cualquier otra información que la CA pueda necesitar. Los campos son:
 - `scriptorGenericReq`: Datos que enviar a la CA.
- Firma
Después de añadir los datos al lote, la RA lo firma para asegurar que la CA recibe el lote sin que lo haya modificado un tercero. El único campo aquí es:
 - `rasignature`: Firma (detached) del lote, generada por la RA.

6.1.5.1.2 Datos relacionados con KeyOne CA

El lote es generado por la Autoridad de Registro y es enviado a la CA. La CA lee los datos del lote, los procesa y añade los resultados al lote: certificados si el lote es un lote CR, o resultados de revocación si el lote es un lote RR.

Como la RA, la CA también firma el lote después de añadir datos, para asegurar que la RA recibe el lote sin que se haya sido modificado por un tercero.

- Información
Los campos de información identifican la CA que procesó el lote y cuándo lo procesó. Los siguientes campos constituyen la información de la CA:
 - `timeresp`: Fecha y hora en la que el lote fue procesado por la CA.
 - `casubject`: Nombre distintivo de la CA que procesó el lote.
- Certificados
El dato más importantes que la CA añade al lote es la lista de respuestas. Se llama "Certificados" debido a la respuesta a un lote CR (lista de certificados), pero contiene una lista de respuestas de revocación si el lote que se ha procesado es un lote RR.
 - `certReportSeq`: Lista de respuestas.
- Argumentos
Los argumentos son datos adicionales que la RA le envía a la CA. La CA procesa estos datos y retorna otros datos en la respuesta. Los datos que la CA envía pueden ser los mismos datos recibidos (sin modificación) o cualquier otro dato generado como respuesta a los datos recibidos. Los campos son:
 - `scriptorGenericGrant`: Datos que enviar a la RA.
- Cadena de certificación



La CA añade su cadena de certificación completa al lote procesado: sus certificados propios, los certificados raíz (si ella misma no es raíz) y todos los certificados de las CAs subordinadas entre ella y la CA raíz (si hay alguna). Los siguientes campos conforman la cadena de certificación:

- `keyCertSignCertificates`: Lista de certificados para firmar certificados. Los certificados que sirvan simultáneamente para firmar certificados y CRLs también son incluidos. Si la CA no es raíz, esta lista contiene su certificado propio y todos los certificados de CA subordinada que haya entre ella y la CA raíz (si hay alguna). Por el contrario si la CA es raíz, esta lista está vacía.
- `keyCertSignRootCertificate`: Certificado para firmar certificados de la CA raíz. Si la CA es raíz este certificado es su certificado propio.
- `crlSignOnlyCertificates`: Lista de certificados para firmar CRLs. Si la CA no es raíz, esta lista contiene su certificado propio y todos los certificados de las CAs subordinadas que haya entre ella y la CA raíz (si hay alguna). Por el contrario, si la CA es raíz, esta lista está vacía. La lista también está vacía si todas las CAs tienen certificados que sirven simultáneamente para firmar certificados y CRLs.
- `crlSignOnlyRootCertificate`: Certificado para firmar CRLs de la CA raíz. Si la CA es raíz este certificado es su certificado propio.
- `digitalSignatureCertificates`: Lista de certificado para firma digital. Si la CA no es raíz, esta lista contiene su certificado propio y todos los certificados de las CAs subordinadas que haya entre ella y la CA raíz (si hay alguna). Por el contrario, si la CA es raíz, esta lista está vacía.
- `digitalSignatureRootCertificate`: Certificado de firma digital de la CA raíz. Si la CA es raíz este certificado es su certificado propio.
- CRLs

La CA también añade sus CRLs al lote procesado, con la finalidad de mantener RA actualizada con respecto a las CRLs. El campo es:

- `crls`: Lista de CRLs.
- Firma

Después de añadir todos los datos al lote, la CA lo firma para asegurar que la RA lo reciba sin que sea modificado por un tercero. El único campo aquí es:

- `casignature`: Firma del lote, si anexar los datos firmados, generada por la CA.

6.1.5.2 Mensajes NDCCP

NDCCP (protocolo Near Domain Cert-status Coverage) es un protocolo propietario de Safelayer que se utiliza en la comunicación entre un módulo Database Updater (en KeyOne VA) y un módulo Cert-status Server (en KeyOne CA). Esta comunicación tiene lugar para mantener actualizada en KeyOne VA la base de datos de estados de certificado.

Las principales características de este protocolo son:



- Utiliza HTTPS (HTTP protegido con TLS/SSL) como mecanismo de transporte de los mensajes que se intercambian. Por lo tanto, existe un canal de confianza entre el servicio de gestión de la revocación (KeyOne CA) y el servicio de estado de revocación (KeyOne VA).

De este modo, los mensajes NDCCP son incrustados dentro del cuerpo de los mensajes de petición y de respuesta de mensajes HTTP. Los mensajes utilizan los siguientes valores en la cabecera Content-Type:

- `application/x-safelayer-cert-status-req` para los mensajes NDCCP de petición.
- `application/x-safelayer-cert-status-resp` para los mensajes NDCCP de respuesta.
- Los mensajes del protocolo se codifican de forma textual (ASCII).

El mensaje NDCCP de petición contiene un identificador de petición y la respuesta correspondiente que es generada por KeyOne CertStatus también contiene un campo que remite al identificador de la petición asociada. Estos identificadores actúan como *nonce*, por lo que las peticiones y las respuestas quedan protegidas frente a los ataques de repetición.

La petición contiene la hora en la que se generó el mensaje, la firma y el nombre del firmante (nombre distintivo del campo `subject` del certificado del firmante).

Un mensaje de petición NDCCP tiene la siguiente sintaxis:

```
<sessionID>;<startFrom>;<timeRequest>
```

```
[UNSIGNED]
```

```
signature = <signature>
```

Donde:

<sessionID>: Identificador de la petición.

<startFrom>: Instante que considerar cuando se seleccionen certificados cuyo estado haya cambiado. Únicamente serán devuelta información sobre aquellos certificados cuyo estado haya cambiado con posterioridad a este instante de tiempo.

<timeRequest>: Instante en el que se realiza la petición.

<signature>: Firma digital de la parte del mensaje que precede la etiqueta [UNSIGNED].

El mensaje de respuesta NDCCP tiene la siguiente sintaxis:

```
<sessionID>;<nextDate>;<timeResponse>; <moreToCome>
```

```
<certInfo>
```

```
....
```

```
[UNSIGNED]
```



signature = <signature>

Donde:

<sessionID>: Identificador de la petición asociada.

<nextStartFrom>: Valor que se debe incluir en el campo <startFrom> de la siguiente petición.

<timeResponse>: Fecha de la respuesta.

<moreToCome>: Etiqueta indicando si todos los certificados que cumplen la condición request.<startFrom> de la petición han sido incluidos en la respuesta. En caso de que no, otra petición con el valor response.<nextStartFrom> en su campo response.<startFrom> deberá ser emitidos por el cliente.

- <certInfo>: Línea que contiene información sobre un certificado cuyo estado haya cambiado desde request.<startFrom>. Esta línea tiene la siguiente estructura:

<sn>;<status>;<revreason>;<revdate>;<invdate>;<certIss>;<holdCode>

Donde:

- <sn>: Número de serie del certificado.
- <status>: Estado actual del certificado.
- <revreason>: Razón de revocación.
- <invdate>: Fecha en la que se sospecha que se produjo la circunstancia que motivó la revocación del certificado
- <certIss>: Identificación del emisor del certificado sobre cuyo cambio de estado se informa
- <holdCode>: Código de la acción a realizar por quién tenga noticia de que el certificado está suspendido, al pretender utilizarlo.

El mensaje indica qué certificados han sido revocados/suspendidos mediante el campo <sn> (contenido en el campo <certInfo>). El Servicio de Estado de Revocación (KeyOne VA) solicita estados de certificados y el módulo KeyOne CertStatus consulta la base de datos de KeyOne CA (campo status de la tabla de certificados) y responde proveyendo el estado actual de los certificados incluidos en la respuesta NDCCP (campo certInfo.status de la respuesta). Puesto que el módulo KeyOne CertStatus obtiene información de estado de revocación de la base de datos de KeyOne CA, éste provee el estado actual de los certificados.

6.1.5.3 Requisitos funcionales que satisfacen las funciones de seguridad

Los servicios de comunicaciones seguras están compuestos de las siguientes funciones de seguridad:

- Función de firma de lotes (FUNC_BATCHSIG). Esta función genera un lote firmado proporcionando servicios de integridad, autenticidad y no repudio a los datos que contiene el lote. La firma consiste en un PKCS #7 que no anexa los datos

firmados y que contiene la cadena de certificación (excepto el certificado de la CA raíz).

- Función de verificación de lotes (FUNC_BATCHVER). Esta funcionalidad cubre la validación que KeyOne LRA/KeyOne CA realizan de los lotes que reciben de KeyOne CA/KeyOne LRA. Esta validación implica la verificación de la firma digital que incluye el lote KeyOne y la verificación de que quién realizó la firma está efectivamente autorizado para ello.
- Función de verificación NDCCP (FUNC_NDCCPVER). Esta funcionalidad cubre la validación que realiza KeyOne VA de los mensajes NDCCP que recibe de KeyOne CA. Esta validación implica la verificación de la firma digital que incluye el mensaje y la verificación de que quién realizó la firma está efectivamente autorizado para ello.
- Función de firma NDCCP (FUNC_NDCCPSIGCA). Esta funcionalidad genera un mensaje NDCCP firmado en el componente KeyOne CA, proporcionando los servicios de integridad, autenticidad y no repudio a los datos que el mensaje contiene. La firma consiste en un PKCS #7 que no anexa los datos firmados y que contiene la cadena de certificación (excepto el certificado de la CA raíz).
- Generación de respuestas OCSP (FUNC_OCSPRES). Esta función genera, en el componente KeyOne VA, una respuesta OCSP que satisface las especificaciones de IETF RFC 2560, después de recibir una petición OCSP procedente de un cliente.
- SSL/TLS entre los componentes KeyOne. Esta funcionalidad se encarga de establecer el protocolo SSL/TLS entre los componentes KeyOne. Este protocolo de seguridad se utiliza en la comunicación entre KeyOne LRA y KeyOne CA y también entre KeyOne CA y KeyOne VA.
- Función de ofuscación (FUNC_OBFUSCATION). Esta funcionalidad se encarga de proteger los datos sensibles que utiliza el sistema KeyOne (datos de usuarios, parámetros de seguridad, parámetros administrativos y otros) frente a su revelación o modificación no autorizada.

Estos servicios cumplen los siguientes requisitos:

6.1.5.3.1 FDP_ITT.1.1 (iteración 3)

El requisito FDP_ITT.1.1 (iteración 3) necesita la aplicación del servicio de integridad a los datos de usuario. Los datos de usuario pueden incluirse en las comunicaciones entre los componentes KeyOne LRA y KeyOne CA (e.g. información de registro, ...) y en las comunicaciones entre KeyOne CA y KeyOne VA (e.g. información sobre el estado de los certificados de usuario)

Respecto a la comunicación RA-CA cuando se solicita un perfil de certificación, dicho perfil puede contener algunos datos de usuario que sea necesario proteger frente a las modificaciones no autorizadas. Para proteger esta información, después de añadir todos los datos al lote, KeyOne LRA los firma para asegurar que la CA recibe el lote sin que sea modificado por un tercero. El campo *rasignature* contiene la firma del lote (sin anexar los datos del lote) generada por la RA. La generación de la firma digital del lote es provista por funcionalidad que proporciona la función FUNC_BATCHSIG.



Respecto a la comunicación CA-VA, cuando KeyOne CA envía información de revocación a KeyOne VA, el mensaje utilizado contiene información del usuario que debe protegerse frente a modificaciones no autorizadas. Para proteger esta información, después de añadir todos los datos al mensaje NDCCP, KeyOne CA lo firma para asegurar que KeyOne VA recibe el mensaje sin que lo modifique un tercero. El campo `signature` del mensaje NDCCP contiene la firma del mensaje (sin anexar los datos del mensaje) generada por la CA. La generación de la firma digital del mensaje es provista por funcionalidad que proporciona la función `FUNC_NDCCPSIGCA`.

Las comunicaciones entre KeyOne LRA y KeyOne CA (lote KeyOne utilizado como formato de datos) y entre KeyOne CA y KeyOne VA (mensaje NDCCP utilizado como formato de datos) utilizan el protocolo SSL/TLS (con autenticación de cliente) con la finalidad de proporcionar el servicio de integridad a estas comunicaciones. Esta funcionalidad es provista por la función `FUNC_K1SSLTLS`.

6.1.5.3.2 FDP_ITT.1.1 (iteración 4)

El requisito FDP_ITT.1.1 (iteración 4) necesita la aplicación del servicio de confidencialidad sobre los datos de usuario. Los datos de usuario pueden ser incluidos en la comunicaciones entre los componentes KeyOne LRA y KeyOne CA (e.g. información de registro, ...) y en la comunicación entre KeyOne CA y KeyOne VA (e.g. información sobre el estado de los certificados de usuario).

Las comunicaciones entre KeyOne LRA y KeyOne CA (lote KeyOne utilizado como formato de datos) y entre KeyOne CA y KeyOne VA (mensaje NDCCP utilizado como formato de datos) utilizan el protocolo SSL/TLS (con autenticación de cliente) con la finalidad de proporcionar el servicio de confidencialidad a estas comunicaciones. Esta funcionalidad es provista por la función `FUNC_K1SSLTLS`.

6.1.5.3.3 FDP_SDI_CIMC.3.1

Las claves públicas almacenadas dentro de un módulo criptográfico que es conforme con CIMC, aunque no con FIPS 140-1, se protegen frente a la modificación no detectable mediante el uso de firmas digitales.

- Si la clave pública ha sido certificada, entonces:
 - Si el certificado en cuestión es un certificado raíz, el servicio de integridad lo proporcionan los mecanismos de integridad de la tecnología i3D, puesto que los certificados de confianza se almacenan en una base de datos i3D.
 - Si el certificado en cuestión no es un certificado raíz, el servicio de integridad lo proporciona la firma digital que está contenida en el propio certificado X.509.
- Si la clave pública no ha sido certificada, entonces puede estar en los siguientes estados:
 - La clave pública puede estar contenida en una petición de certificado dentro de la base de datos. En este caso, el servicio de integridad siempre lo proporciona el mecanismo de integridad de la base de datos i3D (el servicio es provisto por las funciones `FUNC_I3DSESSION` y `FUNC_I3DHISTORIC`, tal como se explica en la sección FDP_SDI_CIMC.3.1, página 116).

- Si la petición está contenida en un lote KeyOne, dado que éste está firmado, proporciona mediante esta firma el servicio de integridad a todo su contenido.
- En la comunicación entre los componentes RA y CA, la integridad de los datos involucrados es proporcionada por medio de la firma digital del lote y por medio de los mecanismos de integridad que proporciona el protocolo SSL/TLS. La función FUNC_K1SSLTLS proporciona el establecimiento del protocolo SSL/TLS entre los componentes KeyOne.

Por lo tanto, si la clave pública está dentro de un lote KeyOne, la integridad de esta clave se protege con la firma digital que contiene el campo *rasignature* del lote (firma del lote generada por la RA y que no anexa el lote). La generación de la firma digital del lote es provista por la funcionalidad que proporciona la función FUNC_BATCHSIG.

6.1.5.3.4 FCO_NRO_CIMC.3.1

El requisito FCO_NRO_CIMC.3.1 se consigue por medio de la funcionalidad que proporcionan las funciones FUNC_BATCHSIG y FUNC_NDCCPSIGCA.

Este requisito necesita en todo momento del servicio de generación de prueba de origen para la información sobre el estado de certificados y para cualquier otra información que sea relevante para la seguridad.

Puesto que la comunicación entre KeyOne LRA y KeyOne CA involucra información sobre estado de certificados (petición de revocación de KeyOne LRA), este requisito requiere una prueba de origen en esta comunicación. En este caso la evidencia es provista por medio de la firma del lote KeyOne que realizan tanto el componente KeyOne CA como el componente KeyOne LRA. Esta firma se lleva a cabo por medio de la función FUNC_BATCHSIGCA. El campo *rasignature* del lote contiene la firma del lote que genera la RA, sin anexar el lote, y el campo *casignature* contiene la firma del lote que genera la CA, sin anexar el lote.

El sistema KeyOne es capaz de relacionar la identidad del originador de la información y el certificado del originado, con las porciones de la información que son relevantes para la seguridad sobre la que se aplica la evidencia. Esta evidencia es la firma del lote que consiste en un PKCS #7 que contiene la cadena de certificación (excepto el certificado de la CA raíz). La identidad del originador de la información se incluye tanto en los campos *rasubject/casubject* del lote (autor de la generación del lote) y el campo *subject* del certificado implicado en la firma del lote (este certificado se incluye en el lote). Los lotes generados por los componentes KeyOne LRA y KeyOne CA se almacenan en la tabla de lotes de la base de datos de KeyOne.

La comunicación entre KeyOne VA y KeyOne CA también involucra información sobre el estado de certificados. KeyOne CA envía a KeyOne VA información sobre aquellos certificados cuyo estado haya cambiado. La funcionalidad que proporciona la función FUNC_NDCCPSIGCA consiste en la generación de un mensaje NDCCP firmado, proporcionando los servicios de integridad, autenticidad y no repudio a los datos que están contenidos en el mensaje. Este mensaje se utiliza en la comunicación entre los componentes CA (KeyOne CA) y VA (KeyOne VA), y el mensaje de respuesta NDCCP firmada es generada por la Autoridad de Certificación.



El sistema KeyOne es capaz de relacionar la identidad del originador de la información y el certificado del originado, con las porciones de la información que son relevantes para la seguridad sobre la que se aplica la evidencia. Esta evidencia consiste en la firma del mensaje, que tiene el formato PKCS #7, sin anexar el mensaje. La identidad del originador de la información es proporcionada por el certificado involucrado en la firma del mensaje NDCCP (este certificado está incluido en el mensaje).

6.1.5.3.5 FCO_NRO_CIMC.3.2

El requisito FCO_NRO_CIMC.3.2 se consigue mediante la funcionalidad que proporcionan las funciones FUNC_BATCHSIG y FUNC_NDCCPSIGCA.

Este requisito necesita en todo momento del servicio de generación de prueba de origen para la información sobre el estado de certificados y para cualquier otra información que sea relevante para la seguridad.

Puesto que la comunicación entre KeyOne LRA y KeyOne CA involucra información sobre el estado de certificados (petición de revocación de KeyOne LRA), este requisito requiere una prueba de origen en esta comunicación. En este caso la evidencia es proporcionada por medio de la firma del lote que realizan tanto el componente KeyOne CA como el componente KeyOne LRA. La firma se lleva a cabo por medio de la función FUNC_BATCHSIGCA. El campo *rasignature* del lote contiene la firma del lote que genera la RA, sin anexar el lote, y el campo *casignature* contiene la firma del lote que genera la CA, sin anexar el lote.

El sistema KeyOne es capaz de relacionar la identidad del originador de la información y el certificado del originado, con las porciones de la información que son relevantes para la seguridad sobre la que se aplica la evidencia. Esta evidencia consiste en la firma del lote, la cual es un PKCS #7 que contiene la cadena de certificación (excepto el certificado de la CA raíz).). La identidad del originador de la información se incluye tanto en los campos *rasubject/casubject* del lote (autor de la generación del lote) y el campo *subject* del certificado implicado en la firma del lote (este certificado se incluye en el lote). Los lotes generados por los componentes KeyOne LRA y KeyOne CA se almacenan en la tabla de lotes de la base de datos de KeyOne.

La comunicación entre KeyOne VA y KeyOne CA también involucra información sobre el estado de certificados. KeyOne CA envía a KeyOne VA información sobre aquellos certificados cuyo estado haya cambiado. La funcionalidad que proporciona la función FUNC_NDCCPSIGCA consiste en la generación de un mensaje NDCCP firmado, proporcionando los servicios de integridad, autenticidad y no repudio a los datos que están contenidos en el mensaje. Este mensaje se utiliza en la comunicación entre los componentes CA (KeyOne CA) y VA (KeyOne VA), y el mensaje de respuesta NDCCP firmada es generada por la Autoridad de Certificación.

El sistema KeyOne es capaz de relacionar la identidad del originador de la información y el certificado del originado, con las porciones de la información que son relevantes para la seguridad sobre la que se aplica la evidencia. Esta evidencia consiste en la firma del mensaje, que tiene el formato PKCS #7, sin anexar el mensaje. La identidad del originador de la información es proporcionada por el certificado involucrado en la firma del mensaje NDCCP (este certificado está incluido en el mensaje).

6.1.5.3.6 FCO_NRO_CIMC.3.3

El requisito FCO_NRO_CIMC.3.3 se cumple por medio de la utilización de la funcionalidad que proporcionan las funciones FUNC_BATCHVER y FUNC_NDCCPVER.

FCO_NRO_CIMC.3.3 requiere la verificación de la prueba de origen para toda información que sea relevante para la seguridad.

Respecto a la función FUNC_BATCHVER, antes de procesar una petición de certificación/revocación, KeyOne CA verifica la evidencia generada por la RA. Si la verificación de la firma digital falla, entonces se genera un reporte de información y un registro de *log* y el lote no es procesado. KeyOne CA también verifica que el originador de la evidencia (incluida en el lote KeyOne) está autorizado para enviar peticiones de certificación/revocación. KeyOne LRA también verifica las firmas digitales contenidas en los lotes recibidos de KeyOne CA, rechazándolos cuando estas verificaciones fallan.

6.1.5.3.7 FCO_NRO_CIMC.4.1

El requisito FCO_NRO_CIMC.4.1 se consigue mediante el uso de la funcionalidad que proporciona la función FUNC_BATCHVER.

FCO_NRO_CIMC.4.1 requiere que el TSF, con respecto a los mensajes de registro de certificación que envía el titular del certificado, sólo acepte aquellos mensajes que estén protegidos con un código de autenticación, *hash* con clave o algoritmo de firma digital. El requisito FCO_NRO_CIMC.4.1 se cumple, puesto que KeyOne CA sólo acepta los lotes de certificación de una RA que contengan firmas digitales que puedan validar.

6.1.5.3.8 FCO_NRO_CIMC.4.2

The FCO_NRO_CIMC.4.2 requirement is accomplished by means the use of functionality offered by the FUNC_BATCHVER function.

Because KeyOne CA only accepts certification batches from a RA that contain digital signatures that can be validated, then FCO_NRO_CIMC.4.2 requirement is accomplished.

6.1.5.3.9 FDP_CIMC_CSE.1.1

El requisito FDP_CIMC_CSE.1.1 necesita que la información de estado de certificado sea exportada del sistema KeyOne en mensajes cuyo formato sea conforme con el establecido por el estándar X.509 para las CRLS y con el estándar que define RFC 2560.

La función FUNC_OCSPRES incluye la funcionalidad para generar las respuestas OCSP que KeyOne CA enviará a sus clientes. Este mensaje se genera como respuesta tras recibir y procesar la correspondiente petición OCSP. La respuesta OCSP generada por KeyOne VA es conforme al formato que se especifica en RFC 2560 (*Online Certificate Status Protocol – OCSP*). Esta respuesta incluye todos los campos obligatorios de la respuesta y es posible configurar algunos campos y extensiones para que sean incluidos en el mensaje.



6.1.5.3.10 FDP_CIMC_OCSP.1.1

La función FUNC_OCSPRES incluye la funcionalidad para generar las respuestas OCSP que KeyOne VA envía a los clientes. Este mensaje se genera como respuesta tras recibir y procesar la correspondiente petición OCSP. La respuesta OCSP generada por KeyOne VA es conforme al formato que se especifica en RFC 2560 (*Online Certificate Status Protocol – OCSP*). Esta respuesta incluye todos los campos obligatorios de la respuesta y es posible configurar algunos campos y extensiones para que sean incluidos en el mensaje.

Tal como el requisito FDP_CIMC_OCSP.1.1 requiere, durante el proceso de generación de la respuesta OCSP, KeyOne VA verifica (función FUNC_OCSPRES) que todos los campos obligatorios de la respuesta OCSP básica contienen valores que son acordes con IETF RFC 2560. Algunos de los elementos que se validarán son los siguientes:

- El campo `version` debe contener el valor 0.
- Si el campo `issuer` contiene un nombre nulo, la respuesta debe contener la extensión crítica `issuerAltName`.
- El campo `signatureAlgorithm` contiene el OID de un algoritmo de firma digital aprobado por FIPS.
- El campo `thisUpdate` indica el instante de tiempo en el que se sabe correcto el estado que se indica.
- El campo `producedAt` indica el instante de tiempo en el que el servidor OCSP firmó la respuesta.
- El instante especificado en `nextUpdate`, no debe ser anterior al que se indica en el campo `thisUpdate`.

6.1.5.3.11 FMT_MOF_CIMC.6.1

La función FUNC_OCSPRES incluye la funcionalidad para generar las respuestas OCSP que KeyOne VA envía a los clientes. Este mensaje se genera como respuesta tras recibir y procesar la correspondiente petición OCSP. La respuesta OCSP generada por KeyOne VA es conforme al formato que se especifica en RFC 2560 (*Online Certificate Status Protocol – OCSP*). Esta respuesta incluye todos los campos obligatorios de la respuesta y es posible configurar algunos campos y extensiones para que sean incluidos en el mensaje.

El componente KeyOne VA componen implementa funcionalidad para configurar algunos campos de los mensajes de respuesta OCSP.

Esta funcionalidad es proporcionada por la función FUNC_OCSPPROF. Tal como requiere FMT_MOF_CIMC.6.1, cuando KeyOne VA genera una respuesta OCSP, se verifica la consistencia de la respuesta generada con respecto al perfil OCSP definido.

6.1.5.3.12 FPT_ITC.1.1 (iteración 2)

El requisito FPT_ITC.1.1 (iteración 2) necesita la protección frente a la revelación no autorizada de los datos transmitidos desde el sistema KeyOne a un producto IT de confianza que sea remoto.

Los datos sensibles que gestiona el sistema KeyOne (datos de usuario, parámetros de seguridad, parámetros administrativos y otros) se protegen mediante la función de seguridad FUNC_OBFUSCATION. Todos estos datos se almacenan en una base de datos i3D de KeyOne y por lo tanto, son transmitidos por el sistema KeyOne a esta base de datos. La función FUNC_OBFUSCATION protege los datos sensibles que utilizan las aplicaciones KeyOne de la revelación y modificación no autorizada.

6.1.5.3.13 FDP_CIMC_CER.1.3

FDP_CIMC_CER.1.3 requiere que el sistema verifique, antes de emitir un certificado, que su futuro titular posee la clave privada que corresponde a la clave pública que consta en la petición de certificación, salvo que el par de claves sea generado por el TSF, y siempre que la clave privada permita realizar firmas digitales. En el caso de que la clave privada no se pueda utilizar para realizar firmas digitales, la verificación de que el titular posee la clave privada que corresponde a la clave pública que consta en la petición se lleva a cabo mediante la verificación de la firma del lote KeyOne que haya generado un Registrador autorizado. Esta verificación está incluida en la funcionalidad que proporciona la función FUNC_BATCHVER.

6.1.5.3.14 FDP_UCT.1.1 (iteración 2)

Este requisito se refiere a la comunicación de datos de usuario a través de un canal externo (comunicación entre el TOE y un producto IT confiable o un usuario).

Las comunicaciones a las que afectad este requisito son las siguientes:

- Comunicaciones entre el TOE y el servidor de base de datos. Puesto que el cliente de la base de datos y los servidores KeyOne están en la misma máquina *host*, las comunicaciones entre el TOE y el cliente Oracle se realizan bajo la protección física que se aplica sobre dicha máquina. Respecto a la comunicación entre el cliente y el servidor Oracle, se puede proteger por medio del protocolo SSL que se establece entre el cliente y el servidor Oracle.
- Las comunicaciones entre el TOE y el SCD (dispositivo de creación de firmas), y entre el TOE y el HSM (módulo de seguridad *hardware*) se protegen mediante la protección física que se aplica a la máquina *host*.

6.1.5.3.15 FPT_ITT.1.1 (iteración 3)

Este requisito requiere la protección (integridad) de los datos TSF, cuando éstos se transmiten entre partes separadas del TOE.

El requisito FPT_ITT.1.1 (iteración 3) necesita la aplicación del servicio de integridad sobre los datos TSF. Los datos TSF se pueden incluir en la comunicación entre los componentes KeyOne LRA y KeyOne CA, en la comunicación entre KeyOne RA y KeyOne CA y en la comunicación entre KeyOne CA y KeyOne VA.

Para proteger los datos TSF en la comunicación entre KeyOne RA/KeyOne LRA y KeyOne CA, después de añadir todos los datos al lote, KeyOne RA/KeyOne LRA lo firma para asegurar que la CA recibe el lote sin que sea modificado por un tercero. El campo *rasignature* del lote contiene la firma del lote que genera la Autoridad de Registro, sin anexar el lote. La generación de la firma digital del lote la proporciona la funcionalidad de la función FUNC_BATCHSIG y el campo *casignature* contiene la firma del lote que genera la CA, sin anexar el lote.



Respecto a la comunicación entre KeyOne CA y KeyOne VA, cuando KeyOne CA envía información de revocación a KeyOne VA, para proteger esta información, después de añadir todos los datos al mensaje NDCCP, KeyOne CA lo firma para asegurar que KeyOne VA recibe el mensaje sin que lo modifique un tercero. El campo *signature* del mensaje NDCCP contiene la firma del mensaje (sin anexas los datos del mensaje) generada por la CA. La generación de la firma digital del mensaje es provista por funcionalidad que proporciona la función FUNC_NDCCPSIGCA.

Las comunicaciones entre KeyOne RA/KeyOne LRA y KeyOne CA (lote KeyOne utilizado como formato de datos) y entre KeyOne CA y KeyOne VA (mensaje NDCCP utilizado como formato de datos) utilizan el protocolo SSL/TLS (con autenticación de cliente) con la finalidad de proporcionar el servicio de integridad a estas comunicaciones. Esta funcionalidad es provista por la función FUNC_K1SSLTLS.

6.1.5.3.16 FPT_ITT.1.1 (iteration 4)

Este requisito requiere la protección (confidencialidad) de los datos TSF, cuando éstos se transmiten entre partes separadas del TOE.

El requisito FPT_ITT.1.1 (iteración 4) necesita la aplicación del servicio de confidencialidad sobre los datos TSF. Los datos TSF se pueden incluir en la comunicación entre los componentes KeyOne RA/KeyOne LRA y KeyOne CA, en la comunicación entre KeyOne RA y KeyOne CA y en la comunicación entre KeyOne CA y KeyOne VA.

Las comunicaciones entre KeyOne RA/KeyOne LRA y KeyOne CA (lote KeyOne utilizado como formato de datos) y entre KeyOne CA y KeyOne VA (mensaje NDCCP utilizado como formato de datos) utilizan el protocolo SSL/TLS (con autenticación de cliente) con la finalidad de proporcionar el servicio de confidencialidad a estas comunicaciones. Esta funcionalidad es provista por la función FUNC_K1SSLTLS.

6.1.5.3.17 FDP_CIMC_BKP.2.1

El requisito FDP_CIMC_BKP.2.1 requiere que los datos de *backup* (generados por la función de seguridad FUNC_BACKUP) sean protegidos frente a la modificación mediante el uso de firmas digitales, *hashes* con clave o códigos de autenticación.

El sistema KeyOne incluye una funcionalidad de *backup* que se encarga de realizar la copia de seguridad completa del sistema KeyOne que sea necesaria para reconstruir su estado actual y la copia de la versión del *software* utilizado para instalar inicialmente el sistema KeyOne. Los datos del *backup* del sistema que son necesarios para reproducir el estado del sistema en el instante en que dicho *backup* se realiza incluye toda la información necesaria que esté guardada en el disco duro de la máquina en la que se haya instalado el sistema KeyOne. Esta información se protege mediante la función de seguridad FUNC_OBFUSCATION, la cual protege a estos datos de su modificación no autorizada.

6.1.5.3.18 FDP_CIMC_BKP.2.2

El requisito FDP_CIMC_BKP.2.2 necesita que los parámetros críticos y demás informaciones confidenciales de los datos que contiene la copia de seguridad (generados por la función de seguridad FUNC_BACKUP) sean almacenados sólo bajo la forma cifrada.

El sistema KeyOne incluye una funcionalidad de *backup* que se encarga de realizar la copia de seguridad completa del sistema KeyOne que sea necesaria para reconstruir su estado actual y la copia de la versión del *software* utilizado para instalar inicialmente el sistema KeyOne. Los datos del *backup* del sistema que son necesarios para reproducir el estado del sistema en el instante en que dicho *backup* se realiza incluye toda la información necesaria que esté guardada en el disco duro de la máquina en la que se haya instalado el sistema KeyOne. Esta información se protege mediante la función de seguridad FUNC_OBFUSCATION, la cual protege a estos datos de que sean revelados de forma no autorizada.

6.1.6 Gestión de certificados

Una parte importante del sistema KeyOne es la gestión de todos los aspectos de la gestión de la certificación: verificación de las peticiones de certificación, generación de certificados y CRLs y gestión de perfiles de certificación. Esta sección contiene información sobre los requisitos y las funciones que se relacionan con estos aspectos de la seguridad.

6.1.6.1 Requisitos funcionales que satisfacen las funciones de seguridad

Los servicios de gestión de la certificación se componen de las siguientes funciones de seguridad:

- Función de verificación de peticiones de certificación (FUNC_CERTREQVER). Esta función se encarga de verificar las peticiones de certificación que recibe KeyOne CA. Esta validación incluye la verificación de la prueba de posesión que incluye la petición y que es generada por la Autoridad de Registro.
- Función de generación de certificados (FUNC_CERTSGENE). Esta funcionalidad se encarga de generar certificados que cumplen con el estándar X.509. la función asegura que los certificados que se generan son consistentes con el perfil de certificación actualmente definido.
- Función de generación de PKCS #12 (FUNC_PKCS12GENE). Esta funcionalidad se encarga de generar PKCS #12 según las especificaciones de "PKCS #12 Personal Information Exchange Syntax".
- Función de generación de CRLs (FUNC_CRLSGENE). Esta funcionalidad se encarga de generar CRLs de acuerdo con la recomendación ITU-T X.509. La función asegura que los CRLs son consistentes con el perfil actualmente definido.
- Perfil de certificación (FUNC_CERTPROF). Esta funcionalidad se encarga de proporcionar a KeyOne CA las funciones para que un administrador gestione los perfiles de certificación.
- Perfil de revocación (FUNC_REVPROF). Esta funcionalidad se encarga de proporcionar a KeyOne CA las funciones para que un administrador gestione los perfiles de revocación.
- Perfil OCSP (FUNC_OCSPPROF). Esta funcionalidad se encarga de proporcionar a KeyOne VA las funciones para que un administrador gestione los perfiles de OCSP.



Estos servicios satisfacen los siguientes requisitos:

6.1.6.1.1 FDP_CIMC_CER.1.3

FDP_CIMC_CER.1.3 requiere que el sistema verifique, antes de emitir un certificado que su futuro titular posee la clave privada que corresponde a la clave pública que consta en la petición de certificación, salvo que el par de claves sea generado por el TSF, y siempre que la clave privada permita realizar firmas digitales.

En el caso de que la clave privada se pueda utilizar para realizar firmas digitales, la verificación de que el titular posee la clave privada que corresponde a la clave pública que consta en la petición se lleva a cabo mediante la verificación de la firma que haya generado la entidad peticionaria que se incluye en la petición de certificado PKCS #10 o X.509. La función FUNC_CERTREQVER verifica la petición de certificación autofirmada que se incluye en un lote de certificación KeyOne.

6.1.6.1.2 FDP_CIMC_CER.1.1

La función de generación de certificados (FUNC_CERTSGENE) se encarga de generar, en el componente KeyOne CA, certificados que son conformes al estándar X.509. Por consiguiente se satisface el requisito FDP_CIMC_CER.1.1.

6.1.6.1.3 FDP_CIMC_CER.1.2

Antes de generar un certificado X.509, la función FUNC_CERTSGENE asegura que el certificado que se van a generar sea consistentes con el perfil de certificación definido actualmente. Por consiguiente, la función FUNC_CERTSGENE cubre el cumplimiento del requisito FDP_CIMC_CER.1.2 .

6.1.6.1.4 FDP_SDI_CIMC.3.1

El requisito FDP_SDI_CIMC.3.1 obliga a proporcionar el servicio de integridad (por medio de firmas digitales, *hashes* con clave o códigos de autenticación) a las claves públicas que se guardan en un módulo criptográfico validado por CIMC pero no por FIPS 140-1. En el caso de que la clave pública haya sido certificada, la integridad la proporciona la firma digital del certificado. Puesto que el estándar X.509 incluye en su formato un campo para la firma digital, la función FUNC_CERTSGENE cubre el cumplimiento del requisito FDP_SDI_CIMC.3.1.

6.1.6.1.5 FMT_MOF_CIMC.3.1

Antes de generar un certificado X.509, la función FUNC_CERTSGENE asegura que el certificado que se van a generar sea consistentes con el perfil de certificación definido actualmente. Por consiguiente, la función FUNC_CERTSGENE cubre el cumplimiento del requisito FMT_MOF_CIMC.3.1.

6.1.6.1.6 FDP_CIMC_CER.1.4

La función FUNC_CERTSGENE comprueba el cumplimiento de las siguientes restricciones en la generación de certificados X.509:

- El campo `version` debe contener el valor entero 0, 1 o 2.

- Si el certificado contiene `issuerUniqueID` o `subjectUniqueID` entonces el campo `version` debe contener el valor entero 1 o 2.
- Si el certificado contiene `extensions`, el campo `version` debe contener el valor entero 2.
- El `serialNumber` debe ser único, por lo que respecta a la Autoridad de Certificación.
- El campo `validity` debe especificar un valor `notBefore` que no sea anterior al momento actual y un valor `notAfter` que no sea anterior al valor que especifica `notBefore`.
- Si el campo `issuer` contiene un nombre nulo, el certificado debe contener la extensión crítica `issuerAltName`.
- Si el campo `subject` contiene un nombre nulo, el certificado debe contener la extensión crítica `subjectAltName`.
- El campo `signature` y el algoritmo en el campo `subjectPublicKeyInfo` deben contener el OID de un algoritmo aprobado o recomendado por FIPS.

Puesto que la función `FUNC_CERTSGENE` comprueba el cumplimiento de estas restricciones, esta función cubre el cumplimiento del requisito `FDP_CIMC_CER.1.4`.

6.1.6.1.7 FDP_ETC_CIMC.5.1

La función de generación de PKCS# 12 (`FUNC_PKCS12GENE`) se encarga de generar en el componente KeyOne CA estructuras PKCS #12 que siguen las especificaciones de "PKCS #12 Personal Information Exchange Syntax".

El requisito `FDP_ETC_CIMC.5.1` fuerza que la exportación de las claves privadas u secretas del TOE se realice bajo encriptación o utilizando un procedimientos de conocimiento compartido. En el caso de la distribución electrónica de claves secretas y privadas, su exportación se realizará sólo bajo encriptación.

Respecto a la exportación de claves privadas en formato PKCS #12, puesto que la función `FUNC_PKCS12GENE` sigue las especificaciones PKCS #12, se cumple dicho requisito (la especificación PKCS #12 se basa en un modo de privacidad que utiliza el cifrado para proteger la revelación de información de carácter personal).

6.1.6.1.8 FDP_CIMC_CRL.1.1

La función de generación de CRLs (`FUNC_CRLSGENE`) se encarga de generar, en el componente KeyOne CA, CRLs que son conformes a la recomendación ITU-T X.509. Esta función comprueba que se cumplan las siguientes restricciones en la generación de CRLs:

- Si el campo `version` está presente, debe contener un 1.
- Si la CRL contiene cualquier extensión crítica, el campo `versión` debe contener el valor entero 1.
- Si el campo `issuer` contiene un nombre nulo, la CRL debe contener la extensión crítica `issuerAltName`.



- El campo `signature` y el algoritmo en el campo `subjectPublicKeyInfo` deben contener el OID de un algoritmo aprobado o recomendado por FIPS.
- El campo `thisUpdate` debe indicar la fecha de emisión de la CRL.
- El instante indicado en el campo `nextUpdate` no debe ser anterior al instante del campo `thisUpdate`.

Por lo tanto, se satisface el requisito FDP_CIMC_CRL.1.1.

6.1.6.1.9 FMT_MOF_CIMC.5.1

La función de generación de CRLs (FUNC_CRLSGENE) se encarga de generar, en el componente KeyOne CA, CRLs que son conformes a la recomendación ITU-T X.509.

Antes de generar una CRL X.509, la función FUNC_CRLSGENE asegura que la CRL emitida sea consistente con el perfil de la lista de revocación de certificados. Por consiguiente la función FUNC_CRLSGENE cubre el cumplimiento del requisito FMT_MOF_CIMC.5.1.

6.1.6.1.10 FMT_MOF_CIMC.3.1

Este requisito implica la funcionalidad que permite gestionar un perfil de certificado a un administrador autorizado. A través de la función perfil de certificación (FUNC_CERTPROF) KeyOne CA proporciona un mecanismo de plantillas de certificación.

Las plantillas de certificación determinan las características de los certificados que emite la CA (como las extensiones del certificado). Una plantilla de certificación, también llamada política de certificación o simplemente política, es un conjunto de normas programables que definen restricciones sobre los tipos de peticiones de certificado que la CA acepta, así como las características de los certificados emitidos de ese tipo de petición (por ejemplo, extensiones de certificado).

Se pueden definir múltiples perfiles de certificación en una CA, uno por cada tipo de petición de certificación que se vaya a procesar. Las peticiones se pueden clasificar en diferentes categorías según el uso que se pretenda dar a los certificados, el tipo de entidad para quién se vaya a emitir o cualquier otro criterio. El conjunto de plantillas de certificación no tiene porqué ser exhaustivo, esto es, no es necesario definir una plantilla de certificación para cada tipo posible de petición que emitirá. Se pueden definir múltiples plantillas de certificación para la CA, una para cada tipo de petición de certificado que se procesará. Además, varias plantillas se pueden definir para el mismo tipo de petición de modo que tengan pequeñas diferencias como el período de validez del certificado. Los administradores autorizados de la CA deben asignar un nombre a cada perfil de certificación que definan.

Alternativamente, cuando un certificado o un PKCS #12 es emitido directamente desde la aplicación de administración de KeyOne CA, el administrador de la CA debe seleccionar explícitamente la plantilla de certificación que se debe aplicar.

Aplicación de la Plantilla de Certificación

Para emitir un certificado, KeyOne CA aplica una plantilla de certificación a una petición de certificado. Esto se conoce como aplicación de la plantilla de certificación y es el primer paso del proceso de emisión de un certificado. Este paso

consiste en la aplicación de las reglas que define la plantilla de certificación a los diferentes campos y extensiones, teniendo en cuenta en algunos caso el valor que se propone en la petición de certificado. Esto puede resultar en campos y extensiones que se añaden, que se quitan o que se modifican.

Después de la aplicación de la plantilla de certificación, se requiere un segundo paso para completar el proceso de emisión de un certificado. Este paso se llama generación del certificado/PKCS #12 y consiste en la firma de un certificado y posiblemente en la generación de un PKCS #12, en el caso de que la petición no incluya la clave pública (el PKCS #12 se utilizará para entregar a su titular el certificado y la clave privada asociada).

El paso de generación del certificado/PKCS #12 no se realizará si la plantilla de certificación no se puede aplicar a la petición de certificado.

Certificación vs. plantillas de certificado

As the result of applying a certification template to a certificate request, a certificate template is obtained. A certificate template contains the same information that a certificate but it does not include a signature (a certificate that is not yet signed). In fact, the original certificate request may also be represented as a certificate template. The internal representation of a certificate template is the `CertTemplate` ASN.1 structure defined in IETF RFC 2511.

Besides fields and extensions that will be part of the final certificate, the certificate template may also include other information that will not be directly included in the certificate or not even used to build the certificate itself. In particular, when the original certificate request does not include a public key, the certificate template resulting from the certification template application will include information on:

- How the key pair is to be generated by the CA engine in the later certificate generation phase (this includes information on the key algorithm and related parameters according to the particular algorithm, e.g. the RSA key size).
- The password of the PKCS #12 to generate along with the certificate. Usually this information is also included in the original certificate request.

Reglas de Plantilla de Certificación

La definición de una plantilla de certificación consta de varios campos, cada uno de los cuales especifica cómo establecer un determinado campo o extensión de los certificados que se emitirán con esa plantilla. Ejemplos de campos de certificado son la versión y el periodo de validez del certificado. Ejemplos de extensiones incluyen los nombres alternativos del titular y las restricciones básicas que define el estándar X.509 (en adelante el término campo será utilizado para referirse tanto a los campos como a las extensiones de certificado).

Para algunos campos, la petición de certificado puede proponer el valor que se debe poner en el certificado que se emita. En estos casos, el perfil de certificación puede imponer restricciones en los valores que son admisibles. A estos campos se les llama campos negociables.

Para cada campo de la plantilla de certificación se pueden definir un conjunto de reglas de aplicación. Estas reglas son de los siguientes tipos:



- Ausencia o presencia del campo
Determina si el campo se incluirá o no en los certificados que se emitan.
Estas reglas permiten controlar qué extensiones serán incluidas.
- Opcionalidad del campo
Para algunos campos negociables es posible especificar que sólo se incluyan en el certificado en el caso de que la petición de certificado proporcione un valor para ellos.
- Valor del campo
Estas reglas determinan el valor que se deberá poner en un campo de certificado que se emita.
- Restricciones del valor del campo
Para algunos campos negociables es posible especificar restricciones o rangos de valores permitidos en la petición que dichos campos o extensiones deben respetar para que se puedan incluir en el certificado que se emita. Esto permite limitar los valores que se pueden solicitar.
- Valor por defecto del campo
Cuando se especifica que un campo negociable siempre debe estar presente, esta regla determina el valor que debe asignarse al campo en el caso de que no se haya incluido en la petición de certificado.
- Criticidad de la Extensión
Estas reglas determinan si una extensión específica debe marcarse o no como crítica cuando se incluya en un certificado que se emita.

Cualquier extensión contenida en la petición de certificado que no corresponda a alguno de los campos de la plantilla de certificación será ignorada y no se incluirá en el certificado que se emita.

Las plantillas se pueden añadir importar, examinar, modificar y borrar en cualquier momento. KeyOne CA requiere que sea definida al menos una plantilla de certificación antes de poder comenzar a emitir certificados.

6.1.6.1.11 FMT_MOF_CIMC.3.2

Estos requisitos implican la funcionalidad que permite gestionar un perfil de certificación a un administrador autorizado. KeyOne CA proporciona un mecanismo de gestión de plantillas, mediante la función perfil de certificación (FUNC_CERTPROF).

Vea FMT_MOF_CIMC.3.1, página 160, para obtener más información sobre las plantillas de certificación o sobre la función FUNC_CERTPROF.

6.1.6.1.12 FMT_MOF_CIMC.3.3

Estos requisitos implican la funcionalidad que permite gestionar un perfil de certificación a un administrador autorizado. KeyOne CA proporciona un mecanismo

de plantillas de certificación, mediante la función perfil de certificación (FUNC_CERTPROF).

Vea FMT_MOF_CIMC.3.1, página 160, para obtener más información sobre las plantillas de certificación.

6.1.6.1.13 FMT_MOF_CIMC.3.4

Estos requisitos implican la funcionalidad que permite gestionar un perfil de certificación a un administrador autorizado. KeyOne CA proporciona un mecanismo de plantillas de certificación, mediante la función perfil de certificación (FUNC_CERTPROF).

Vea FMT_MOF_CIMC.3.1, página 160, para obtener más información sobre las plantillas de certificación o sobre la función FUNC_CERTPROF.

6.1.6.1.14 FMT_MOF_CIMC.5.1

Este requerimiento implica la funcionalidad que permite gestionar un perfil de revocación a un administrador autorizado. Mediante la función perfil de certificación (FUNC_REVPROF) KeyOne CA proporciona un mecanismo de plantillas de revocación.

Mientras que una plantilla de certificación define los campos y extensiones para un tipo de certificados que se emiten, una plantilla de CRL determina cómo KeyOne CA debe fijar los campos y las extensiones de una lista de certificados revocados (CRL) particular que emita la CA. Ejemplos de campos de CRL son la versión de la CRL y la fecha de próxima actualización de la CRL. Ejemplos de extensiones de la CRL son el número de la CRL y la extensión *issuing distribution point* que define el estándar X.509.

A diferencia de una plantilla de certificación, una plantilla de CRL no se aplica a ninguna petición. En su lugar la plantilla de la CRL se utiliza directamente para generar una CRL junto con la información que exista en la base de datos de la CA sobre los certificados actualmente revocados. Debido a esta diferencia, no se habla de reglas de aplicación de la plantilla de CRL, sino simplemente de campos de plantilla de CRL.

Conjunto de plantillas de CRL

La CA puede emitir una sola CRL o varias. En el segundo caso, habitualmente cada CRL cubrirá un conjunto diferente de razones de revocación o de tipos de entidad y a cada CRL se le asignarán diferentes puntos de distribución (métodos para obtener información de la CRL). Esta información también está contenida en las plantillas de CRL correspondientes.

Se debe definir una plantilla de CRL para cada CRL que la CA vaya a emitir. Al menos debe definirse una plantilla de CRL. A diferencia de las plantillas de certificación, puesto que el número de plantillas de CRL determina el número de CRLs, no se probable que este parámetro varíe durante la vida de la CA. Además las plantillas de CRL deben estar completamente definidas antes de comenzar a emitir certificados, de modo que la información sobre los puntos de distribución de CRLs se pueda incluir adecuadamente en los certificados.



Generación de CRLs utilizando plantillas de CRL

Cuando se tienen que emitir las CRLs, KeyOne CA utiliza para generarlas el conjunto de plantillas de CRL que se hayan definido. Este proceso puede involucrar a todas las plantillas o sólo a algunas de ellas, dependiendo de qué CRLs deban actualizarse. Por ejemplo, la primera vez que se generan las CRLs se utilizan todas las plantillas. Por el contrario, cuando se actualizan las CRLs, en realidad sólo se generan aquellas CRLs que hayan expirado o cuyo contenido deba cambiar, por lo que sólo se utilizarán las plantillas de CRL correspondientes.

Cuando se tiene que emitir una CRL que corresponde a una determinada plantilla de CRL (bien sea porque se emite por primera vez o porque necesita ser actualizada), dicha plantilla se utiliza para determinar lo siguiente:

- El valor de algunos campos de la CRL (por ejemplo, la versión y la fecha de la próxima actualización de la CRL). Los campos que la plantilla no especifica son establecidos automáticamente por KeyOne CA.
- Las extensiones que la CRL debe contener y si son críticas o no. Para algunas extensiones, el valor de la extensión puede estar definido en la plantilla de CRL (e.g. la extensión *Issuing Distribution Point*). En otros casos, el valor de la extensión es calculado automáticamente por KeyOne CA (e.g. la extensión *CRL number*).
- Qué certificados revocados deben incluirse en la CRL. Esto sólo es aplicable si la plantilla de CRL define las razones de revocación que deben cubrir las CRL. En este caso KeyOne CA incluirá cada certificado revocado en la(s) CRL(s) apropiadas según cual sea la razón de su revocación (esta información y otros datos del certificado se obtienen de la base de datos de la CA).

Además, como se ha dicho anteriormente, el conjunto de CRLs que se haya definido no se utiliza solamente cuando se emiten CRLs, sino también cuando se emiten certificados. Concretamente, las plantillas de CRLs se utilizan para determinar si la extensión *CRL distribution points* debe incluirse en cada certificado emitido y cuál debe ser su valor.

6.1.6.1.15 FMT_MOF_CIMC.5.2

Este requerimiento implica la funcionalidad que permite gestionar un perfil de revocación a un administrador autorizado. Mediante la función perfil de certificación (FUNC_REVPROF) KeyOne CA proporciona un mecanismo de plantillas de revocación.

Vea FMT_MOF_CIMC.5.1, página 163, para obtener más información sobre las plantillas de revocación o sobre la función FUNC_REVPROF.

6.1.6.1.16 FMT_MOF_CIMC.5.3

Este requerimiento implica la funcionalidad que permite gestionar un perfil de revocación a un administrador autorizado. Mediante la función perfil de certificación (FUNC_REVPROF) KeyOne CA proporciona un mecanismo de plantillas de revocación.

6.1.6.1.17 FMT_MOF_CIMC.6.1

Este requerimiento implica la funcionalidad que permite gestionar un perfil de respuestas OCSP a un administrador autorizado. Mediante la función perfil de OCSP (FUNC_OCSPROF) KeyOne VA proporciona la funcionalidad para configurar algunos de los campos de las respuestas OCSP.

KeyOne VA permite generar respuestas OCSP básicas y un administrador autorizado puede configurar ciertos campos de este tipo de respuestas.

6.1.6.1.18 FMT_MOF_CIMC.6.2

Este requerimiento implica la funcionalidad que permite gestionar un perfil de respuestas OCSP a un administrador autorizado. Mediante la función perfil de OCSP (FUNC_OCSPROF) KeyOne VA proporciona la funcionalidad para configurar algunos de los campos de las respuestas OCSP.

KeyOne VA permite generar respuestas OCSP básicas y un administrador autorizado puede configurar ciertos campos de este tipo de respuestas.

6.1.6.1.19 FMT_MOF_CIMC.6.3

Este requerimiento implica la funcionalidad que permite gestionar un perfil de respuestas OCSP a un administrador autorizado. Mediante la función perfil de OCSP (FUNC_OCSPROF) KeyOne VA proporciona la funcionalidad para configurar algunos de los campos de las respuestas OCSP.

KeyOne VA permite generar respuestas OCSP básicas y un administrador autorizado puede configurar ciertos campos de este tipo de respuestas.

6.1.6.1.20 FDP_ACF_CIMC.2.1

Las claves privadas de personal CIMC se almacenan en un FIPS 140-2 nivel 2

6.1.6.1.21 FDP_ACF_CIMC.2.2

Las únicas claves privadas de titulares de certificados que se almacenan en el TOE son las claves que se hayan copiado en el componente KeyOne Archive (con la finalidad de posibilitar la recuperación de claves).

Estas claves son cifradas utilizando claves privadas de larga duración para la protección de claves (claves componente) y este cifrado se realiza utilizando un módulo criptográfico que se haya validado como módulo FIPS 140-2 nivel 2 (administrador del componente KeyArchive).

6.1.6.1.22 FDP_ACF_CIMC.3.1

Las únicas claves de usuario que se guardan en un módulo criptográfico validado como CIMC, pero no como FIPS, son las claves privadas de sujeto

Estas claves son las que han sido copiadas por el componente KeyOne Archive (con la finalidad de posibilitar la recuperación de estas claves). Estas claves se cifran utilizando un módulo criptográfico (administrador del componente KeyArchive) que se haya validado como módulo FIPS 140-2 nivel 2.



6.1.6.1.23 FMT_MTD_CIMC.4.1

Las claves privadas CIMC se guardan en un módulo criptográfico que se haya validado como módulo FIPS 140-2 nivel 3. Cuando se exportan estas claves del TOE (apagado del sistema), las claves se cifran con este módulo de seguridad *hardware* FIPS 140-2 nivel 3.

6.1.6.1.24 FMT_MTD_CIMC.5.1

FMT_MTD_CIMC.5.1 requiere que las claves secretas TSF que se almacenan dentro del TOE, aunque no en un módulo criptográfico validado como FIPS 140-1, se almacenen de forma cifrada. Este cifrado debe ser realizado por un módulo criptográfico que haya sido validado como un módulo FIPS 140-1.

Todas las claves secretas o bien se almacenan en módulos criptográficos validados como módulos FIPS 140-2 nivel 3, o bien son cifradas utilizando un módulo de *hardware* de seguridad FIPS 140-2 nivel 3.

6.1.6.1.25 FMT_MTD_CIMC.7.1

Las claves privadas y secretas no se exportan del TOE. Las claves privadas y secretas se mantienen en el almacén seguro en el que fueron generadas y nunca pueden ser exportadas desde allí.

6.1.6.1.26 FCS_CKM_CIMC.5.1

El TOE no mantiene en claro ni las claves secretas ni las privadas.

6.1.7 Almacén privado seguro

El almacén privado seguro es un objeto seguro en el que se almacenan datos de configuración sensibles para protegerlos frente al acceso y la modificación ilícitas. Ejemplos de información que se pueden almacenar en este almacén protegido son: claves privadas, certificados raíz, datos de configuración, etc.

Este almacén privado seguro de KeyOne, típicamente almacena los siguientes datos de las aplicaciones KeyOne:

- Datos de configuración del sistema
- Datos de configuración de las aplicaciones
- Claves de servicio de las aplicaciones. Como claves de servicio (aplicaciones) se consideran el conjunto de claves asimétricas que una aplicación KeyOne 3.0 necesita para operar correctamente: claves de infraestructura (SSL, firma de lotes KeyOne, ...), claves para firmar certificados y CRLs, claves para firmar mensajes OCSP, ...
- Claves de ofuscación de datos. La ofuscación de datos es un término genérico que abarca todas las claves auxiliares (simétricas y asimétricas) que están involucradas en los mecanismos de protección de datos de KeyOne 3.0 (claves *master*, claves de protección de claves, claves de protección de datos, ...).

En el almacén privado seguro los datos que se almacenan se organizan en forma de árbol, identificándose cada entrada por su tipo, su nombre y la entrada predecesora (entrada padre). Para cada entrada, se puede definir un conjunto de atributos, cada uno de los cuales tiene un nombre y un valor. Es posible definir atributos que sean referencias a otras entradas y atributos que sean referencias externas (referencias a entradas en otros almacenes privados seguros).

La implementación del registro utiliza dos tablas i3D, pero los siguientes datos se almacenan en disco (datos necesarios para acceder al almacén privado seguro):

- La configuración de la clave de ofuscación de datos.
- Las claves de ofuscación de datos.
- La clave que protege la configuración de la base de datos del sistema.
- La configuración de la base de datos del sistema.
- Las claves de servicio de las aplicaciones.
- Los certificados del sistema (certificados de aplicación y certificados que se necesitan para operar).

6.1.7.1 Funcionalidad del almacén privado seguro

La principal funcionalidad del almacén privado seguro es proveer un almacenamiento seguro a los certificados, las claves y los datos de configuración. El almacén privado seguro también mantiene un registro histórico de los certificados expirados. Todos los datos almacenados en el almacén privado seguro se protegen con técnicas criptográficas frente al acceso y la modificación no autorizados.

El almacén privado seguro puede contener cualquier tipo de objeto de objetos firmados, como por ejemplo certificados. La firma de cualquier objeto firmado es validada antes de ser insertada en el almacén privado seguro. Si la firma no se valida como correcta, el objeto no se inserta. Siguiendo esta regla, es fácil de ver que el almacén privado seguro almacena jerarquías de certificación completas.

Los objetos firmados (como los certificados) no son válidos siempre. Cada objeto tiene una fecha de expiración. Cuando se alcanza la fecha de expiración, el objeto ya no se puede utilizar más y debe borrarse del almacén privado seguro. El mecanismo para borrar objetos expirados es el siguiente:

- Cada vez que se accede a un objeto, el almacén privado seguro recorre varios objetos hasta que localiza el objeto requerido. Cualquier objeto inválido que se detecte durante este recorrido es eliminado.
- No todos los objetos del almacén privado seguro son validados cuando se busca un objeto determinado. Pueden quedar remanentes objetos inválidos en el almacén privado seguro, pero estos objetos serán borrados tan pronto como alguien intente acceder a ellos.
- Los certificados caducados cuya clave privada esté almacenada en el almacén privado seguro no se borran, sino que se pasan al registro histórico del almacén privado seguro.



Para proteger el almacén privado seguro frente a modificaciones no autorizadas (servicio de integridad) se utilizan los dos mecanismos siguientes:

- Mecanismo de integridad que utiliza la base de datos i3D (se aplica a los datos incluidos en la tabla i3D).
- Mecanismo de integridad utilizado en el almacén privado seguro. Este mecanismo consiste en la aplicación de un algoritmo de *hash* a los datos que deben protegerse. El resultado de la aplicación del algoritmo de *hash* se cifra junto a otros datos utilizando un algoritmo simétrico. Tanto el tipo del algoritmo de *hash* como el tipo de algoritmo de cifrado simétrico son configurables..

El almacén privado seguro también se protege contra la revelación no autorizada (servicio de confidencialidad) mediante el cifrado de los datos utilizando el mecanismo de ofuscación (el tipo de algoritmo es configurable).

6.1.7.1.1 Registro histórico

El registro histórico del almacén privado seguro almacena los certificados caducados y sus correspondientes claves privadas. Cuando un certificado expira y el almacén privado seguro lo advierte, se comprueba si también se guarda en el almacén la clave privada asociada. Si la clave privada está presente, el certificado y su clave privada se pasan al registro histórico. En cualquier otro caso el certificado es borrado.

Los certificados caducados y sus correspondientes claves privadas se guardan en el registro para descifrar datos cifrados y/o para validar las firmas que se realizaron cuando el certificado aún era válido. Si el certificado es borrado y no se guarda en el histórico, los datos permanecerán cifrados para siempre porque fallará la validación de las firmas.

6.1.7.2 Requisitos funcionales satisfechos por las funciones de seguridad

Los servicios de gestión del almacén privado seguro constan de las siguientes funciones de seguridad:

- Función de acceso al almacén privado seguro (FUNC_PSSINSERT). Esta funcionalidad se encarga de verificar los objetos firmados cuando se insertan en el almacén privado seguro. La función borrará los objetos inválidos que encuentre (los certificados expirados cuya clave privada se guarda en el almacén privado seguro no se borran, sino que se pasan al registro histórico).

Este servicio cumple los siguientes requisitos:

6.1.7.2.1 FDP_SDI_CIMC.3.2

FDP_SDI_CIMC.3.2 requiere que la firma digital, *hash* con clave o código de autenticación que se utilice para proteger una clave pública sea verificada en cada acceso que se haga a la clave.

Las claves públicas se protegen utilizando la firma digital de su certificado correspondiente. Estos certificados se almacenan en el almacén privado seguro (PSS)

y la integridad de las claves públicas que se guardan en el PSS se asegura de la siguiente manera:

- la función FUNC_PSSINSERT garantiza que cada vez que se inserta en el almacén privado seguro un nuevo objeto firmado (certificado, CRL), éste es validado (la firma digital relacionada con el certificado o la CRL es también validada).
- El contenido del almacén privado seguro se mantiene en la base de datos i3D del registro y, por lo tanto, la integridad de los datos guardados en la base de datos está asegurada por las funciones FUNC_I3DSESSION y FUNC_I3DHISTORIC.
- En los accesos a cualquier objeto que se guarda en el PSS se comprueba la fecha de expiración. Los objetos caducados se borran del almacén privado seguro y su utilización es prohibida.
- Cuando cualquier aplicación arranca, el contenido del almacén privado seguro se lee de la base de datos i3D y se copia en la memoria de la máquina. En este paso, se verifica la integridad del PSS por lo que queda garantizada la integridad de las claves públicas.

6.1.8 Gestión del archivo de claves

El TOE incluye el componente KeyOne KeyArchive para almacenar de forma segura las claves que genera KeyOne CA. Estas claves se almacenan en una base de datos segura y en formato PKCS #12 (este PKCS #12 se cifra con una clave simétrica mantenida en el dispositivo PKCS #11). KeyOne Archive incorpora el rol adicional de *Key Recovery Officer*. El componente Key Archive se puede configurar para que el proceso de recuperación requiera el concurso de N *Key Recovery Officers* ($N \geq 1$).

Pueden haber N *officers* por razones de seguridad, de forma que la misma persona no pueda acceder a las claves archivadas. La conexión entre los *Key Recovery Officers* y el almacén de recuperación de claves se realiza de forma remota mediante una conexión web segura. Los *Key Recovery Officers* tienen acceso remoto a la base de datos en la que se guardan los PKCS #12.

Si alguien pierde su clave privada o no recuerda la contraseña de acceso la PKCS #12, los administradores pueden proporcionarlas, si antes han acreditado su identidad (el *Key Recovery Officer* debe ser añadido en KeyOne Console como un usuario *Key Recovery Officer*). Una vez que el *Key Recovery Officer* ha sido autenticado, se genera el PKCS #12 y se entregará al usuario final. El PIN de este PKCS #12 también se genera y se divide en N trozos (N número de *Key Recovery Officers*). Cada *Key Recovery Officer* obtendrá un trozo del PIN y lo entregará al usuario final para que reconstruya el PIN original del PKCS #12.

6.1.8.1 Requisitos funcionales satisfechos por las funciones de seguridad

Los servicios de gestión de Key Archive constan de las siguientes funciones de seguridad:

- Función de recuperación de claves (FUNC_KEYRECOV). Esta funcionalidad se encarga de recuperar las claves privadas a través de los *Key Recovery Officers*.



Esta función está localizada en el componente KeyOne Key Archive (producto KeyOne CA).

Estos servicios satisfacen los siguientes requisitos:

6.1.8.1.1 FDP_ETC_CIMC.5.1

FDP_ETC_CIMC.5.1 requiere que las claves privadas y secretas sólo puedan ser exportarse del TOE de forma cifrada, o utilizando un procedimiento que involucre conocimiento dividido. En el caso de la distribución electrónica, estas claves sólo se pueden exportar del TOE de forma cifrada.

Respecto a la exportación de claves privadas en un proceso de recuperación de claves, esta exportación cumple con las restricciones impuestas por el requisito FDP_ETC_CIMC.5.1. Puesto que la función de recuperación de claves exporta claves privadas en formato PKCS #12 firmado simétricamente, la clave privada se exporta de forma cifrada.

6.1.9 Copia de seguridad y recuperación

El TOE incluye la funcionalidad que se encarga de recuperar un sistema ante la eventualidad evento de un fallo de sistema o cualquier otro fallo grave. Para ser capaz de recuperarse de fallos y de otros eventos no deseables impredecibles, el sistema KeyOne es capaz de hacer copias de seguridad completas del sistema. La copia de seguridad KeyOne se utilizará para restablecer el sistema KeyOne al estado operacional que tenía en un instante de tiempo anterior.

Esta funcionalidad solamente la puede invocar un usuario con el rol *System Administrator*, y sólo este rol puede configura los parámetros que involucra esta funcionalidad.

Los datos que almacenar en la copia de seguridad del sistema que son necesarios para poder recuperar el estado que tenía el sistema en el instante en que se realizó dicha copia de seguridad son los siguientes:

- Claves criptográficas y otros datos que se guardan en el HSM utilizado.
- Toda la información que contienen las bases de datos de KeyOne.
- Toda la información necesaria que se guarda en el disco duro de la máquina en el que el sistema KeyOne está instalado.

6.1.9.1 Requisitos funcionales satisfechos por las funciones de seguridad

Los servicios de copia de seguridad y recuperación constan de las siguientes funciones de seguridad:

- Función de copia de seguridad y de recuperación (FUNC_BACKUP). Esta funcionalidad involucra tareas relacionadas con la funcionalidad de copia de seguridad y de recuperación localizada en el sistema KeyOne.

Estos servicios satisfacen los siguientes requisitos:

6.1.9.1.1 FDP_CIMC_BKP.1.1

El requisito FDP_CIMC_BKP.1.1 requiere que el TOE proporcione una función de copia de seguridad. La función de seguridad FUNC_BACKUP implementa una herramienta de línea de comandos que está relacionado con el proceso de copia de seguridad. Este proceso de copia de seguridad hace un *backup* completo del estado del sistema, de forma que permita su reconstrucción a partir de la copia, y también una copia de la versión de la distribución y de los parches utilizados para instalar inicialmente el sistema KeyOne.

6.1.9.1.2 FDP_CIMC_BKP.1.2

El requisito FDP_CIMC_BKP.1.2 requiere la posibilidad de invocar la función de copia de seguridad bajo demanda. La función de seguridad FUNC_BACKUP implementa una herramienta de línea de comandos que está relacionado con el proceso de copia de seguridad. Este proceso de copia de seguridad hace un *backup* completo del estado del sistema, de forma que permita su reconstrucción a partir de la copia, y también una copia de la versión de la distribución y de los parches utilizados para instalar inicialmente el sistema KeyOne. Los usuarios con el rol *System Administrator* pueden invocar el proceso de copia de seguridad bajo demanda.

6.1.9.1.3 FDP_CIMC_BKP.1.3

FDP_CIMC_BKP.1.3 se garantiza por medio de la función de seguridad FUNC_BACKUP. La función de seguridad FUNC_BACKUP implementa una herramienta de línea de comandos que está relacionado con el proceso de copia de seguridad. Los datos almacenados en la copia de seguridad del sistema que son necesarios para recuperar el estado que tenía el sistema en el instante de realizar dicha copia son los siguientes:

- Claves criptográficas y otros datos que se guardan en le HSM utilizado.
- Toda la información que contienen las bases de datos de KeyOne.
- Toda la información necesaria que se guarda en el disco duro de la máquina en el que el sistema KeyOne está instalado.

6.1.9.1.4 FDP_CIMC_BKP.1.4

El requisito FDP_CIMC_BKP.1.4 requiere que el TOE provea una función de recuperación que sea capaz de restaurar el estado del sistema a partir de una copia de seguridad. La función de seguridad FUNC_BACKUP implementa una herramienta de línea de comandos que está relacionada con el proceso de copia de seguridad. Este proceso de restauración reconstruye el estado del sistema KeyOne a partir del resultado de un proceso de copia de seguridad y de una copia de la misma versión de la distribución y de los parches que se utilizaron para instalar inicialmente el sistema KeyOne.

6.2 Tabla de correspondencia entre requisitos funcionales y funciones de seguridad

Esta sección incluye una tabla de correspondencias entre los requisitos funcionales de seguridad del TOE que se incluyen en esta Declaración de Seguridad y las funciones de seguridad del TOE que es especifican en el documento [FUNCSPEC].

Adicionalmente se ha incluido, una correspondencia entre las funciones de seguridad del TOE y los requisitos funcionales de seguridad del TOE

<i>Requisito funcional</i>	<i>Función de seguridad</i>
FAU_GEN.1.1 (Iter. 2)	FUNC_SADG
FAU_GEN.1.2 (Iter. 2)	FUNC_SADG
FAU_GEN.2.1 (Iter. 2)	FUNC_SADG
FAU_SEL.1.1 (Iter. 2)	FUNC_SELL
FAU_STG.1.1 (Iter. 2)	(El TSF no tiene ninfunca función que permita borrar registros de la base de datos de auditoría)
FAU_STG.1.2 (Iter. 2)	FUNC_DBIV
FAU_STG.4.1 (Iter. 2)	FUNC_CDBC
FPT_STM.1.1 (Iter. 2)	FUNC_I3DSESSION, FUNC_I3DHISTORIC
FMT_MOF.1.1 (Iter. 2)	FUNC_ACCESSCTRL
FDP_ACC.1.1 (Iter. 2)	FUNC_ACCESSCTRL
FDP_ACF.1.1 (Iter. 2)	FUNC_ACCESSCTRL
FDP_ACF.1.2 (Iter. 2)	FUNC_ACCESSCTRL
FDP_ITT.1.1 (Iter. 3)	FUNC_BATCHSIG, FUNC_NDCCPSIGCA, FUNC_K1SSLTLS
FDP_ITT.1.1 (Iter. 4)	FUNC_K1SSLTLS
FDP_UCT.1.1 (Iter. 2)	Ver Nota sobre el requisito FDP_UCT.1.1 en esta sección.
FPT_RVM.1.1 (Iter. 2)	FUNC_ACCESSCTRL
FPT_ITC.1.1 (Iter. 2)	FUNC_OBFUSCATION
FIA_UAU.1.1 (Iter. 2)	FUNC_UIDAUT
FIA_UAU.1.2 (Iter. 2)	FUNC_UIDAUT
FIA_UID.1.1 (Iter. 2)	FUNC_UIDAUT
FIA_UID.1.2 (Iter. 2)	FUNC_UIDAUT
FIA_USB.1.1 (Iter. 2)	FUNC_UIDAUT

FPT_ITT.1.1 (Iter. 3)	FUNC_BATCHSIG, FUNC_NDCCPSIGCA, FUNC_K1SSLTLS
FPT_ITT.1.1 (Iter. 4)	FUNC_K1SSLTLS
FPT_CIMC_TSP.1.1	FUNC_I3DSESSION
FPT_CIMC_TSP.1.2	FUNC_I3DSESSION
FPT_CIMC_TSP.1.3	FUNC_I3DSESSION, FUNC_I3DHISTORIC
FPT_CIMC_TSP.1.4	FUNC_I3DSESSION
FDP_SDI_CIMC.3.1	FUNC_I3DSESSION, FUNC_I3DHISTORIC, FUNC_BATCHSIG, FUNC_K1SSLTLS, FUNC_CERTSGENE
FDP_SDI_CIMC.3.2	FUNC_PSSINSERT, FUNC_I3DSESSION, FUNC_I3DHISTORIC
FDP_ETC_CIMC.5.1	FUNC_PKCS12GENE, FUNC_KEYRECOV
FDP_CIMC_CSE.1.1	FUNC_OCSPRES
FDP_CIMC_CER.1.1	FUNC_CERTSGENE
FDP_CIMC_CER.1.2	FUNC_CERTSGENE
FDP_CIMC_CER.1.3	FUNC_BATCHVER, FUNC_CERTREQVER
FDP_CIMC_CER.1.4	FUNC_CERTSGENE
FDP_CIMC_CRL.1.1	FUNC_CRLSGENE
FDP_CIMC_OCSP.1.1	FUNC_OCSPRES
FCO_NRO_CIMC.3.1	FUNC_BATCHSIG, FUNC_NDCCPSIGCA
FCO_NRO_CIMC.3.2	FUNC_BATCHSIG, FUNC_NDCCPSIGCA
FCO_NRO_CIMC.3.3	FUNC_BATCHVER, FUNC_NDCCPVER
FCO_NRO_CIMC.4.1	FUNC_BATCHVER
FCO_NRO_CIMC.4.2	FUNC_BATCHVER
FMT_MTD_CIMC.7.1	(Las claves privadas y secretas son exportadas desde el TOE)
FMT_MOF_CIMC.3.1	FUNC_CERTSGENE, FUNC_CERTPROF
FMT_MOF_CIMC.3.2	FUNC_ACCESSCTRL, FUNC_CERTPROF
FMT_MOF_CIMC.3.3	FUNC_ACCESSCTRL, FUNC_CERTPROF
FMT_MOF_CIMC.3.4	FUNC_ACCESSCTRL, FUNC_CERTPROF
FMT_MOF_CIMC.5.1	FUNC_CRLSGENE, FUNC_REVPROF
FMT_MOF_CIMC.5.2	FUNC_ACCESSCTRL, FUNC_REVPROF
FMT_MOF_CIMC.5.3	FUNC_ACCESSCTRL, FUNC_REVPROF
FMT_MOF_CIMC.6.1	FUNC_OCSPRES, FUNC_OCSPPROF

FMT_MOF_CIMC.6.2	FUNC_ACCESSCTRL, FUNC_OCSPPROF
FMT_MOF_CIMC.6.3	FUNC_ACCESSCTRL, FUNC_OCSPPROF
FDP_ACF.1.3 (Iter. 2)	(Se aplica una operación none)
FDP_ACF.1.4 (Iter. 2)	(Se aplica una operación none)
FDP_ACF_CIMC.2.1	(Las claves privadas del personal CIMC se guardan en un módulo de seguridad hardware FIPS 140-2 level 2)
FDP_ACF_CIMC.2.2	Ver Nota sobre el requisito FDP_ACF_CIMC.2.2 en esta sección.
FDP_ACF_CIMC.3.1	Ver Nota sobre el requisito FDP_ACF_CIMC.3.1 en esta sección.
FDP_CIMC_BKP.1.1	FUNC_BACKUP
FDP_CIMC_BKP.1.2	FUNC_BACKUP
FDP_CIMC_BKP.1.3	FUNC_BACKUP
FDP_CIMC_BKP.1.4	FUNC_BACKUP
FDP_CIMC_BKP.2.1	FUNC_I3DSESSION, FUNC_I3DHISTORIC, FUNC_DBIV
FDP_CIMC_BKP.2.2	FUNC_I3DSESSION, FUNC_I3DHISTORIC, FUNC_DBIV
FMT_MTD_CIMC.4.1	Ver Nota sobre el requisito FMT_MTD_CIMC.4.1 en esta sección
FMT_MTD_CIMC.5.1	Ver Nota sobre el requisito FMT_MTD_CIMC.5.1 en esta sección
FCS_CKM_CIMC.5.1	(El TOE no mantiene en claro ni claves secretas ni privadas.)
FDP_CIMC_BKP.1.1	FUNC_BACKUP
FDP_CIMC_BKP.1.2	FUNC_BACKUP
FDP_CIMC_BKP.1.3	FUNC_BACKUP
FDP_CIMC_BKP.1.4	FUNC_BACKUP
FDP_CIMC_BKP.2.1	FUNC_OBFUSCATION, FUNC_I3DSESSION, FUNC_I3DHISTORIC
FDP_CIMC_BKP.2.2	FUNC_OBFUSCATION

Tabla 6-3. Tabla de correspondencias ente requisitos funcionales y funciones de seguridad

Nota sobre el requisito FDP_ACF_CIMC.2.2

Las únicas claves privadas de titulares de certificado que se guardan en el TOE son las claves que se copian en el componente KeyOne KeyArchive (con la finalidad de permitir su recuperación). Estas claves son cifradas utilizando claves privadas de larga duración para la protección de claves (claves componente) y este cifrado se realiza

utilizando un módulo criptográfico que se haya validado como módulo FIPS 140-2 nivel 2 (administrador del componente KeyArchive).

Nota sobre el requisito FDP_ACF_CIMC.3.1

Las únicas claves de usuario que se guardan en un módulo criptográfico validado como CIMC, pero no como FIPS, son las claves privadas de sujeto. Estas claves son las claves que han sido copiadas por el componente KeyOne Archive (con la finalidad de posibilitar la recuperación de estas claves). Estas claves se cifran utilizando un módulo criptográfico (administrador del componente KeyArchive) que se haya validado como módulo FIPS 140-2 nivel 2.

Nota sobre el requisito FMT_MTD_CIMC.4.1

Las claves privadas CIMC se guardan en un módulo criptográfico que se haya validado como módulo FIPS 140-2 nivel 3. Cuando se exportan estas claves del TOE (apagado del sistema), las claves se cifran con este módulo de seguridad *hardware* FIPS 140-2 nivel 3.

Nota sobre el requisito FMT_MTD_CIMC.5.1

FMT_MTD_CIMC.5.1 requiere que las claves secretas TSF que se almacenan dentro del TOE, aunque no en un módulo criptográfico validado como FIPS 140-1, se almacenen de forma cifrada. Este cifrado debe ser realizado por un módulo criptográfico que haya sido validado como un módulo FIPS 140-1.

Todas las claves secretas o bien se almacenan en módulos criptográficos validados como módulos FIPS 140-2 nivel 3, o bien son cifradas utilizando un módulo de *hardware* de seguridad FIPS 140-2 nivel 3.

Nota sobre el requisito FDP_UCT.1.1

La comunicación entre el cliente y el servidor Oracle se protege mediante el protocolo SSL que se establece entre dicho cliente y servidor. Este protocolo está implementado por el sistema gestor de la base de datos.

<i>Requisito funcional</i>	<i>Función de seguridad</i>
FUNC_SADG	FAU_GEN.1.1 (Iter. 2), FAU_GEN.1.2 (Iter. 2), FAU_GEN.2.1 (Iter. 2)
FUNC_SELL	FAU_SEL.1.1 (Iter. 2)
FUNC_DBIV	FAU_STG.1.2 (Iter. 2), FDP_CIMC_BKP.2.1
FUNC_CDBC	FAU_STG.4.1 (Iter. 2)
FUNC_I3DSESSION	FPT_STM.1.1 (Iter. 2), FPT_CIMC_TSP.1.1, FPT_CIMC_TSP.1.2, FPT_CIMC_TSP.1.3, FPT_CIMC_TSP.1.4, FDP_SDI_CIMC.3.1, FDP_SDI_CIMC.3.2, FDP_CIMC_BKP.2.1
FUNC_I3DHISTORIC	FPT_STM.1.1 (Iter. 2), FPT_CIMC_TSP.1.3, FDP_SDI_CIMC.3.1, FDP_SDI_CIMC.3.2, FDP_CIMC_BKP.2.1

FUNC_ACCESSCTRL	FMT_MOF.1.1 (Iter. 2), FDP_ACC.1.1 (Iter. 2), FDP_ACF.1.1 (Iter. 2), FDP_ACF.1.2 (Iter. 2), FPT_RVM.1.1, FMT_MOF_CIMC.3.2, FMT_MOF_CIMC.3.3, FMT_MOF_CIMC.3.4, FMT_MOF_CIMC.5.2, FMT_MOF_CIMC.5.3, FMT_MOF_CIMC.6.2, FMT_MOF_CIMC.6.3
FUNC_BATCHSIG	FDP_ITT.1.1 (Iter. 3), FDP_SDI_CIMC.3.1, FCO_NRO_CIMC.3.1, FCO_NRO_CIMC.3.2, FPT_ITT.1.1 (Iter. 3)
FUNC_NDCCPSIGCA	FDP_ITT.1.1 (Iter. 3), FPT_ITT.1.1 (Iter. 3), FCO_NRO_CIMC.3.1, FCO_NRO_CIMC.3.2
FUNC_K1SSLTLS	FDP_ITT.1.1 (Iter. 3), FDP_ITT.1.1 (Iter. 4), FDP_SDI_CIMC.3.1, FPT_ITT.1.1 (Iter. 3), FPT_ITT.1.1 (Iter. 4)
FUNC_OBFUSCATION	FPT_ITC.1.1 (Iter. 2), FDP_CIMC_BKP.2.2, FDP_CIMC_BKP.2.1
FUNC_UIDAUT	FIA_UAU.1.1 (Iter. 2), FIA_UAU.1.2 (Iter. 2), FIA_UID.1.1 (Iter. 2), FIA_UAU.1.2 (Iter. 2), FIA_USB.1.1 (Iter. 2)
FUNC_CERTSGENE	FDP_SDI_CIMC.3.1, FDP_CIMC_CER.1.1, FDP_CIMC_CER.1.2, FDP_CIMC_CER.1.4, FMT_MOF_CIMC.3.1
FUNC_PSSINSERT	FDP_SDI_CIMC.3.2
FUNC_PKCS12GENE	FDP_ETC_CIMC.5.1
FUNC_KEYRECOV	FDP_ETC_CIMC.5.1
FUNC_OCSPRES	FDP_CIMC_CSE.1.1, FDP_CIMC_OCSP.1.1, FMT_MOF_CIMC.6.1
FUNC_BATCHVER	FDP_CIMC_CER.1.3, FCO_NRO_CIMC.3.3, FCO_NRO_CIMC.4.1, FCO_NRO_CIMC.4.2
FUNC_CERTREQVER	FDP_CIMC_CER.1.3
FUNC_CRLSGENE	FDP_CIMC_CRL.1.1, FMT_MOF_CIMC.5.1
FUNC_CERTPROF	FMT_MOF_CIMC.3.1, FMT_MOF_CIMC.3.2, FMT_MOF_CIMC.3.3, FMT_MOF_CIMC.3.4
FUNC_REVPROF	FMT_MOF_CIMC.5.1, FMT_MOF_CIMC.5.2, FMT_MOF_CIMC.5.3
FUNC_OCSPPROF	FMT_MOF_CIMC.6.1, FMT_MOF_CIMC.6.2, FMT_MOF_CIMC.6.3
FUNC_BACKUP	FDP_CIMC_BKP.1.1, FDP_CIMC_BKP.1.2, FDP_CIMC_BKP.1.3, FDP_CIMC_BKP.1.4,
FUNC_NDCCPVER	FCO_NRO_CIMC.3.3

Tabla 6-4. Tabla de correspondencia entre las funciones de seguridad y los requisitos funcionales

6.3 Fortaleza de las funciones

Este TOE puede operar en un rango de entornos que va desde los entornos benignos hasta los hostiles. El nivel mínimo de fortaleza de las funciones del TOE y de los requisitos funcionales de seguridad del entorno SOF-basic. Sin embargo, el nivel SOF-basic se aplicará excepto donde sea requerido otro grado de fortaleza de los requisitos de función, tal como se especifica a lo largo de esta sección. El sistema KeyOne incluye mecanismos de seguridad, utilizados, por ejemplo, por el servicio de autenticación, que utilizan mecanismos probabilísticos o permutativos.

La fortaleza de los algoritmos criptográficos queda fuera del ámbito de los *Common Criteria*. La fortaleza de las funciones sólo es de aplicación a los mecanismos probabilísticos y permutacionales que no sean criptográficos. Por consiguiente, la demanda mínima SOF que se incluye en esta declaración de seguridad no se aplica a ningún mecanismo criptográfico, por lo que respecta a la evaluación *Common Criteria*.

6.3.1 Mecanismos de autenticación

Los mecanismos de autenticación que se especifican en FIA_UAU.1 iteraciones 1 y 2 cumplirán con los siguientes requisitos, en cuanto a la fortaleza de las funciones:

1. Para cada intento de autenticación, existe una posibilidad inferior a uno entre 1.000.000 de que un intento aleatorio tenga éxito o se produzca una falsa aceptación (e.g. adivinar una contraseña o PIN, índice de falsas aceptaciones de un dispositivo biométrico, combinación de métodos de autenticación).
2. Para múltiples intentos de usar el mecanismo de autenticación durante un periodo de un minuto, existe una posibilidad inferior a uno entre 100.000 de que un intento aleatorio tenga éxito o se produzca una falsa aceptación.

Este mecanismo de autenticación está relacionado con la función de seguridad FUNC_UIDAUT.

6.3.2 Módulos criptográficos

Los módulos criptográficos validados por el FIPS 140-2 deben realizar todas las funciones criptográficas realizadas por CIMCs. Los módulos criptográficos validados por el FIPS 140-2 también son necesarios para generar claves criptográficas y guardar claves privadas y secretas como texto.

6.3.2.1 Cifrado y módulos validados por el FIPS 140-2

Las referencias a FIPS 140-1 aluden a la versión más actualizada del estándar (disponible en <http://csrc.nist.gov/cryptval>).

6.3.2.2 Algoritmos de Cifrado

Algoritmos de cifrado especificados para:



FAU_STG.1 Almacenamiento seguro de trazas de auditoría

FCO_NRO_CIMC.4 Verificación avanzada del origen

FDP_ACF_CIMC.2 Protección de la confidencialidad de claves privadas de usuario

FDP_ACF_CIMC.3 Protección de la confidencialidad de claves secretas

FDP_CIMC_BKP.2 Copia de seguridad y recuperación extendida CIMC

FDP_ETC_CIMC.4 Exportación de claves privadas y secretas de usuario

FDP_ETC_CIMC.5 Exportación de claves privadas y secretas

FDP_SDI_CIMC.3 Control de la seguridad de claves públicas guardadas

FMT_MTD_CIMC.4 Protección de la confidencialidad de claves privadas del TSF

FMT_MTD_CIMC.5 Protección de la confidencialidad de claves secretas del TSF

FMT_MTD_CIMC.6 Exportación de claves secretas y privadas del TSF

FMT_MTD_CIMC.7 Exportación extendida de claves privadas y secretas del TSF

FPT_CIMC_TSP.1 Evento de firma de logs

FPT_CIMC_TSP.2 Evento de sellado de tiempo de logs

FPT_TST_CIMC.2 Test de integridad del software/firmware

FPT_TST_CIMC.3 Test de carga del software/firmware debe realizarse mediante un algoritmo aprobado o recomendado por el FIPS.

6.3.2.3 Módulos criptográficos Validados por el FIPS 140-2

Módulos criptográficos especificados para:

FCS_CKM.1 Generación de clave criptográfica

FDP_ACF_CIMC.2 Protección de la confidencialidad de claves privadas de usuario

FDP_ACF_CIMC.3 Protección de la confidencialidad de claves secretas de usuario

FDP_ETC_CIMC.4 Exportación de claves privadas y secretas de usuario

FDP_ETC_CIMC.5 Exportación extendida de claves privadas y secretas de usuario

FDP_SDI_CIMC.3 Control de la integridad de claves públicas guardadas

FMT_MTD_CIMC.4 Protección de la confidencialidad de claves privadas del TSF

FMT_MTD_CIMC.5 Protección de la confidencialidad de claves secretas del TSF

FMT_MTD_CIMC.6 Exportación de claves privadas y secretas del TSF

FMT_MTD_CIMC.7 Exportación extendida de claves privadas y secretas del TSF

FPT_CIMC_TSP.1 El evento de firma de logs debe estar validado según el FIPS 140-2.

6.3.2.4 Procedimientos de conocimiento parcial

Procedimientos de conocimiento parcial especificadas en:

FDP_ETC_CIMC.4 Exportación de claves privadas y secretas de usuario

FDP_ETC_CIMC.5 Exportación extendida de claves privadas y secretas de usuario

FMT_MTD_CIMC.6 Exportación de claves privadas y secretas del TSF

FMT_MTD_CIMC.7 La exportación extendida de claves privadas y secretas del TSF debe implementarse y validarse según lo especificado en FIPS 140-2.

6.3.2.5 Códigos de Autenticación

Códigos de autenticación especificados en:

FAU_STG.1 Almacenamiento seguro de las trazas de auditoría

FCO_NRO_CIMC.4 Verificación avanzada del origen

FDP_CIMC_BKP.2 Copia de seguridad y recuperación CIMC extendida

FPT_CIMC_TSP.1 Evento de firma de log de auditoría

FDP_SDI_CIMC.3 Control de la integridad de claves públicas almacenadas

FPT_TST_CIMC.2 Test de integridad del software/firmware

FPT_TST_CIMC.3 El test de carga del software/firmware debe incluir un código de autenticación aprobado o recomendado por el FIPS.

6.3.2.6 Niveles de módulos criptográficos para funciones criptográficas que implican claves privadas o secretas

Todas las operaciones criptográficas realizadas a petición del TOE (incluyendo la generación de claves) deben realizarse en un módulo criptográfico validado por el FIPS 140-1 que opere en un modo de operación aprobado o recomendado por el FIPS.

Tabla 6-5. Nivel FIPS 140-1 para módulo criptográfico validado especifica el nivel general FIPS 140-1 requerido para el módulo criptográfico validado de cada categoría de uso de una clave privada o secreta. Si el CIMC genera claves privadas de titular de certificado, debe aplicarse el nivel general FIPS 140-1 requerido para *Protección de Claves Privadas de Larga Duración*.

Nivel General FIPS 140-1 Requerido para Módulos Criptográficos CIMC	
<i>Categoría de Uso</i>	<i>Nivel de Seguridad 3</i>
Firma de Certificado y Estado de Certificado - Firma con una parte	3



- Firma con varias partes	2
Integridad o Aprobación de autenticidad	
- Aprobación individual	2
- Aprobación dual	2
Autenticación General	2
Protección de clave privacidad de largo plazo	3
Confidencialidad a largo plazo	2
Protección de clave privada a corto plazo	2
Confidencialidad a corto plazo	1

Tabla 6-5. Nivel FIPS 140-1 para módulo criptográfico validado.

6.3.2.7 Funciones criptográficas que no implican claves privadas o secretas

En los componentes del TOE pueden realizarse otras funciones criptográficas que no requieren claves privadas o secretas:

- a) *Generación de Hashes*: Durante la generación y verificación de firmas, pueden usarse funciones de hash unidireccionales (las firmas suelen generarse aplicando la clave privada al hash del mensaje). La generación de un hash no requiere una clave. Por ello, la generación de hashes no tiene los mismos requisitos de confidencialidad que otras funciones criptográficas.
- b) *Verificación de firma*: Las firmas se verifican a partir del texto de un mensaje y una clave pública. Los módulos criptográficos que sólo generan verificaciones de firma o hashes sin clave requieren un nivel general 3 de FIPS 140-1.

6.4 Medidas de control

La Metodología de Desarrollo de Safelayer, mercedores de un certificado ISO 9001:2000, cumplen los requisitos de control de la seguridad impuestos en el TOE.

Así consta en los siguientes documentos:

- Documentos generales de la compañía relativos al departamento de software:
 - QM "Safelayer Quality Manual", Código 28348ACB
Este documento define el Sistema de Gestión de Calidad de la Compañía.
 - DMP "Document Management Plan", Código 9D495947

Este documento establece las reglas y procedimientos de control para toda la documentación gestionada por la compañía.

- Proceso central de desarrollo
 - DM "Development Methods", Código A2A7DE72
Este documento define el ciclo de desarrollo desde un punto de vista técnico.
 - SSL "Software Security Lifecycle", Código 51D94682
Este documento aporta otras consideraciones sobre la implementación de medidas de seguridad. En concreto, este documento complementa los procesos básicos de desarrollo para dichas medidas.
- Disciplinas generales que dan soporte al proceso
 - QP "Quality Plan", Código D03B789F
Este documento describe las auditorías, comprobaciones, procesos y medidas que garantizan la excelencia y constante evolución de nuestro proceso de desarrollo de software.
 - SM "Software Management", Código 3857D336
Este documento establece actividades y técnicas de organización y gestión del desarrollo.
 - CM "Gestión de la configuración Plan", Código 411A0E26
Este documento establece los procedimientos de control de versiones, generación de códigos de identificación de software, distribución y ramificación, generación del histórico de desarrollo.
 - SEM "Security Manual", Código 987EEACF
Este documento regula los procedimientos que garantizan la seguridad en el desarrollo (los productos de seguridad sólo pueden obtenerse en un entorno de desarrollo seguro).
- Procedimiento detallado que puede aplicarse a áreas y actividades específicas
 - GOC "Guía para la Organización de Código Fuente en Safelayer ", Código 2B395E88
Este documento contiene las reglas y recomendaciones para organizar los archivos fuente y binarios de un proyecto.
Reglas y flags de compilación.
 - CCS "CVS – Control de Versiones", Código 0347E6C0
Este documento explica como utilizar debidamente el CVS.
 - MUATR "Manual de Uso de la ATR", Código E1B0EBCD
Este documento explica como utilizar la aplicación ATR.



- BUG "Safelayer Bugzilla Usage Guidelines", Código A790CDA4
Este documento describe el uso de la base de datos de Propuestas de Cambios de Ingeniería y el sistema de soporte.
- DSUG "Manual de Usuario del Servidor de Documentación de Safelayer", Código 5C4AC6D9
Este documento es una guía de uso del sistema de gestión de la documentación (donde por ejemplo se definen los códigos de referencia utilizados para cada documento).
- PDP "Product Secure Delivery Procedures", Código 2DB0CC43
Este documento detalla los procedimientos de distribución con los correspondientes niveles de seguridad.

La documentación específica del TOE contribuye al cumplimiento de los requisitos de control seleccionados. La siguiente tabla describe dichos requisitos:

Clase de control	Requisito de control	Título del documento
Gestión de la configuración	ACM_AUT.1	Configuration Management
	ACM_CAP.4	Manual de uso de la ATR
	ACM_SCP.2	Safelayer Bugzilla Usages Guidelines
Distribución y Operación	ADO_DEL.2	Product Secure Delivery Procedures

	ADO_IGS.1	<p>KeyOne Console 3.0 Administration and User Manual</p> <p>KeyOne CA 3.0 - Administration Manual</p> <p>KeyOne VA 3.0 - Manual Installation and Uninstallation Manual for KeyOne 3.0 Products</p> <p>Signing scripts</p> <p>KeyOne CRL Authority 3.0 Manual</p> <p>KeyOne RA 3.0 Manual</p> <p>KeyOne LRA 3.0 User and Administration Manual</p> <p>KeyOne TSA 3.0 Manual</p> <p>KeyOne 3.0 Logs Registration Administration</p> <p>KeyOne 3.0 Batch Format Description</p> <p>KeyOne 3.0 Master Document</p> <p>KeyOne 3.0 i3D Database Management</p> <p>KeyOne 3.0 Template Textual Specification Syntax</p>
Desarrollo	ADV_FSP.2	KeyOne 3.0 Functional Specification
	ADV_HLD.2	KeyOne 3.0 High Level Design



	ADV_LLD.1	KeyOne 3.0 Low Level Design
	ADV_RCR.1	KeyOne 3.0 Functional Specification KeyOne 3.0 High Level Design KeyOne 3.0 Low Level Design
	ADV_SPM.1	KeyOne 3.0 Security Policy
	ADV_IMP.1	KeyOne 3.0 Low Level Design

<p>Guías de referencia</p>	<p>AGD_ADM.1</p>	<p>KeyOne Console 3.0 Administration and User Manual</p> <p>KeyOne CA 3.0 - Administration Manual</p> <p>KeyOne VA 3.0 - Manual</p> <p>Certificate, Keys and CRL Management in KeyOne 3.0</p> <p>KeyOne CRL Authority 3.0 Manual</p> <p>KeyOne RA 3.0 Manual</p> <p>KeyOne LRA 3.0 User and Administration Manual</p> <p>KeyOne TSA 3.0 Manual</p> <p>KeyOne 3.0 Logs Registration Administration</p> <p>KeyOne 3.0 Batch Format Description</p> <p>KeyOne 3.0 Master Document</p> <p>KeyOne 3.0 i3D Database Management</p> <p>KeyOne 3.0 Template Textual Specification Syntax</p>
----------------------------	------------------	--



	AGD_USR.1	<p>KeyOne Console 3.0 Administration and User Manual</p> <p>KeyOne CA 3.0 – User Manual</p> <p>KeyOne VA 3.0 - Manual</p> <p>Certificate, Keys and CRL Management in KeyOne 3.0 Applications</p> <p>KeyOne CRL Authority 3.0 Manual</p> <p>KeyOne RA 3.0 Manual</p> <p>KeyOne LRA 3.0 User and Administration Manual</p> <p>KeyOne TSA 3.0 Manual</p> <p>KeyOne 3.0 Logs Registration Administration</p> <p>KeyOne 3.0 Batch Format Description</p> <p>KeyOne 3.0 Master Document</p> <p>KeyOne 3.0 i3D Database Management</p> <p>KeyOne 3.0 Template Textual Specification Syntax</p>
Soporte del ciclo de vida	ALC_DVS.1	<p>Security Manual</p> <p>Software Security Lifecycle</p>
	ALC_TAT.1	Guía para la organización del código fuente
	ALC_FLR.2	<p>Product Nonconformities Handling Procedures</p> <p>Safelayer Bugzilla Usage Guidelines</p>
	ALC_LCD.1	Software Security Lifecycle
Tests	ATE_COV.2	Quality Assurance-Test Plan
	ATE_FUN.1	<p>Quality Assurance-Test Plan</p> <p>Quality Assurance-Test Description</p> <p>Quality Assurance-Test Result</p>

	ATE_IND.2	Quality Assurance-Test Plan
	ATE_DPT.1	Quality Assurance-Test Plan
Control de la vulnerabilidad	AVA_SOF.1	KeyOne 3.0 - Security Target KeyOne 3.0 - Strength of TOE Security Function Analysis
	AVA_MSU.2	KeyOne 3.0 Misuse Analysis
	AVA_VLA.2	KeyOne 3.0 Vulnerability Analysis

Tabla 6-6. Documentación de requisitos de control de la seguridad

6.5 Funciones de seguridad que usan mecanismos probabilísticos o permutacionales

Las siguientes funciones de seguridad descritas en "KeyOne 3.0 - Functional Specification, internal code: 6D6436D9" utilizan mecanismos criptográficos.

Security Function
FUNC_DBIV
FUNC_CDBC
FUNC_I3DSESSION
FUNC_I3DHISTORIC
FUNC_BATCHSIG
FUNC_BATCHVER
FUNC_NDCCPSIGCA
FUNC_NDCCPVER
FUNC_K1SSLTLS
FUNC_OBFUSCATION



FUNC_OCSPRES
FUNC_UIDAUT
FUNC_CERTREQVER
FUNC_CERTSGENE
FUNC_PKCSGENE
FUNC_CRLSGENE
FUNC_CERTPROF
FUNC_REVPROF
FUNC_OCSPPROF
FUNC_PSSINSERT
FUNC_KEYRECOV
FUNC_BACKUP

7 Reivindicaciones

El TOE está adecuado al Nivel de seguridad 3 de Perfil de Protección de Componentes de Gestión y Emisión de Certificados (CIMC) (que aumenta las especificaciones de EAL3) detallado por NIST el 31 de octubre de 2001.

Además, KeyOne 3.0 cumple todos los Requisitos de Confianza para el nivel de certificación de Criterios Comunes EAL4, aumentado con ALC_FLR.2. Estos Requisitos de Confianza son los siguientes:

- ACM_CAP.4 Apoyo a la generación y procedimientos de aceptación
- ACM_AUT.1 Automatización CM parcial
- ALC_LCD.1 Modelo de ciclo de vida definido del desarrollador

Los requisitos de seguridad FPT_CIMC_TSP.1.3 se consiguen porque para cada modificación (adición, actualización o eliminación) de los registros de la base de datos, el mecanismo i3D garantiza la generación de una firma digital que asegura la integridad de la base de datos, entonces el sistema KeyOne trabaja como si estuviera configurado en la máxima frecuencia, y de ese modo la frecuencia más segura (refinamiento de los requerimientos FPT_CIMC_TSP.1.3).

8 Razonamiento

This section includes the rationale for the functional and assurance requirements specified for the TOE.

The rationale is based on specified objectives, threats, assumptions, and policies.

8.1 Security Objectives Rationale

This section demonstrates that the stated security objectives counter all identified threats, policies, or assumptions.

8.1.1 Security Objectives Coverage

The following tables provide a mapping of security objectives to the environment defined by the threats, policies, and assumptions, illustrating that each security objective covers at least one threat, policy or assumption and that each threat, policy or assumption is covered by at least one security objective. The Table X maps security objectives for the TOE to threats, Table Y maps security objectives for the environment to threats, and Table Z maps security objectives for both the TOE and the environment to threats. Table XX maps the organizational security policies to security objectives. Table YY maps assumptions to IT security objectives, listing which objectives each assumption helps to cover. The items in the tables are ordered alphabetically, sorted on the first column.

<i>IT Security Objective</i>	<i>Threat</i>
O.Certificates	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Control unknown source communication traffic	T.Hacker gains access
O.Non-repudiation	T.Sender denies sending information
O.Preservation/trusted recovery of secure state	T.Critical system component fails
O.Sufficient backup storage and effective restoration	T.Critical system component fails, T.User error makes data inaccessible

Table 8-1. Relationships of Security Objectives for the TOE to Threats

Non-IT Security Objective	Threat
O.Administrators, Operators, Officers and Auditors guidance documentation	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions , T.Social engineering
O.Competent Administrators, Operators, Officers and Auditors	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.CPS	T.Administrative errors of omission
O.Cryptographic functions	T.Disclosure of private and secret keys, T.Modification of secret/private keys
O.Installation	T.Critical system component fails
O.Lifecycle security	T.Critical system component fails, T.Malicious code exploitation
O.Notify Authorities of Security Issues	T.Hacker gains access
O.Periodically check integrity	T.Malicious code exploitation
O.Physical Protection	T.Hacker physical access
O.Repair identified security flaws	T.Flawed code , T.Critical system component fail
O.Security roles	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Social Engineering Training	T.Social Engineering
O.Trusted path	T.Hacker gains access, T.Message content modification
O.Validation of security function	T.Malicious code exploitation, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions

Table 8-2. Relationship of Security Objectives for the Environment to Threats

Non-IT Security Objective	Threat
O.Object and data recovery free from malicious code	T.Modification of secret/private keys, T.Malicious code exploitation
O.Procedures for preventing malicious code	T.Malicious code exploitation, T.Social engineering
O.Protect stored audit records	T.Modification of secret/private keys, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Protect user and TSF data during internal	T.Message content modification,

transfer	T.Disclosure of private and secret keys
O.React to detected attacks	T.Hacker gains access
O.Require inspection for downloads	T.Malicious code exploitation
O.Respond to possible loss of stored audit records	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Restrict actions before authentication	T.Hacker gains access, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Security-relevant configuration management	T.Administrative errors of omission
O.Time stamps	T.Critical system component fails, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Configuration management	T.Critical system component fails, T.Malicious code exploitation
O.Data import/export	T.Message content modification
O.Detect modifications of firmware, software, and backup data	T.User error makes data inaccessible, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Individual accountability and audit records	T.Administrative errors of omission, T.Hacker gains access, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions, T.User abuses authorization to collect and/or send data
O.Integrity protection of user data and software	T.Modification of private/secret keys, T.Malicious code exploitation
O.Limitation of administrative access	T.Disclosure of secret and private keys, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Maintain user attributes	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions



O.Manage behavior of security functions	T.Critical system component fails, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions

Table 8-3. Relationship of Security Objectives for Both the TOE and the Environment to Threats

Security Policy	Security Objective
P.Authorized use of information	O.Auditors review audit logs O.Maintain user attributes O.Restrict actions before authentication O.Security roles O.User authorization management
P.Cryptography	O.Cryptographic functions

Table 8-4. Relationship of Security Policies to Security Objectives

Assumption	IT Security Objective
A.Auditors Review Audit Logs	O.Auditors Review Audit Logs
A.Authentication Data Management	O.Authentication Data Management
A.Communications Protection	O.Communications Protection
A.Competent Administrators, Operators, Officers and Auditors	O.Competent Administrators, Operators, Officers and Auditors, O.Installation, O.Security-relevant configuration management, O.User authorization management, O.Configuration Management
A.Cooperative Users	O.Cooperative Users
A.CPS	O.CPS, O.Security-relevant configuration management, O.User authorization management, O.Configuration Management
A.Disposal of Authentication Data	O.Disposal of Authentication Data
A.Malicious Code Not Signed	O.Procedures for preventing malicious code,

	O.Require inspection for downloads, O.Malicious Code Not Signed
A.Notify Authorities of Security Issues	O.Notify Authorities of Security Issues
A.Operating System	O.Operating System
A.Physical Protection	O.Physical Protection
A.Social Engineering Training	O.Social Engineering Training
A.NTP Client	O.Time Stamp

Table 8-5. Relationship of Assumptions to IT Security Objectives

8.1.2 Security Objectives Sufficiency

The following discussions provide information regarding:

1. Why the identified security objectives provide for effective countermeasures to the threats;
2. Why the identified security objectives provide complete coverage of each organizational security policy;
3. Why the identified security objectives uphold each assumption.

8.1.2.1 Threats and Objectives Sufficiency

8.1.2.1.1 Authorized users

T.Administrative errors of omission addresses errors that directly compromise organizational security objectives or change the technical security policy enforced by the system or application.

It is countered by:

O.CPS provides Administrators, Operators, Officers, and Auditors with information regarding the policies and practices used by the system. Providing this information ensures that these authorized users of the system are aware of their responsibilities, thus reducing the likelihood that they will fail to perform a security-critical operation.

O.Individual accountability and audit records provides individual accountability for audited events. Each user is uniquely identified so that auditable actions can be traced to a user. Audit records provide information about past user behavior to an authorized individual through system mechanisms. These audit records will expose administrators that fail to perform security-critical operations so they can be held accountable.

O.Security-relevant configuration management ensures that system security policy data and enforcement functions, and other security-relevant configuration data are managed and updated. This ensures that they are



consistent with organizational security policies and that all changes are properly tracked and implemented.

T.User abuses authorization to collect and/or send data addresses the situation where an authorized user abuses granted authorizations by browsing files in order to collect data and/or violates export control policy by sending data to a recipient who is not authorized to receive the data.

It is countered by:

O.Individual accountability and audit records provides individual accountability for audited events. Each user is uniquely identified so that auditable actions can be traced to a user. Audit records provide information about past user behavior to an authorized individual through system mechanisms. This audit records will expose users who abuse their authorized to collect and/or send data.

T.User error makes data inaccessible addresses a user accidentally deleting user data. Consequently, the user data is inaccessible. Examples include the following:

- User accidentally deletes data by striking the wrong key on the keyboard or by striking the enter key as an automatic response.
- User does not understand the implications of the prompt at hand and inadvertently gives a response that deletes user data.
- User misunderstands a system command and issues a command that unintentionally deletes user data.

It is countered by:

O.Sufficient backup storage and effective restoration ensures that there is sufficient backup storage and effective restoration to recreate the system, when required. This ensures that user data is available from backup, even if the current copy is accidentally deleted.

O.Detect modifications of firmware, software, and backup data ensures that if the backup components have been modified, that it is detected. If modifications of backup data can not be detected, the backup copy is not a reliable source for restoration of user data.

T.Administrators, Operators, Officers and Auditors commit errors or hostile actions addresses:

- Errors committed by administrative personnel that directly compromise organizational security objectives, change the technical security policy enforced by the system or application, or
- Malicious obstruction by administrative personnel of organizational security objectives or modification of the system's configuration to allow security violations to occur.

It is countered by:

O.Competent Administrators, Operators, Officers and Auditors ensures that users are capable of maintaining effective security practices. This reduces the likelihood that they will commit errors.



O.Administrators, Operators, Officers and Auditors guidance documentation which deters administrative personnel errors by providing adequate guidance.

O.Certificates ensures that certificates, certificate revocation lists, and certificate status information are valid. The validation of information provided by Officers that is to be included in certificates helps to prevent improperly entered information from appearing in certificates.

O.Detect modifications of firmware, software, and backup data ensures that if the backup components have been modified, that it is detected.

O.Individual accountability and audit records provides individual accountability for audited events. Each user is uniquely identified so that auditable actions can be traced to a user. Audit records provide information about past user behavior to an authorized individual through system mechanisms. These audit records will expose administrators that perform inappropriate operations so they can be held accountable.

O.Limitation of administrative access. The administrative functions are designed in such a way that administrative personnel do not automatically have access to user objects, except for necessary exceptions. In general, the exceptions tend to be role specific. Limiting the set of operations that a user may perform limits the damage that a user may cause.

O.Maintain user attributes. Maintains a set of security attributes (which may include group membership, access rights, etc.) associated with individual users in addition to user identity. This prevents users from performing operations that they are not authorized to perform.

O.Manage behavior of security functions provides management controls/functions for security mechanisms. This ensures that security mechanisms which protect against hostile users are properly configured.

O.Protect stored audit records ensures that audit records are protected against unauthorized access, modification, or deletion to provide for traceability of user actions.

O.Respond to possible loss of stored audit records ensures that only auditable events executed by the Auditor shall be audited if the audit trail is full. This ensures that operations that are performed by users other than the Auditor are audited and so can be detected.

O.Restrict actions before authentication ensures that only a limited set of actions may be performed before a user is authenticated.

O.Security roles ensures that security-relevant roles are specified and that users are assigned to one (or more) of the defined roles. This prevents users from performing operations that they are not authorized to perform.

O.Time stamps ensures that time stamps are provided to verify a sequence of events. This allows the reconstruction of a timeline of events when performing an audit review.



O.Validation of security function. Ensure that security-relevant software, hardware, and firmware are correctly functioning through features and procedures such as underlying machine testing and integrity checks.

8.1.2.1.2 System

T.Critical system component fails addresses the failure of one or more system components that results in the loss of system-critical functionality. This threat is relevant when there are components that may fail due to hardware and/or software imperfections and the availability of system functionality is important.

It is countered by:

O.Configuration management assures that a configuration management program is implemented. The configuration management program includes configuration identification and change control. This ensures that critical system components do not fail as a result of improper configuration.

O.Installation ensures that the TOE is delivered, installed, managed, and operated in a manner which maintains IT security. This ensures that critical system components do not fail as a result of improper installation.

O.Manage behavior of security functions provides management controls/functions for security mechanisms. This ensures that critical system components do not fail as a result of improper configuration of security mechanisms.

O.Preservation/trusted recovery of secure state ensures that the system remains in a secure state throughout operation in the presence of failures and subsequent system recovery. This objective is relevant when system failures could result in insecure states that, when the system returns to operational mode (or continues to operate), could lead to security compromises.

O.Sufficient backup storage and effective restoration ensures that there is sufficient backup storage and effective restoration to recreate the system, when required. This ensures that data is available from backup, even if the current copy is lost through failure of a system component (e.g., a disk drive).

O.Time stamps provides time stamps to ensure that the sequencing of events can be verified. If the system must be reconstructed, it may be necessary to establish the order in which transactions were performed to return the system to a state consistent with the state when a critical component failed.

O.Lifecycle security provides tools and techniques that are used throughout the development phase reducing the likelihood of hardware or software imperfections. **O.Lifecycle security** also addresses the detection and resolution of flaws discovered during the operational phase that may result in failure of a critical system component.

O.repair identified security flaws. The vendor repairs security flaws that have been identified by a user. Such security flaws may result in critical system component failures if not repaired.

T.Flawed code addresses accidental or deliberate flaws in code made by the developer. Examples of accidental flaws are lack of engineering detail or bad design. An example of a deliberate flaw would be the inclusion of a trapdoor for later entry into the TOE.

It is countered by:

O.Repair identified security flaws ensures that identified security flaws are repaired.

T.Malicious code exploitation addresses the threat where an authorized user, IT system, or hacker downloads and executes malicious code, which causes abnormal processes that violate the integrity, availability, or confidentiality of the system assets. The execution of malicious code is done through a triggering event.

It is countered by:

O.Configuration management assures that a configuration management program is implemented. The configuration management program includes configuration identification and change control. This ensures that malicious code is not introduced during the configuration process.

O.Integrity protection of user data and software ensures that appropriate integrity protection is provided for user data and software. This prevents malicious code from attaching itself to user data or software.

O.Object and data recovery free from malicious code ensures that the system recovers to a viable state after malicious code has been introduced and damage has occurred. The malicious code, e.g., virus or worm, is removed as part of the process.

O.Periodically check integrity ensures that periodic integrity checks are performed on both system and software. If these checks fail, malicious code may have been introduced into the system.

O.Procedures for preventing malicious code provides a set of procedures and mechanisms that work to prevent incorporation of malicious code into the system.

O.Require inspection for downloads ensures that software that is downloaded/transferred is inspected prior to being made operational.

O.Validation of security function. Ensure that security-relevant software, hardware, and firmware are correctly functioning through features and procedures such as underlying machine testing and integrity checks.

O.Lifecycle security provides tools and techniques that are used throughout the development phase, reducing the likelihood that malicious code was included in the product by the developer.

O.Lifecycle security also addresses the detection and resolution of flaws discovered during the operational phase, such as modifications of components by malicious code.

T.Message content modification addresses the situation where a hacker modifies information that is intercepted from a communications link between two unsuspecting entities before passing it on to the intended recipient. Several kinds of



modification are possible: modification of a single message, deletion or reordering of selected messages, insertion of bogus messages, replay of previous messages, and modification of accompanying message security attributes.

It is countered by:

O.Data Import/Export protects data when being transmitted to or from the TOE. Protection of data in transit permits the TOE or the external user to detect modified messages, message replay, or fraudulent messages.

O.Protect user and TSF data during internal transfer protects data being transmitted between separated parts of the TOE. Protection of data in transit permits the TOE to detect modified messages, message replay, or fraudulent messages.

O.Trusted path ensures that a trusted path is established between the user and the system. The trusted path protects messages from interception or modification by a hacker.

8.1.2.1.3 Cryptography

T.Disclosure of private and secret keys addresses the unauthorized disclosure of secret and/or private keys.

It is countered by:

O.Administrators, Operators, Officers and Auditors guidance documentation ensures that adequate documentation on securely configuring and operating the CIMC is available to Administrators, Operators, Officers and Auditors. This documentation will minimize errors committed by those users.

O.Cryptographic functions ensures that TOE implements approved cryptographic algorithms for encryption/decryption, authentication, and signature generation/verification; approved key generation techniques and uses validated cryptographic modules. Use of validated cryptographic modules ensures that cryptographic keys are adequately protected when they are stored within cryptographic modules.

O.Limitation of administrative access. The administrative functions are designed in such a way that administrative personnel do not automatically have access to user objects, except for necessary exceptions. In general, the exceptions tend to be role specific. Limiting the number of users who have access to cryptographic keys reducing the likelihood of unauthorized disclosure.

O.Protect user and TSF data during internal transfer protects private and secret keys from unauthorized disclosure during transmission between separated parts of the TOE.

T.Modification of private/secret keys addresses the unauthorized revision of a secret and/or private key.

It is countered by:

O.Cryptographic functions ensures that TOE implements approved cryptographic algorithms for encryption/decryption, authentication, and



signature generation/verification; approved key generation techniques and uses validated cryptographic modules. Use of validated cryptographic modules ensures that cryptographic keys are adequately protected when they are stored within cryptographic modules.

O.Integrity protection of user data and software that ensures that appropriate integrity protection is provided for secret and private keys.

O.Object and data recovery free from malicious code ensures that the system recovers to a viable state after malicious code has been introduced and damage has occurred. If the malicious code cause private or secret keys to be revised in an unauthorized manner, this objective ensures that they are recovered to their correct values.

O.Protect stored audit records ensures that audit records are protected against unauthorized access, modification, or deletion to provide for traceability of user actions. This objective ensures that modifications to private and secret keys can be detected through the audit trail.

T.Sender denies sending information addresses the situation where the sender of a message denies sending the message to avoid accountability for sending the message and for subsequent action or inaction.

It is countered by:

O.Non-repudiation which ensures that the sender/originator of a message cannot successfully deny sending the message to the recipient.

8.1.2.1.4 External Attacks

T.Hacker gains access addresses:

- Weak system access control mechanisms or user attributes
- Weak implementation methods of the system access control
- Vulnerabilities found in system or application code that allow a hacker to break into a system undetected.

It is countered by:

O.Restrict actions before authentication ensures that only a limited set of actions may be performed before a user is authenticated. This prevents a hacker who is unable to circumvent the access control mechanisms from performing security-relevant operations.

O.Control unknown source communication traffic ensures that communication traffic from an unknown source is controlled (e.g., rerouted or discarded) to prevent potential damage. Various kinds of hacker attacks can be detected or prevented by rerouting or discarding suspected hacker traffic.

O.Individual accountability and audit records provides individual accountability for audited events. Each user is uniquely identified so that auditable actions can be traced to a user. Audit records provide information about past user behavior to an authorized individual through system



mechanisms. This allows for the detection of unauthorized activity. Once detected, the damage resulting from such activity can be eliminated or mitigated.

O.Notify Authorities of Security Issues ensures that proper authorities are notified regarding any security issues that impact their systems. This minimizes the potential for the loss or compromise of data.

At this Security Level it is also countered by:

O.React to detected attacks ensures that automated notification or other reactions to the TSF discovered attacks is implemented in an effort to identify attacks and to create an attack deterrent. This objective is relevant if actions that the organization deems essential also pose a potential attack that could be exploited.

At this Security Level it is also countered by:

O.Trusted path ensures that a trusted path is established between the user and the system. The trusted path is used to protect authentication data, thus reducing the likelihood that a hacker can masquerade as an authorized user.

T.Hacker physical access addresses the threat where an individual exploits physical security weaknesses to gain physical control of system components.

It is countered by:

O.Physical Protection ensures that physical access controls are sufficient to thwart a physical attack on system components.

T.Social Engineering addresses the situation where a hacker uses social engineering techniques to gain information about system entry, system use, system design, or system operation.

It is countered by:

O.Administrators, Operators, Officers and Auditors guidance documentation which deters administrative personnel errors by providing adequate guidance.

O.Procedures for preventing malicious code provides a set of procedures and mechanisms that work to prevent incorporation of malicious code into the system. The introduction of malicious code into the system may be a goal of the social engineering attack.

O.Social Engineering Training which ensures that general users, Administrators, Operators, Officers, and Auditors are trained in techniques to thwart social engineering attacks.

8.1.2.2 *Policies and Objectives Sufficiency*

P.Authorized use of information establishes that information is used only for its authorized purpose(s).

This is addressed by the following objectives: **O.Maintain user attributes**, **O.Restrict actions before authentication**, **O.Security roles**, and **O.User authorization management**. **O.Restrict actions before authentication** ensures that the capability to perform security-relevant operations is limited to those who have been authorized to perform those operations. **O.Maintain user attributes**, **O.Security roles**, and **O.User authorization management** ensure that users are only authorized to perform those operations that are necessary to perform their jobs. Finally, **O.Auditors review audit logs** deters users from misusing the authorizations they have been provided.

P.Cryptography establishes that accepted cryptographic standards and operations shall be used in the design of the TOE. This is addressed by **O.Cryptographic functions** which ensures that such standards are used.

8.1.2.3 Assumptions and Objectives Sufficiency

8.1.2.3.1 Personnel

A.Auditors Review Audit Logs establishes that audit logs are necessary for security-relevant events and that they must be reviewed by auditors. This is addressed by **O.Auditors Review Audit Logs**, which ensures that security-relevant events recorded in audit logs are reviewed by auditors.

A.Authentication Data Management establishes that management of user authentication data is external to the TOE. This is addressed by **O.Authentication Data Management**, which ensures that users modify their authentication data in accordance with appropriate security policy.

A.Competent Administrators, Operators, Officers and Auditors establishes that security of the TOE is dependent upon those that manage it. This is addressed by the following objectives:

- **O.Competent Administrators, Operators, Officers and Auditors**, which ensures that the system managers will be competent in its administration.
- **O.Installation**, which ensures that the responsible for the TOE ensures that the TOE is delivered, installed, managed and operated in a manner which maintains IT security.
- **O.Security-relevant configuration management**, which ensures that the organizational security policies are consistent with the system security policy data, enforcement functions, and other security-relevant configuration data.
- **O.Configuration Management**, which ensures that the system connectivity (software, hardware and firmware) and components (software, hardware and firmware) are identified, that the configuration data are audited, and that the changes to the configuration items are controlled.

A.CPS establishes that Administrators, Operators, Officers, and Auditors are familiar with the CP and CPS under which the TOE is operated. This is addressed by the following objectives:

- **O.CPS**, which ensures that Administrators, Operators, Officers, and Auditors are familiar with the CP and CPS under which the TOE is operated.



- **O.Security-relevant configuration management**, which ensures that the organizational security policies are consistent with the system security policy data, enforcement functions, and other security-relevant configuration data.
- **O.User authorisation management**, which ensures that the user authorisation and privilege data are consistent with organizational security and personnel policies.
- **O.Configuration Management**, which ensures that the system connectivity (software, hardware and firmware) and components (software, hardware and firmware) are identified, that the configuration data are audited, and that the changes to the configuration items are controlled. **A.Malicious Code Not Signed** establishes that code not designed for the TOE will not be signed by a trusted party. This is addressed by the following objectives:
- **O.Malicious Code Not Signed**, which ensures that code must be signed by a trusted party or it will not be loaded onto the system.
- **O.Procedures for preventing malicious code**, which incorporates malicious code prevention procedures and mechanisms.
- **O.Require inspection for downloads**, which ensures inspection of downloads/transfers.

A.Disposal of Authentication Data establishes that users shall not retain access to the system after their authorization has been removed. This is addressed by **O.Disposal of Authentication Data**, which ensures that access to the system will be denied after a user's privileges have been removed. **A.Malicious Code Not Signed** establishes that code not designed for the TOE will not be signed by a trusted party. This is addressed by **O.Malicious Code Not Signed**, which ensures that code must be signed by a trusted party or it will not be loaded onto the system.

A.Notify Authorities of Security Issues establishes that users notify proper authorities of any security issues that impact their systems to minimize the potential for the loss of compromise of data. This is addressed by **O.Notify Authorities of Security Issues** which ensures that user notify proper authorities of any security issues that impact their systems.

A.Social Engineering Training establishes that individuals will attempt to gain access to the system using social engineering practices. This is addressed by **O.Social Engineering Training**, which ensures that all users will be training to thwart social engineering attacks.

A.Cooperative Users establishes that a secure IT environment is required to securely operate the TOE, and that users must work within the constraints of that environment. This is addressed by **O.Cooperative Users**, which ensures that users will cooperate with the constraints established.

8.1.2.3.2 Connectivity

A.Operating System establishes that an insecure operating system will compromise system security. This is addressed by **O.Operating System**, which ensures that an operating system that meets security requirements recommended by the National Institute of Standards and Technology will be used.

A.NTP Client establishes that an erroneous time provided by the system clock where the components of the TOE are running, will compromise system security. This is addressed by **O.Time Stamp**, which provides time stamps.

8.1.2.3.3 Physical

A.Communications Protection establishes that the communications infrastructure is outside the TOE. This is addressed by **O.Communications Protection**, which ensures that adequate physical protections are afforded the necessary communications infrastructure.

A.Physical Protection establishes that physical modification of the TOE hardware, software, and firmware will compromise system security. This is addressed by **O.Physical Protection**, which ensures that adequate physical protection will be provided.

8.2 Security Requirements Rationale

This section provides the rationale for necessity and sufficiency of security requirements, demonstrating that each of the security objectives is addressed by at least one security requirement, and that every security requirement is directed toward solving at least one objective.

8.2.1 Security Requirements Coverage

The following tables provide a mapping of the relationships of security requirements to objectives, illustrating that each security requirement covers at least one objective and that each objective is covered by at least one security requirement. The first table in this section, Table 8.6, addresses the mapping of security functional requirements to security objectives. The second table, Table 8.7, addresses the mapping of security assurance requirements to security objectives.

<i>Functional Requirement</i>	<i>Objective</i>
FAU_GEN.1 Audit data generation (iterations 1 and 2)	O.Individual accountability and audit records
FAU_GEN.2 User identity association (iterations 1 and 2)	O.Individual accountability and audit records
FAU_SAR.1 Audit review	O.Individual accountability and audit records
FAU_SAR.3 Selectable audit review	O.Individual accountability and audit records
FAU_SEL.1 Selective Audit (iterations 1 and 2)	O.Individual accountability and audit records
FAU_STG.1 Protected audit trail storage (iterations 1 and 2)	O.Protect stored audit records
FAU_STG.4 Prevention of audit data loss (iterations 1 and 2)	O.Respond to possible loss of stored audit records
FCO_NRO_CIMC.3 Enforced proof of origin and verification of origin	O.Non-repudiation, O.Control unknown source communication traffic

FCO_NRO_CIMC.4 Advanced verification of origin	O.Non-repudiation
FCS_CKM.1 Cryptographic key generation	O.Cryptographic functions
FCS_CKM.4 Cryptographic key	O.Procedures for preventing malicious code, O.React to detected attacks
FCS_CKM_CIMC.5 CIMC private and secret key zeroization	O.Procedures for preventing malicious code, O.React to detected attacks
FCS_COP.1 Cryptographic operation	O.Cryptographic functions
FDP_ACC.1 Subset access control (iterations 1 and 2)	O.Limitation of administrative access
FDP_ACF.1 Security attribute based access control (iterations 1 and 2)	O.Limitation of administrative access
FDP_ACF_CIMC.2 User private key confidentiality protection	O.Certificates, O.Procedures for preventing malicious code
FDP_ACF_CIMC.3 User secret key confidentiality protection	O.Certificates, O.Procedures for preventing malicious code
FDP_CIMC_BKP.1 CIMC backup and recovery	O.Object and data recovery free from malicious code, O.Preservation/trusted recovery of secure state, O.Sufficient backup storage and effective restoration
FDP_CIMC_BKP.2 Extended CIMC backup and recovery	O.Detect modifications of firmware, software, and backup data, O.Object and data recovery free from malicious code
FDP_CIMC_CER.1 Certificate Generation	O.Certificates
FDP_CIMC_CRL.1 Certificate revocation list validation	O.Certificates
FDP_CIMC_CSE.1 Certificate status export	O.Certificates
FDP_CIMC_OCSP.1 OCSP basic response validation	O.Certificates
FDP_ETC_CIMC.5 Extended user private and gsecret key export	O.Data import/export
FDP_ITT.1 Basic internal transfer protection (iterations 1 and 3)	O.Integrity protection of user data and software, O.Protect user and TSF data during internal transfer
FDP_ITT.1 Basic internal transfer protection (iterations 2 and 4)	O.Protect user and TSF data during internal transfer
FDP_SDI_CIMC.3 Stored public key integrity monitoring and action	O.Integrity protection of user data and software
FDP_UCT.1 Basic data exchange confidentiality (iterations 1 and 2)	O.Data import/export
FIA_AFL.1 Authentication failure handling	O.React to detected attacks
FIA_ATD.1 User attribute definition	O.Maintain user attributes
FIA_UAU.1 Timing of authentication (iterations 1	O.Limitation of administrative access,

and 2)	O.Restrict actions before authentication
FIA_UID.1 Timing of identification (iterations 1 and 2)	O.Individual accountability and audit records, O.Limitation of administrative access
FIA_USB.1 User-subject binding (iterations 1 and 2)	O.Maintain user attributes
FMT_MOF.1 Management of security functions behavior (iterations 1 and 2)	O.Configuration management, O.Manage behavior of security functions, O.Security-relevant configuration management
FMT_MOF_CIMC.3 Extended certificate profile management	O.Configuration management
FMT_MOF_CIMC.5 Extended certificate revocation list profile management	O.Configuration management
FMT_MOF_CIMC.6 OCSP Profile Management	O.Configuration management
FMT_MSA.1 Management of security attributes	O.Maintain user attributes, O.User authorization management
FMT_MSA.2 Secure security attributes	O.Security-relevant configuration management
FMT_MSA.3 Static attribute initialisation	O.Security-relevant configuration management
FMT_MTD.1 Management of TSF data	O.Individual accountability and audit records, O.Protect stored audit records
FMT_MTD_CIMC.4 TSF private key confidentiality protection	O.Detect modifications of firmware, software, and backup data, O.Integrity protection of user data and software
FMT_MTD_CIMC.5 TSF secret key confidentiality protection	O.Detect modifications of firmware, software, and backup data, O.Integrity protection of user data and software
FMT_MTD_CIMC.7 Extended TSF private and secret key export	O.Data import/export
FMT_SMR.2 Restrictions on security roles	O.Security roles
FMT_SMF.1 Specification of Management Functions	O.Security roles, O.Maintain user attributes, O.Manage behaviour of security functions
FPT_AMT.1 Abstract machine testing	O.Periodically check integrity, O.Validation of security function
FPT_CIMC_TSP.1 Audit log signing event	O.Protect stored audit records
FPT_ITC.1 Inter-TSF confidentiality during transmission (iterations 1 and 2)	O.Data import/export
FPT_IIT.1 Basic internal TSF data transfer protection (iterations 1-4)	O.Protect user and TSF data during internal transfer
FPT_RVM.1 Non-bypassability of the TSP (iteration 1)	O.Operating System

FPT_RVM.1 Non-bypassability of the TSP (iteration 2)	O.Limitation of administrative access
FPT_SEP.1 TSF domain separation	O.Operating System
FPT_STM.1 Reliable time stamps (iterations 1 and 2)	O.Individual accountability and audit records, O.Time stamps
FPT_TST_CIMC.2 Software/firmware integrity test	O.Detect modifications of firmware, software, and backup data, O.Integrity protection of user data and software, O.Object and data recovery free from malicious code, O.Periodically check integrity, O.Procedures for preventing malicious code, O.Validation of security function
FPT_TST_CIMC.3 Software/firmware load test	O.Integrity protection of user data and software, O.Object and data recovery free from malicious code, O.Periodically check integrity, O.Require inspection for downloads
FTP_TRP.1 Trusted path	O.Trusted path
FPT_ACC.1 Access Control	O.Protect stored audit records

Table 8-6. Security Functional Requirements Related to Security Objectives

Assurance Requirement	Objective
ACM_AUT.1 Automation	EAL4, O.Configuration management
ACM_CAP.4 Generation support and acceptance procedures	EAL 4, O.Configuration management
ACM_SCP.2 Problem tracking CM Coverage	EAL 4, O.Configuration management
ADO_DEL.2 Detection of modification	EAL 4
ADO_IGS.1 Installation, Generation, and Start-up Procedures	EAL 4, O.Installation
ADV_FSP.2 Fully defined external interfaces	EAL 4, O.Lifecycle security
ADV_HLD.2 Security enforcing high-level design	EAL 4, O.Lifecycle security
ADV_IMP.1 Subset of the implementation of the TSF	EAL 4, O.Lifecycle security
ADV_LLD.1 Descriptive low-level design	EAL 4, O.Lifecycle security

ADV_RCR.1 Informal Correspondence Demonstration	O.Lifecycle security, EAL 4
ADV_SPM.1 Informal TOE security policy model	EAL 4, O.Lifecycle security
AGD_ADM.1 Administrator Guidance	O.Administrators, Operators, Officers and Auditors guidance documentation, O.Auditors Review Audit Logs, O.Competent Administrators, Operators, Officers and Auditors, O.Configuration Management, O.Installation, O.Malicious Code Not Signed, O.Procedures for preventing malicious code, O.Require inspection for downloads, O.Security-relevant configuration management, O.User authorization management, EAL 4
AGD_USR.1 User Guidance	O.Administrators, Operators, Officers and Auditors guidance documentation, O.Malicious Code Not Signed, O.Procedures for preventing malicious code, O.Require inspection for downloads, EAL 4
ALC_DVS.1 Identification of security measures	EAL 4
ALC_FLR.2 Flaw reporting procedures	O.Lifecycle security O.Repair identified security flaws, EAL4
ALC_LCD.1 Developer defined life-cycle model	EAL 4
ALC_TAT.1 Well-defined development tools	EAL 4
ATE_COV.2 Analysis of coverage	EAL 4
ATE_DPT.1 Testing - High-Level Design	EAL4
ATE_FUN.1 Functional testing	EAL 4



ATE_IND.2 Independent Testing	EAL 4
AVA_MSU.2 Validation of analysis	EAL 4
AVA_SOF.1 Strength of TOE Security Function Evaluation	EAL 4
AVA_VLA.2 Independent vulnerability analysis	EAL 4

Table 8-7. Security Assurance Requirements Related to Security Objectives

8.2.2 Security Requirements Sufficiency

Security Objectives for the TOE

8.2.2.1 Authorized Users

O.Certificates is provided by **FDP_CIMC_CER.1 (Certificate Generation)** which ensures that certificates are valid, and **FDP_CIMC_CRL.1 (Certificate revocation list validation)**, **FDP_CIMC_CSE.1 (Certificate status export)**, and **FDP_CIMC_OCSP.1 (OCSP basic response validation)** which ensure that certificate revocation lists and certificate status information are valid. In the case that the TOE maintains a copy of the certificate subject's private key, **FDP_ACF_CIMC.2 (User private key confidentiality protection)** ensures that the certificate is not invalidated by the disclosure of the private key by the TOE. In the case that a secret key is used by the certificate subject as an authenticator in requesting a certificate, **FDP_ACF_CIMC.3 (User secret key confidentiality protection)** ensures that an attacker can not obtain a bad certificate by obtaining a user's authenticator from the TOE and then using that authenticator to obtain a bad certificate.

8.2.2.2 System

O.Preservation/trusted recovery of secure state is provided by **FDP_CIMC_BKP.1 (CIMC backup and recovery)** which cover the requirement that the state of the system be preserved so that it can be recovered in the event of a secure component failure.

O.Sufficient backup storage and effective restoration is provided by **FDP_CIMC_BKP.1 (CIMC backup and recovery)** which cover the requirement that sufficient backup data is created and stored and that an effective restoration procedure is provided.

8.2.2.3 External Attacks

O.Control unknown source communication traffic is provided by **FCO_NRO_CIMC.3 (Enforced proof of origin and verification of origin)** which covers the requirement that the TOE discard messages from an unknown source that contain security-relevant information.

8.2.2.4 Cryptography

O.Non-repudiation is provided by **FCO_NRO_CIMC.3 (Enforced proof of origin and verification of origin)** which covers the requirement that messages containing security-

relevant data are not accepted by the TOE unless they contain evidence of origin and **FCO_NRO_CIMC.4 (Advanced verification of origin)** which covers the requirement that digital signatures be used so that the evidence of origin for a message may be verified by a third-party.

Non-IT Security Objectives for the Environment

O.Administrators, Operators, Officers and Auditors guidance documentation is provided by **AGD_ADM.1 (Administrator Guidance)** and **AGD_USR.1 (User Guidance)** which ensure that adequate guidance on the secure operation of the TOE is provided to Administrators, Operators, Officers, and Auditors.

O.Auditors Review Audit Logs is provided by **A.Auditors Review Audit Logs** which ensures that auditors review the audit logs. It is also supported by **AGD_ADM.1 (Administrator Guidance)** which ensures that Auditors are provided with the information they need to understand the contents of the audit logs.

O.Authentication Data Management is provided by **A.Authentication Data Management** which covers the requirement that an authentication data management policy be enforced.

O.Communications Protection is provided by **A.Communications Protection** which covers the requirement that the system be adequately physically protected against loss of communications.

O.Competent Administrators, Operators, Officers and Auditors is provided by **A.Competent Administrators, Operators, Officers and Auditors** which covers the requirement that Administrators, Operators, Officers, and Auditors be capable of managing the TOE and the security of the information it contains. It is also supported by **AGD_ADM.1 (Administrator Guidance)** which ensures that Administrators, Operators, Officers, and Auditors are provided with the information they need to properly manage the TOE and its security functionality.

O.CPS is provided by **A.CPS** which covers the requirement that Administrators, Operators, Officers, and Auditors be familiar with the CP and CPS under which the TOE is operated.

O.Installation is provided by **ADO_IGS.1 (Installation, Generation, and Start-up Procedures)** and **AGD_ADM.1 (Administrator Guidance)** which cover the requirement that Administrators, Operators, Officers, and Auditors be provided with documentation describing the procedures necessary to securely install and operate the TOE. **A.Competent Administrators, Operators, Officers and Auditors** covers the requirement that competent Administrators, Operators, Officers, and Auditors, who are capable of securely managing the TOE, are used.

O.Malicious Code Not Signed is provided by **A.Malicious Code Not Signed** which covers the requirement that malicious code destined for the TOE is not signed by a trusted entity. It is also supported by **AGD_ADM.1 (Administrator Guidance)** and **AGD_USR.1 (User Guidance)** which ensure that entities that are trusted to sign code are aware of their responsibilities.

O.Notify Authorities of Security Issues is provided by **A.Notify Authorities of Security Issues** which covers the requirement that proper authorities be notified of any security issues that impact their systems.



O.Physical Protection is provided by **A.Physical Protection** which covers the requirement that TOE hardware, software, and firmware critical to security policy enforcement be protected from unauthorized physical modification.

O.Social Engineering Training is provided by **A.Social Engineering Training** which covers the requirement that general users, administrators, operators, officers, and auditors are trained in techniques to thwart social engineering attacks.

O.Cooperative Users is provided by **A.Cooperative Users** which covers the requirement that users act in a cooperative manner.

O.Lifecycle security is provided by **ADV_FSP.2 (Fully defined external interfaces)**, **ADV_HLD.2 (Security enforcing high-level design)**, **ADV_LLD.1 (Descriptive low-level design)**, **ADV_RCR.1 (Informal correspondence demonstration)**, and **ADV_SPM.1 (Information TOE security policy model)** which cover the requirement that security is designed into the CIMC. **ALC_FLR.2 (Flaw reporting procedures)** that flaws are detected and resolved during the operational phase.

O.Repair identified security flaws is provided by **ALC_FLR.2 (Flaw reporting procedures)** which cover the requirement that vendor repair security flaws that have been identified by a user.

O.Disposal of Authentication Data is provided by **A.Disposal of Authentication Data**, which covers the requirement that authentication data be disposed of properly after access has been removed.

IT Security Objectives for the Environment

O.Cryptographic functions is provided by **FCS_CKM.1 (Cryptographic key generation)** and **FCS_COP.1 (Cryptographic operation)** which cover the requirement that approved algorithms be used for encryption/decryption, authentication, and signature generation/verification and that approved key generation techniques be used.

O.Operating System is provided by **A.Operating System** which covers the requirement that the operating system(s) on which the TSF operates provides security functions required by the CIMC to counter the perceived threats for the appropriate Security Level. It is also supported by **FPT_RVM.1 (Non-bypassability of the TSP) (iteration 1)** and **FPT_SEP.1 (TSF domain separation)** which ensure that the operating system(s) on which the TSF operates provides domain separation and non-bypassability.

O.Periodically check integrity is provided by **FPT_AMT.1 (Abstract machine testing)** which covers the requirement provide periodic integrity checks on the system and **FPT_TST_CIMC.2 (Software/firmware integrity test)** and **FPT_TST_CIMC.3 (Software/firmware load test)** cover the requirement to periodically check the integrity of software.

O.Security roles is provided by **FMT_SMR.2 (Restrictions on security roles)** which covers the requirement that a set of security roles be maintained and that users be associated with those roles, and **FMT_SMF.1 (Specification of Management Functions)** which covers the requirement that and specific management functions are provided, like the management of users and permissions of access on the part of the users, and the administration of users authentication.

O.Validation of security function is provided by **FPT_AMT.1 (Abstract machine testing)** which covers the requirement to ensure that security-relevant hardware and firmware are functioning correctly and **FPT_TST_CIMC.2 (Software/firmware integrity test)** which covers the requirement to ensure that security-relevant software is functioning correctly.

O.Trusted Path is provided by **FTP_TRP.1 (Trusted path)** which covers the requirement that a trusted path between the user and the system be provided.

Security Objectives for the TOE and Environment

O.Configuration Management is provided by **FMT_MOF.1 (Management of security functions behavior) (iterations 1 and 2)** which covers the requirement that only authorized users can change the configuration of the system. **FMT_MOF_CIMC.3 (Extended certificate profile management)** covers the requirement that Administrators be able to control the types of information that are included in generated certificates. **FMT_MOF_CIMC.5 (Extended certificate revocation list profile management)** covers the requirement that Administrators be able to control to the types of information that are included in generated certificate revocation lists. **FMT_MOF_CIMC.6 (OCSP Profile Management)** covers the requirement that Administrators be able to control to the types of information that are included in generated OCSP responses. **O.Configuration Management** is supported by **AGD_ADM.1 (Administrator Guidance)** which covers the requirement that Administrators be provided with documentation describing the configuration management features of the TOE and by **A.Competent Administrators, Operators, Officers and Auditors** and **A.CPS** which ensure that Administrators are competent and are familiar with the CPS under which the TOE is to be operated. **O.Configuration Management** is also supported by **ACM_AUT.1 (Partial CM automation)**, **ACM_CAP.4 (Generation support and acceptance procedures)**, and **ACM_SCP.2 (Problem tracking CM coverage)** which ensure that a configuration management system is implemented and used.

O.Data import/export is provided by **FDP_UCT.1 (Basic data exchange confidentiality) (iterations 1 and 2)** and **FPT_ITC.1 (Inter-TSF confidentiality during transmission) (iterations 1 and 2)** which cover the requirement that data other than private and secret keys be protected when they are transmitted and from the CIMC. **FDP_ETC_CIMC.5 (Extended user private and secret key export)** and **FMT_MTD_CIMC.7 (Extended TSF private and secret key export)** cover the requirement that private and secret keys be protected when they are transmitted to and from the TOE.

O.Detect modifications of firmware, software, and backup data is provided by **FPT_TST_CIMC.2 (Software/firmware integrity test)** which covers the requirement that modifications to software or firmware be detected and **FDP_CIMC_BKP.2 (Extended CIMC backup and recovery)** which covers the requirement that modifications to backup data be detected. Since **FPT_TST_CIMC.2** and **FDP_CIMC_BKP.2** make use of digital signatures, keyed hashes, or authentication codes to detect modifications, **FMT_MTD_CIMC.4 (TSF private key confidentiality protection)** and **FMT_MTD_CIMC.5 (TSF secret key confidentiality protection)** are necessary to ensure that an attacker who has modified firmware, software, or backup data can not prevent detection of the modification by computing a new digital signature, keyed hash, or authentication code.



O.Individual accountability and audit records is provided by a combination of requirements. **FIA_UID.1 (Timing of identification) (iterations 1 and 2)** covers the requirement that users be identified before performing any security-relevant operations. **FAU_GEN.1 (Audit data generation) (iterations 1 and 2)** and **FAU_SEL.1 (Selective audit) (iterations 1 and 2)** cover the requirement that security-relevant events be audited while **FAU_GEN.2 (User identity association) (iterations 1 and 2)** and **FPT_STM.1 (Reliable time stamps) (iterations 1 and 2)** cover the requirement that the date and time of audited events are recorded in the audit records along with the identities of the entities responsible for the actions. **FMT_MTD.1 (Management of TSF data)** covers the requirement that audit data be available for review by ensuring that users, other than Auditors, can not delete audit logs. Finally, **FAU_SAR.1 (Audit review)** and **FAU_SAR.3 (Selectable audit review)** cover the requirement that the audit records are made available for review so that individuals can be held accountable for their actions.

O.Integrity protection of user data and software is provided by **FDP_ITT.1 (Basic internal transfer protection) (iterations 1 and 3)** and **FDP_SDI_CIMC.3 (Stored public key integrity monitoring and action)** which cover the requirement that user data be protected and **FPT_TST_CIMC.2 (Software/firmware integrity test)** and **FPT_TST_CIMC.3 (Software/firmware load test)** which cover the requirement that software and firmware be protected. Since data and software are protected using cryptography, **FMT_MTD_CIMC.4 (TSF private key confidentiality protection)** and **FMT_MTD_CIMC.5 (TSF secret key confidentiality protection)** are required to protect the confidentiality of the private and secret keys used to protect the data and software.

O.Limitation of administrative access is provided by **FDP_ACC.1 (Subset access control) (iterations 1 and 2)**, **FDP_ACF.1 (Security attribute based access control) (iterations 1 and 2)**, **FIA_UAU.1 (Timing of authentication) (iterations 1 and 2)**, and **FIA_UID.1 (Timing of identification) (iterations 1 and 2)**. **FIA_UAU.1 (Timing of authentication) (iterations 1 and 2)** and **FIA_UID.1 (Timing of identification) (iterations 1 and 2)** ensure that Administrators, Operators, Officers, and Auditors can not perform any security-relevant operations until they have been identified and authenticated and **FDP_ACC.1 (Subset access control) (iterations 1 and 2)** and **FDP_ACF.1 (Security attribute based access control) (iterations 1 and 2)** ensure that Administrators, Operators, Officers, and Auditors can only perform those operations necessary to perform their jobs. **FPT_RVM.1 Non-bypassability of the TSP (iteration 2)** ensure that Administrators, Operators, Officers, and Auditors can not perform operations that they are not authorized to perform by bypassing the TSP enforcement functions.

O.Maintain user attributes is provided by **FIA_ATD.1 (User attribute definition)** and **FIA_USB.1 (User-subject binding) (iterations 1 and 2)** which cover the requirement to maintain a set of security attributes associated with individual users and/or subjects acting on users' behalves. **FMT_MSA.1 (Management of security attributes)** ensures that only authorized users can modify security attributes. **FMT_SMF.1 (Specification of Management Functions)** ensures that and specific management functions are provided, like the management of users and permissions of access on the part of the users, and the administration of users authentication.

O.Manage behavior of security functions is provided by **FMT_MOF.1 (Management of security functions behavior) (iterations 1 and 2)** which covers the requirement that authorized users be able to configure, operate, and maintain the security mechanisms. **FMT_SMF.1 (Specification of Management Functions)** ensures that and specific management functions are provided, like the management of users and

permissions of access on the part of the users, and the administration of users authentication.

O.Object and data recovery free from malicious code is provided by **FPT_TST_CIMC.2 (Software/firmware integrity test)** and **FPT_TST_CIMC.3 (Software/firmware load test)** which cover the requirement that the recovered state is free from malicious code. **FDP_CIMC_BKP.1 (CIMC backup and recovery)**, and **FDP_CIMC_BKP.2 (Extended CIMC backup and recovery)** cover the requirement to be able to recover to a viable state.

O.Procedures for preventing malicious code is provided by **FPT_TST_CIMC.2 (Software/firmware integrity test)** which ensures that only signed code can be executed and **AGD_ADM.1 (Administrator Guidance)**, **AGD_USR.1 (User Guidance)** and **A.Malicious Code Not Signed** which ensure that those who are capable of signing code do not to sign malicious code. It is also supported by **FDP_ACF_CIMC.2 (User private key confidentiality protection)**, **FDP_ACF_CIMC.3 (User secret key confidentiality protection)**, **FCS_CKM.4 (Cryptographic key destruction)** and **FCS_CKM_CIMC.5 (CIMC private and secret key zeroization)** which ensure that an untrusted entity can not use a trusted entity's key to sign malicious code.

O.Protect stored audit records is provided by **FAU_STG.1 (Protected audit trail storage) (iterations 1 and 2)** which covers the requirement that audit records be protected against modification or unauthorized deletion and **FMT_MTD.1 (Management of TSF data)** which covers the requirement that audit records be protected from unauthorized access. **FPT_CIMC_TSP.1 (Audit log signing event)** is required so that modifications to the audit logs can be detected. **A.Physical Protection** is also required in order to protect the audit records from unauthorized physical modification. **FPT_ACC.1 (Access Control)** is also required in order to protect the audit records from unauthorized modification providing from any external program with access to the audit database.

O.Protect user and TSF data during internal transfer is provided by **FDP_ITT.1 (Basic internal transfer protection) (iterations 1-4)** which covers the requirement that user data be protected during internal transfer and **FPT_ITT.1 (Basic internal TSF data transfer protection) (iterations 1-4)** which covers the requirement that TSF data be protected during internal transfer.

O.Require inspection for downloads is provided by **FPT_TST_CIMC.3 (Software/firmware load test)** which covers the requirement that downloaded software can not be loaded until it has been signed and by **AGD_ADM.1 (Administrator Guidance)**, **AGD_USR.1 (User Guidance)**, and **A.Malicious Code Not Signed** which ensure that those who are capable of signing code do not to sign malicious code.

O.Respond to possible loss of stored audit records is provided by **FAU_STG.4 (Prevention of audit data loss) (iterations 1 and 2)** which covers the requirement that no auditable events, except those taken by the Auditor, can be performed when audit trail storage is full.

O.Restrict actions before authentication is provided by **FIA_UAU.1 (Timing of authentication) (iterations 1 and 2)** which covers the requirement that no security-relevant actions are performed on behalf of a user until that user has been authenticated.

O.Security-relevant configuration management is provided by **FMT_MSA.3 (Static attribute initialisation)** and **FMT_MSA.2 (Secure security attributes)** which cover the requirement that security attributes have secure values. **FMT_MOF.1 (Management of**



security functions behavior) (iterations 1 and 2) ensures that security-relevant configuration data can only be modified by those who are authorized to do so. **O.Security-relevant configuration management** is also supported by **AGD_ADM.1 (Administrator Guidance)** which covers the requirement that Administrators be provided with documentation describing the configuration management features of the TOE and by **A.Competent Administrators, Operators, Officers and Auditors** and **A.CPS** which ensure that Administrators are competent and are familiar with the CPS under which the TOE is to be operated.

O.Time stamps is provided by **FPT_STM.1 (Reliable time stamps) (iterations 1 and 2)** which covers the requirement that the time stamps be reliable, and by **A.NTP Client** which ensures this reliability by means the guarantee that all the hosts included in the TOE have installed an NTP client that synchronises the system clock with a reliable clock that obtains the Coordinated Universal Time from a reliable source.

O.User authorization management is provided by **FMT_MSA.1 (Management of security attributes)** which covers the requirement that Administrators manage and update user's security attributes. **O.User authorization management** is also supported by **AGD_ADM.1 (Administrator Guidance)** which covers the requirement that Administrators be provided with documentation describing the user authorization management features of the TOE and by **A.Competent Administrators, Operators, Officers and Auditors** and **A.CPS** which ensure that Administrators are competent and are familiar with the CPS under which the TOE is to be operated.

O.React to detected attacks is provided by **FCS_CKM.4 (Cryptographic key destruction)** and **FCS_CKM_CIMC.5 (CIMC private and secret key zeroization)** which cover the requirement that the user who detected the attack be able to destroy any plaintext keys within the TOE in order to prevent the attacker from obtaining copies of these keys. **FIA_AFL.1 (Authentication failure handling)** covers the requirement that the TSF respond to detected attacks (in the form of repeated authentication attempts) by taking actions to prevent the attacker from successfully authenticating him/herself. In the case that an attack is detected by an Administrator, Auditor, Officer, or Operator.

8.2.3 Rationale for operations of Security Requirements

This section contains justifications of some operations (assignments, selections, ...) that has been applied to security requirements of the TOE or of the environment.

8.2.3.1 Rationale for operations of Security Requirements applied to the TOE

8.2.3.1.1 FIA_UAU.1.1

In this requirement the following actions have been included as events that are not security relevant:

- Indication of the authentication mode

The following user authentication modes can be indicated:

- Certificate (proof of possession using the private key related to the certificate that will be presented)

- Password
- Recovery password

Because the introduction of the indication of one of these authentication modes does not modify any security parameter of the system, but only it is an input parameter for the FIA_UIDAUT function (depending on this parameter the application will request the correct type of authentication data), then this event is not security relevant.

- Introduction of the authentication data

These data can consist in the following type of information: password or the proof of possession generated using the private key of the indicated certificate. The introduction of these data does not modify any security parameter of the system, but only it is an input parameter for the FIA_UIDAUT function. For security reasons, characters introduced in the password field are presented on screen as asteriks, and the proof of possession is generated inside the smartcard that contains the private key related to the presented certificate. Consequently, this event is not security relevant.

- Cancel the login procedure

The cancellation of the login procedure returns the state of the application to the previous state, and it eliminates any information introduced during the login process. Consequently, this event is not security relevant.

8.2.3.1.2 FIA_UID.1.1

In this requirement the following actions have been included as events that are not security relevant:

- Indication of the identification mode

The following user identification modes can be indicated:

- Certificate (presented in an authentication token, as an smart card)
- Username
- Recovery password

Because the introduction of the indication of one of these identification modes does not modify any security parameter of the system, but only it is an input parameter for the FIA_UIDAUT function (depending on this parameter the application will request the correct type of identification data), then this event is not security relevant.

- Introduction of the identification data

These data can consist in the following type of information: username or the indicated certificate. The introduction of these data does not modify any security parameter of the system, but only it is an input parameter for the FIA_UIDAUT function. Both the username and public key certificate can be considered not sensitive data in the system. Consequently, this event is not security relevant.

- Cancel the login procedure



The cancellation of the login procedure returns the state of the application to the previous state, and it eliminates any information introduced during the login process. Consequently, this event is not security relevant.

8.2.3.1.3 FDP_SDI_CIMC.3.2

In this requirement the following action has been included as event that is generated if the verification carried out to protect a public key fails: Generation of a report and forbid the use of the public key.

This action is consistent with the maintenance of the security, because:

- A report is generated and this guarantees that the system can monitor this security-relevant event in order it could be reviewed by an auditor for identifying the causes of the verification failure.
- The system forbids the use of the public key, and this guarantees that the integrity protection of these data is preserved.

8.2.3.1.4 FAU_STG.1.1

The TSF protects the stored audit records from unauthorized deletion because the KTS does not have any functionality to delete records from the audit database. From the KeyOne applications, it is not possible to delete any registry from any database managed by these applications.

8.2.3.2 Rationale for operations of Security Requirements applied to the environment

8.2.3.2.1 FIA_UAU.1.1, FIA_UID.1.1

In these requirements the following action has been included as event that is not security relevant: Request for username and password.

The introduction of these data does not modify any security parameter of the system, but only they are input parameters for the identification and authentication function. For security reasons, characters introduced in the password field are presented on screen as hidden characters, and the username can be considered not sensitive data in the system. Consequently, this event is not security relevant.

8.2.3.2.2 FPT_TST_CIMC.2.2

In this requirement the following action has been included in the assignment operation: *report the test failure*.

This completion is consistent with maintenance of security because the inclusion of the action "report the test failure" in this requirement maintains the satisfaction of the the following security objectives by the FPT_TST_CIMC.2.2 requirement:

- O.Detect modifications of firmware, software, and backup data. The FPT_TST_CIMC.2.2 covers the requirement that modification to software or firmware be detected. This detection is possible by means the report that is generated after the test failure at power-up or on-demand.

- O.Integrity protection of user data and software. The FPT_TST_CIMC.2.2 covers the requirement that integrity of software and firmware be protected by means the detection of integrity errors at power-up and on-demand. This integrity protection is provided by security mechanisms that report the test failure after a lack of integrity.
- O.Object and data recovery free from malicious code. The FPT_TST_CIMC.2.2 requirement assures that the system stays free from malicious code since this requirement guarantees that any intrusion to the system is reported indicating a test failure.
- O.Periodically check integrity. The FPT_TST_CIMC.2.2 covers the requirement to periodically check the integrity of software (at power-up or on-demand). This check generates the evidence of a report indicating the test failure.
- O.Procedures for preventing malicious code. The FPT_TST_CIMC.2.2 ensures that only authenticated code (by means error detection code, authentication code, keyed hash or digital signature) can be executed. A report indicating a test failure, after power-up or on-demand, guarantees that the system is prevented against malicious code.
- O.Validation of security function. The FPT_TST_CIMC.2.2 ensures that only authenticated software and firmware is functioning, and therefore that they work correctly through features and procedures. The generation of a report indicating a test failure contributes guarantees in the maintenance of this security objective.

8.2.3.2.3 FPT_TST_CIMC.3.2

In this requirement the following action has been included in the assignment operation: *does not allow the execution of the component where the test has failed.*

This completion is consistent with maintenance of security because the inclusion of the action "does not allow the execution of the component where the test has failed" in this requirement maintains the satisfaction of the the following security objectives by the FPT_TST_CIMC.3.2 requirement:

- O.Integrity protection of user data and software. The FPT_TST_CIMC.3.2 covers the requirement that integrity of software and firmware be protected by means the detection of integrity errors at power-up and on-demand. This integrity protection is provided by security mechanisms that does not allow the execution of a software or firmware that is externally loaded into the system, where the test has failed.
- O.Object and data recovery free from malicious code. The FPT_TST_CIMC.3.2 requirement contributes in which the system remains free from malicious code since this requirement guarantees that no component (software or firmware) is executed if the integrity test has failed, when this component has been tried to load in the system.
- O.Periodically check integrity. The FPT_TST_CIMC.3.2 requirement contributes in the periodical check of the integrity of software, when a component (software or firmware) has been tried to load in the system, because if the test failed, then the system does not allow the execution of this component.



- O.Require inspection for downloads. The FPT_TST_CIMC.3.2 requirement contributes in the inspection for downloads, because this requirement guarantees that no component (software or firmware) is executed if the integrity test has failed, when this component has been tried to load in the system.

8.3 Internal Consistency and Mutual Support

This section demonstrates that the stated security requirements together form a mutually supportive and internally consistent whole. Internal consistency is demonstrated in an analysis of dependencies. Mutual support is shown through consideration of the interactions between and among the SFRs.

8.3.1 Rationale that Dependencies are Satisfied

The selected security requirements include related dependencies, both direct and indirect. The indirect dependencies are those required by the direct dependencies. All of these dependencies must be met or their exclusion justified.

8.3.1.1 Security Functional Requirements Dependencies

The following table provide a summary of the security functional requirements dependency analysis for this Security Target.

Component	Dependencies	Which is:
FAU_GEN.1 Audit data generation	FPT_STM.1 Reliable time stamps	Included
FAU_GEN.2 User identity association	FAU_GEN.1 Audit data generation	Included
	FIA_UID.1 Timing of identification	Included
FAU_SAR.1 Audit review	FAU_GEN.1 Audit data generation	Included
FAU_SAR.3 Selectable audit review	FAU_SAR.1 Audit review	Included
FAU_SEL.1 Selective Audit	FAU_GEN.1 Audit data generation	Included
	FMT_MTD.1 Management of TSF data	Included
FAU_STG.1 Protected audit trail storage	FAU_GEN.1 Audit data generation	Included
FAU_STG.4 Prevention of audit data loss	FAU_STG.1 Protected audit trail storage	Included

FCO_NRO_CIMC.3 Enforced proof of origin and verification of origin	FIA_UID.1 Timing of identification	Included
FCO_NRO_CIMC.4 Advanced verification of origin	FCO_NRO_CIMC.3	Included
FCS_CKM.1 Cryptographic key generation	FCS_CKM.2 Cryptographic key distribution or FCS_COP.1 Cryptographic operation	FCS_COP.1 Included
	FCS_CKM.4 Cryptographic key destruction	Included
	FMT_MSA.2 Secure security attributes	Included
FCS_CKM.4 Cryptographic key destruction	FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation	FCS_CKM.1 Included
	FMT_MSA.2 Secure security attributes	Included
FCS_CKM_CIMC.5 CIMC private and secret key zeroization	FCS_CKM.4 Cryptographic key destruction	Included
	FDP_ACF.1 Security attribute based access control	Included
FCS_COP.1 Cryptographic operation	FCS_CKM.4 Cryptographic key destruction	Included
	FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation	FCS_CKM.1 Included
	FMT_MSA.2 Secure security attributes	Included
FDP_ACC.1 Subset access control	FDP_ACF.1 Security attribute based access control	Included
FDP_ACF.1 Security attribute based access control	FDP_ACC.1 Subset access control	Included
	FMT_MSA.3 Static attribute initialization	Included
FDP_ACF_CIMC.2 User private key confidentiality protection	None	
FDP_ACF_CIMC.3 User secret key confidentiality protection	None	
FDP_CIMC_BKP.1 CIMC backup and recovery	FMT_MOF.1 Management of security functions behavior	Included
FDP_CIMC_BKP.2 Extended CIMC backup and recovery	FDP_CIMC_BKP.1 CIMC backup and recovery	Included

FDP_CIMC_CER.1 Certificate Generation	None	
FDP_CIMC_CRL.1 Certificate revocation list validation	None	
FDP_CIMC_CSE.1 Certificate status export	None	
FDP_CIMC_OCSP.1 OCSP basic response validation	None	
FDP_ETC_CIMC.5 Extended user private and secret key export	None	
FDP_ITT.1 Basic internal transfer protection	FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control	FDP_ACC.1 Included
FDP_SDI_CIMC.3 Stored public key integrity monitoring and action	None	
FDP_UCT.1 Basic data exchange confidentiality	FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control	Included
	FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path	FTP_TRP.1 Included
FIA_AFL.1 Authentication failure handling	FIA_UAU.1 Timing of authentication	Included
FIA_ATD.1 User attribute definition	None	
FIA_UAU.1 Timing of authentication	FIA_UID.1 Timing of identification	Included
FIA_UID.1 Timing of Identification	None	
FIA_USB.1 User-subject binding	FIA_ATD.1 User attribute definition	Included
FMT_MOF.1 Management of security functions behavior	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
	FMT_SMF.1 Specification of management functions	Included
FMT_MOF_CIMC.3 Extended certificate profile management	FMT_MOF.1 Management of security functions behavior	Included
	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
FMT_MOF_CIMC.5 Extended certificate	FMT_MOF.1 Management of security functions behavior	Included

revocation list profile management	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
FMT_MOF_CIMC.6 OCSP profile management	FMT_MOF.1 Management of security functions behavior	Included
	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
FMT_MSA.1 Management of security attributes	FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control	Included
	FMT_SMF.1 Specification of management functions	Included
	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
FMT_MSA.2 Secure security attributes	ADV_SPM.1 Informal TOE security policy model	Included
	FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control	FDP_ACC.1 Included
	FMT_MSA.1 Management of security attributes	Included
	FMT_SMR.1 Security Roles	Included (hierarchical to FMT_SMR.2)
FMT_MSA.3 Static attribute initialization	FMT_MSA.1 Management of security attributes	Included
	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
FMT_MTD.1 Management of TSF data	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
	FMT_SMF.1 Specification of management functions	Included
FMT_MTD_CIMC.4 TSF private key confidentiality protection	None	
FMT_MTD_CIMC.5 TSF secret key confidentiality protection	None	
FMT_MTD_CIMC.6 TSF private and secret key export	None	

FMT_MTD_CIMC.7 Extended TSF private and secret key export	FMT_MTD_CIMC.6	Included
FMT_SMR.2 Restrictions on security roles	FIA_UID.1 Timing of identification	Included
FPT_AMT.1 Abstract machine testing	None	
FPT_CIMC_TSP.1 Audit log signing event	FAU_GEN.1 Audit data generation	Included
	FMT_MOF.1 Management of security functions behavior	Included
FPT_ITC.1 Inter-TSF confidentiality during transmission	None	
FPT_IIT.1 Basic internal TSF data transfer protection	None	
FPT_STM.1 Reliable time stamps	None	
FPT_TST_CIMC.2 Software/firmware integrity test	FPT_AMT.1 Abstract machine testing	Included
FPT_TST_CIMC.3 Software/firmware load test	FPT_AMT.1 Abstract Machine Testing	Included
FPT_TRP.1 Trusted path	None	

Table 8-8. Summary of Security Functional Requirements Dependencies for Security Level 3

8.3.1.2 Security Assurance Requirements Dependencies

The following table provide a summary of the security assurance requirements dependency analysis for Security Level 3, with additional Security Assurance requirements to achieve a complete CC EAL4 Level.

<i>Component</i>	<i>Depends On:</i>	<i>Which is:</i>
ACM_AUT.1	ACM_CAP.3	Included (hierarchical to ACM_CAP.4)
	ALC_DVS.1	included

ACM_CAP.4	ALC_DVS.1	included
	ACM_CAP.3	included (hierarchical to ACM_CAP.4)
ACM_SCP.2	(indirect) ALC_DVS.1	included
	ACM_CAP.3	included (hierarchical to ACM_CAP.4)
ADO_DEL.2	(indirect) ALC_DVS.1	included
	AGD_ADM.1	included
	(indirect) ADV_FSP.1	included (hierarchical to ADV_FSP.2)
ADO_IGS.1	(indirect) ADV_RCR.1	included
	ADV_RCR.1	included
	ADV_FSP.1	included (hierarchical to ADV_FSP.2)
ADV_FSP.2	ADV_RCR.1	included
ADV_HLD.2	ADV_LLD.1	included
	ADV_RCR.1	included
ADV_IMP.1	ALC_TAT.1	included
	(indirect) ADV_FSP.1	Included (hierarchical to ADV_FSP.2)
	(indirect) ADV_HLD.2	included
	ADV_HLD.2	included
	ADV_RCR.1	included
ADV_LLD.1	(indirect) ADV_FSP.1	Included (hierarchical to ADV_FSP.2)
	no dependencies	not applicable
	ADV_FSP.1	Included (hierarchical to ADV_FSP.2)
ADV_RCR.1	(indirect) ADV_RCR.1	included
ADV_SPM.1	ADV_FSP.1	Included (hierarchical to ADV_FSP.2)
	(indirect) ADV_RCR.1	included
AGD_ADM.1	ADV_FSP.1	Included (hierarchical to ADV_FSP.2)



	(indirect) ADV_RCR.1	included
AGD_USR.1	No dependencies	Not applicable
	No dependencies	Not applicable
ALC_DVS.1	No dependencies	Not applicable
ALC_FLR.2	ADV_IMP.1	Included
ALC_LCD.1	(indirect) ADV_FSP.1	Included (hierarchical to ADV_FSP.2)
ALC_TAT.1	(indirect) ADV_HLD.2	included
	(indirect) ADV_LLD.1	included
	(indirect) ADV_RCR.1	included
	ADV_FSP.1	included (hierarchical to ADV_FSP.2)
	ATE_FUN.1	included
ATE_COV.2	(indirect) ADV_RCR.1	included
	ADV_HLD.1	included (hierarchical to ADV_HLD.2)
	ATE_FUN.1	included
ATE_DPT.1	(indirect) ADV_FSP.1	included (hierarchical to ADV_FSP.2)
	(indirect) ADV_RCR.1	included
	No dependencies	Not applicable
	ADV_FSP.1	included (hierarchical to ADV_FSP.2)
ATE_FUN.1	AGD_ADM.1	included
ATE_IND.2	AGD_USR.1	included
	ATE_FUN.1	included
	(indirect) ADV_RCR.1	included
	ADO_IGS.1	included
	ADV_FSP.1	included (hierarchical to ADV_FSP.2)
AVA_MSU.2	AGD_ADM.1	included
	AGD_USR.1	included
	(indirect) ADV_RCR.1	included

	ADV_FSP.1	included (hierarchical to ADV_FSP.2)
	ADV_HLD.1	included (hierarchical to ADV_HLD.2)
AVA_SOF.1	(indirect) ADV_RCR.1	included
	ADV_FSP.1	included (hierarchical to ADV_FSP.2)
	ADV_HLD.2	included
AVA_VLA.2	ADV_IMP.1	included
	ADV_LLD.1	included
	AGD_ADM.1	included
	AGD_USR.1	included
	(indirect) ADV_RCR.1	included
	(indirect) ALC_TAT.1	included

Table 8-9. Summary of Security Assurance Requirements Dependencies for Security Level 3

8.3.2 Rationale that Requirements are Mutually Supportive

The requirements represented in this ST were developed from a variety of sources. The security work mutually so that each SFR is protected against bypassing, tampering, deactivation, and detection attacks by other SFRs.

Bypass

Prevention of bypass is derived as described below:

FIA_UID.1 and FIA_UAU.1 support other functions' allowing user access to data by limiting the actions the user can take prior to identification and authentication.

The management functions, including FMT_MOF.1, FMT_MSA.1, and FMT_MTD.1 support all other SFRs by restricting the ability to change certain management functions to certain specified roles, thus ensuring that other users cannot circumvent these SFRs.

FPT_TST_CIMC.2 provides for integrity testing to ensure that selected security functions are operational, thus checking for bypass.

FMT_MSA.2 and FMT_MSA.3 limit the acceptable values for secure data, thus providing protection from bypass to those SFRs dependent on that data.

Tamper



Prevention of tamper is derived as described below:

FAU_STG.1 protects the integrity of the audit trail.

FCS_CKM.1 and FCS_COP.1 provide for the secure generation and handling of keys, and therefore support those SFRs that may rely on the use of those keys.

FIA_UID.1 and FIA_UAU.1 support other functions allowing user access to data by limiting the actions the user can take prior to identification and authentication.

The management functions, including FMT_MOF.1, FMT_MSA.1, and FMT_MTD.1 support all other SFRs by restricting the ability to change certain management functions to certain specified roles, thus ensuring that other users cannot circumvent these SFRs.

FPT_TST_CIMC.2 provides for integrity testing to ensure that selected security functions are operational, thus checking for tampering.

FDP_ETC_CIMC.5) prevent modification errors during export of secret and/or private keys.

FIA_AFL.1 supports all SFRs dealing with authentication by limiting the number of entry attempts, and then mandating an appropriate action to protect the TOE if too many attempts have been made.

FMT_MSA.2 and FMT_MSA.3 limit the acceptable values for secure data, thus providing protection from tampering to those SFRs dependent on that data.

Deactivation

Prevention of deactivation is derived as described below:

The access control SFP detailed in FDP_ACF.1 along with the other SFRs dealing with access control, provide for rigorous control of allowed data manipulations and thus prevent unauthorized deactivation.

The management functions, including FMT_MOF.1, FMT_MSA.1, and FMT_MTD.1, support all other SFRs by restricting the ability to change certain management functions to certain specified roles, thus ensuring that other users cannot circumvent these SFRs.

FPT_TST_CIMC.2 provides for integrity testing to ensure that selected security functions are operational, thus checking for tampering.

FMT_MSA.2 and FMT_MSA.3 limit the acceptable values for secure data, thus providing protection from deactivation to those SFRs dependent on that data.

Detection

Detection is derived as described below:

The security audit functions, including FAU_GEN.1, FAU_GEN.2, and FAU_SEL.1 provide for the generation of audit data that may be used to detect attempts to defeat specific SFRs or potential misconfiguration that could leave the TOE prone to attack.

FAU_SAR.1 and FAU_SAR.3, support the audit generation SFRs by providing the capability to selectively search the audit records.

FAU_STG.1, and FAU_STG.4 provide for the protection of the audit records.

The management functions, including FMT_MOF.1, FMT_MSA.1, and FMT_MTD.1, support all other SFRs by restricting the ability to change certain management functions to certain specified roles, thus ensuring that other users cannot circumvent these SFRs.

FMT_MSA.2 and FMT_MSA.3 limit the acceptable values for secure data, thus providing detection protection to those SFRs dependent on that data.

FMT_SMR.2 provides for the specification of multiple roles, thus supporting the other detection SFRs.

8.4 Rationale for Strength of Function

The TOE described in this ST is intended to operate in a range of environments, from benign to hostile.

Also, the users may be hostile. Therefore, the TOE requires cryptographic functions to provide for integrity, confidentiality, nondisclosure, and authentication. The authentication strength of function metrics provide for a basic level, and are currently within commercially available products. The cryptographic functions must be included in a cryptographic module that has been validated against FIPS 140-1 *Security Requirements for Cryptographic Modules*. The level required for the cryptographic module depends on the type and use of the key and the CIMC Security Level. The cryptographic module levels are specified in Tabla 6-5. Nivel FIPS 140-1 para módulo criptográfico validado. The increasing FIPS 140-1 level corresponding to the increased CIMC Security Level addresses the increased threats and potential for loss at the higher levels.

The security and assurance requirements specified at CIMC Security Level 3 are intended for environments where the risks and consequences of data disclosure and loss of data integrity are moderate. CIMC Level 3 requires additional integrity controls to ensure data is not modified. CIMC Security Level 3 includes mechanisms to protect against attacks by parties with physical access to the components and includes additional assurance requirements to ensure the CIMC is functioning securely. The EAL for Security Level is EAL3 augmented, as defines [CIMC].

This TOE (KeyOne system) has been designed for accomplishing with Common Criteria EAL4+, so and as it is declared in this Security Target, where users require a moderate to high level of security.



8.5 Assurance Requirements Rationale

8.5.1 Rationale for CIMC security level 3

CIMC is designed to meet Security Level 3 may be appropriate for environments where risks and consequences of data disclosure and loss of data integrity are moderate. Level 3 requires additional integrity controls to ensure data is not modified. A CIMC at Security Level 3 includes protections to protect against someone with physical access to the components and includes additional assurance requirements to ensure the CIMC is functioning securely.

The assurance level for this Security Level is EAL 3/EAL 4 augmented. Augmentation results from the selection of:

ACM_SCP.2 Problem tracking configuration management coverage

A vendor can be expected to apply configuration management to the items called out in ACM_SCP.2.

Specifically, since the product is security related, the tracking of security flaws is a very reasonable expectation and within the bounds of standard, best commercial practice.

ADO_DEL.2 Detection of modification

A vendor can be expected to use a signature or other method to ensure that the code has not been tampered with prior to installation. Since the product is security related, this type of precaution should be expected.

ADV_FSP.2 Fully defined external interfaces

It is not a difficult task to fully define all external interfaces to the product. Indeed, this is necessary to correctly develop the product for interaction with other products. This will provide the necessary detail for supporting both thorough testing of the TOE and the assessment of vulnerabilities.

ADV_IMP.1 Subset of the implementation of the TSF

This high a level of assurance requires that additional documentation regarding the implementation of the product be provided. It is through examination of this portion of the implementation that the product can be adequately evaluated with regard to the requirements.

ADV_LLD.1 Descriptive low-level design

This high a level of assurance requires that additional documentation regarding the design of the product be provided. It is through examination of this design that the product can be adequately evaluated with regard to the requirements.

ADV_SPM.1 Informal TOE security policy model

While the generation of a security policy does require security expertise, this can be performed by a consultant (if necessary) and does not otherwise impact the vendor's existing development process at this Security Level.

ALC_FLR.2 Flaw Report Procedures

EAL 3 and EAL 4 do not have the ALC_FLR component. It is within best commercial practices for a vendor of security products to have flaw reporting procedures covering:

- Addressing user reported problems
- Correcting flaws
- Notifying users and
- Revising procedures to reduce the potential for introducing new and/or additional flaws.

Specific procedures are not defined in the assurance requirement, therefore this should have minimal impact on vendors who have already implemented a flaw reporting program.

ALC_TAT.1 Well-defined development tools

It is important that very secure products be unambiguous.

AVA_MSU.2 Validation of analysis components

A security vendor implementing standard, best commercial practices will not be impacted by this component. AVA_MSU.2 requires that the vendor produce user and administrator documentation that is adequate for understanding the operating modes of the TOE and the required external security controls necessary for secure operation. The vendor is required to analyze this documentation for conformance to the requirements.

AVA_VLA.2 Independent vulnerability analysis

Penetration attacks are very likely given the threat model for this Security Level. As a result, it is important that some penetration analysis and testing be performed.

8.5.2 Rationale for EAL4

The assurance requirements defined for security level 3 of PP CIMC are very nearly from CC EAL4, so EAL4 has been selected to be the overall assurance for this TOE. Additional assurance requirements are rationalized below:

ACM_AUT.1 Partial CM automation

Automation in the configuration management system can help reduce the risk of human error or negligence.

ACM_CAP.4 Generation support and acceptance procedures

It is important that changes to the TOE be appropriately controlled. This requirement helps to ensure that when changes are made, they are appropriate and correctly applied to the resulting TOE.

ALC_LCD.1 Developer defined life-cycle model

It is important that changes to the TOE be appropriately controlled. This requirement helps to ensure that the development and maintained are appropriately controlled.

8.6 Razonamiento para los requisitos de seguridad extendidos propietarios

Esta Declaración de Seguridad especifica los siguientes dos tipos de requisitos de seguridad extendidos:

- Requisitos de seguridad extendidos CIMC

Estos requisitos están incluidos en la sección Requisitos Funcionales de Seguridad Extendidos CIMC, página 96, y éstos están justificados porque estos requisitos están incluidos en el Perfil de Protección [CIMC].

- Requisitos de seguridad extendidos propietarios

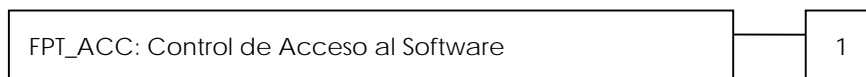
Estos requisitos son requisitos propietarios de esta Declaración de Seguridad. En la siguiente sección se describen.

8.6.1 Requisitos de seguridad extendidos propietarios

Control de Acceso (FPT_ACC)

Comportamiento de la familia

Esta familia define requisitos sobre el control de acceso a las herramientas y programas que pueden estar disponibles para el TOE



Nivel de componentes

En FPT_ACC.1, el TSF debe introducir requisitos necesarios para incluir un control de acceso a cualquier software que pueda estar disponible para el TOE.

Auditoría: FPT_ACC.1

No hay eventos auditable.

FPT_ACC.1 Control de acceso al software

Este componente requiere medidas de control de acceso a ser aplicadas a aquel software que pueda estar disponible para el TOE.

FPT_ACC.1.1

El entorno no debe tener instalado ningún [asignación: *componente software*] que pueda acceder a [asignación: *componente tecnológico*] usado por el TOE.

FPT_ACC.1.2



Si se utiliza [asignación: *un componente software*] que acceda a [asignación: *un componente tecnológica*], entonces este acceso debe estar controlado y supervisado por un [asignación: *rol*].

9 Bibliografía, Definiciones y Acrónimos

9.1 Bibliografía

Los siguientes documentos son referenciados en este documento:

<i>Referencia</i>	<i>Documento referenciado</i>
[CEN01a]	CEN/ISSS <i>Workshop on Electronic Signatures. CEN Hardware Security Modules for CSPs, CC Protection Profile, EESSI Area D2</i> , 2001.
[CEN01b]	CEN/ISSS <i>Workshop on Electronic Signatures. CEN/ISSS WS/E-Sign Workshop Agreement Group F, Security Requirements of Secure Signature Creation Devices (SSCD)</i> , 2001.
[CEN01c]	CEN/ISSS <i>Workshop on Electronic Signatures. Security Requirements for Trustworthy Systems Managing Certificates for Electronic Signatures</i> , June 2003.
[CIMC]	<i>Certificate Issuing and Management Components Family of Protection Profiles</i> , Version 1.0. October 31, 2001. National Security Agency (NSA).
[Eur99a]	European Community. <i>Algorithms and parameters for algorithms, list of algorithms and parameters eligible for electronic signatures, procedures as defined in the directive 1999/93/EC, article 9 on the "Electronic Signature Committee" in the Directive.</i> , 1999.
[Eur99b]	European Community. <i>Directive 1999/93/EC of the European Parliament and of the Council of 13 December 1999 on a community framework for electronic signatures</i> , 1999.
[FIP]	<i>FIPS 140-2 SECURITY REQUIREMENTS FOR CRYPTOGRAPHIC MODULES.</i>
[Ser97]	Service Central de la Sécurité des Systèmes d'Information. <i>Expression des Besoins et Identification des Objectifs de Sécurité</i> , 1.02 edition, 1997.



<i>Referencia</i>	<i>Documento referenciado</i>
[the99a]	the Common Criteria Project Sponsoring Organisations. <i>Common Criteria for Information Technology Security Evaluation Part 3: Security assurance requirements</i> , 2.2, January 2004.
[the99b]	the Common Criteria Project Sponsoring Organisations. <i>Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements</i> , 2.2, January 2004.
[the99c]	the Common Criteria Project Sponsoring Organisations. <i>Common Criteria for Information Technology Security Evaluation Part 1: Introduction and general model</i> , 2.2, January 2004.
[TS1]	ETSI TS 101 456, <i>Policy Requirements for Certification Authorities Issuing Qualified Certificates</i> .
[FUNCSPEC]	KeyOne 3.0 – Product Specification, Safelayer internal code: 6D6436D9
[ALGO]	ETSI SR 002 176 – Electronic Signatures and Infrastructures (ESI); Algorithms and Parameters for Secure Electronic Signatures
[X509]	X509v3: ITU-T Recommendation X.509 ISO/IEC International Standard 9594-8, Information Technology – Open Systems Interconnection – The Directory: Authentication Framework
[TS101862]	ETSI TS 101 862, Qualified Certificate Profile
[PKCS5]	PKCS #5: Password-Based Encryption Standard, RSA Laboratories
[RFC2560]	RFC 2560: Online Certificate Status Protocol - OCSP
[RFC3161]	RFC 3161: Time-Stamp Protocol (TSP)
[Eur03c]	<i>COMMISSION DECISION of 14 July 2003 on the publication of reference numbers of generally recognised standards for electronic signature products in accordance with Directive 1999/93/EC of the European Parliament and of the Council</i>
[CONFIGUIDE]	Configuration Guide – CC EAL4 Certification -, Safelayer internal code: 8ADC23DA
[NPKI]	NATO Public Key Infrastructure (NPKI) Certificate Policy

9.2 Definiciones

Datos de Activación: los valores de datos, a parte de claves, que se requieren para operar módulos criptográficos y que necesitan ser protegidos (por ejemplo, un PIN, una *passphrase*, o una porción de clave manualmente-guardada).

Firma electrónica avanzada: una firma electrónica que cumple los requisitos siguientes:

- está unívocamente asociada al firmante;
- es capaz de identificar al firmante;
- está creada usando mecanismos que el firmante puede mantener bajo su control exclusivo; y
- está asociada a los datos a los que hace referencia de manera que cualquier cambio en los datos es detectable;

Certificado de CA: Un certificado para una CA emitido por otra CA.

Punto de distribución de CRL: Una entrada del directorio u otra fuente de distribución para CRLs; un CRL distribuida a través de un punto de distribución de CRL puede contener entradas de revocación para sólo un subconjunto del conjunto completo de certificados emitidos por un CA o puede contener las entradas de revocación para múltiples CAs

Servicio de Diseminación de Certificados: Un servicio que difunde certificados a Titulares, y si el titular consiente, a las Partes de Confianza. Este servicio difunde también la política de las CA.s y información práctica a Titulares y Partes de Confianza.

Servicio de Generación de Certificados: Un servicio que crea y firma certificados basados en la identidad y otros atributos verificados por el servicio de registro.

Política de Certificado: Un conjunto denominado de reglas que indican la aplicabilidad de un certificado a una comunidad particular y/o clase de aplicación con requisitos comunes de seguridad.

Periodo de validación de un Certificado: El periodo de validez del certificado es el intervalo de tiempo durante que la CA garantiza que mantendrá información acerca del estado del certificado.

Certificado: una confirmación electrónica que asocia la firma-verificación de datos a una persona y confirma la identidad de esa persona.

Certificado: la clave pública de un usuario, junto con alguna otra información suministrada de modo inolvidable por cifrado con la clave privada de la autoridad de certificación que lo emitió.

Componete de Gestión de Emisión de Certificados (CIMC): Un Componete de Gestión de Emisión de Certificados consiste en el *hardware*, en el *software*, y en *firmware* que son responsables de realizar las funciones de un Sistema de Gestión de Emisión de Certificados. Un CIMC no incluye los controles de entorno (por ejemplo, la facilidad de control de acceso, la temperatura), las políticas y los procedimientos, los



controles del personal (por ejemplo, cheques de fondo y espacios libres de seguridad), y otros controles administrativos que completan un CIMS

Manifestación Práctica de Certificación: Una aseveración de las prácticas que una Autoridad Certificadora hace uso al emitir certificados.

Autoridad Certificadora (CA): Una autoridad de confianza por uno o más usuarios que crea y asigna certificados. Opcionalmente la autoridad certificadora puede crear claves.

Ruta de Certificación: Una cadena de múltiples certificados, comprendiendo un certificado del dueño de la clave pública (la entidad final) firmado por una CA, y por cero o más certificados adicionales de CAs firmados por otras CAs.

Proveedor de servicio de certificación: una entidad o una persona legal o natural que emite certificados o proporciona otros servicios relacionados a firmas electrónicas

Firma Digital: Datos añadidos a, o una transformación criptográfica (véa criptografía) de, una unidad de datos que permite a un recipiente de la unidad de datos demostrar la fuente y la integridad de la unidad de datos y proteger contra la falsificación por ejemplo por el recipiente

Firma Electrónica: Datos en formato electrónico que son adheridos a o lógicamente asociados con otros datos electrónicos y que sirve como un método de autenticación de esos datos

Producto de firma electrónica: El hardware o el software, o los componentes pertinentes del mismo, que se pretenden utilizar por un proveedor de servicio de certificación para la suministración de servicios de firma-electrónico o se pretenden utilizar para la creación o verificación de firmas electrónicas

Entidad final: Un sujeto de certificado que utiliza su clave pública para otros propósitos que firmar certificados

Función de Hash: : Una función que mapea cadenas de bits a cadenas de bits de longitud fija a son las siguientes dos propiedades:

- no es computacionalmente factible encontrar para una salida dada una entrada que mapea esa salida.
- no es computacionalmente factible encontrar dada una entrada una segunda entrada que mapee la misma salida.

Calificador de Política: Información dependiendo de la política que acompaña un identificador de la política de certificado en una certificado X. 509.

Clave privada: Aquella clave de un par de claves asimétricas de la entidad que debe sólo sea utilizada por esa entidad.

Public key: Aquella clave de un par de claves asimétricas de la entidad que podrá ser pública.

Certificado cualificado: un certificado que cumple los requisitos impuestos en Anexo I de la Directiva y es proporcionado por un proveedor del servicio de la certificación que cumple los requisitos impuestos en el Anexo II de la Directiva;

Firma electrónica cualificada: una firma electrónica avanzada que se basa en un certificado cualificado y que son creados por un dispositivo de creación de firma segura (Nota: la Definición de 5,1 firma tomada de la Directiva)

Servicio de Registro: Un servicio que verifica la identidad y, si es aplicable, algún atributo específico de un Titular. Los resultados de este servicio son pasados al Servicio de la Generación del Certificado.

Autoridad de Registro (RA): Una entidad que es responsable de la identificación y autenticación de sujetos de los certificados, pero que no firma ni publica certificados (es decir, un RA se delega ciertas tareas a favor de una CA).

Partes de Confianza: Un usuario o el agente que depende de los datos de un certificado para hacer las decisiones.

Servicio de Gestión de Revocación: Un servicio que procesa los pedidos y los informes relativos a la revocación para determinar la acción necesaria a tomar. Los resultados de este servicio se distribuyen a través el Servicio de Estado de Revocación.

Servicio de Estado de Revocación: Un servicio que proporciona información sobre el estado de revocación de los certificados a las partes de confianza. Este servicio puede ser un servicio de tiempo real o puede basarse en información de estado de revocación que se actualiza en intervalos regulares.

Dispositivo de creación de firma segura: un dispositivo de creación de firma que cumple con los requisitos impuso en el Anexo III de la Directiva

Política de Seguridad: El conjunto de reglas impuestas por la autoridad de seguridad que administra el uso y la provisión de servicios y facilidades de seguridad.

Certificado auto-firmado: Un certificado de una CA firmado por la propia CA.

Firmante: Una persona que tiene los datos de creación de firma y actúa en su propio beneficio o a favor de la persona o la entidad naturales o legales que él representa; Nota: el término signatario es utilizado a veces como un sinónimo

Datos de creación de firma: los datos únicos, tal como códigos o claves criptográficas privadas, que son utilizados por el firmante para crear una firma electrónica

Dispositivo de creación de firma: El *software* o el *hardware* configurado para implementar los datos de creación de firma.

Dispositivo de verificación de firma: El *software* o el *hardware* usados para implementar la firma-verificación de los datos.

Datos de verificación de firma: los datos, tal como códigos o claves criptográficas públicas, que son utilizados para verificar una firma electrónica

Servicio de Provisión de Dispositivo del Titular: Un servicio que prepara y proporciona un Dispositivo de Creación de Firma a Titulares.

Titular: Una entidad suscrita a un CSP para tener su clave pública e identidad certificadas en un certificado clave pública.



Servicio de Estampación de Tiempo: Un servicio que proporciona una asociación de confianza entre un dato y un instante particular en el tiempo, para establecer la evidencia segura que indica el instante de tiempo en que el dato existió.

Sistema de confianza: Un sistema de información o producto implementado como hardware y/o software que produce registros seguros y auténticos que están protegidos contra la modificación y asegura adicionalmente la seguridad técnica y criptográfica de los procesos sostenidos por él.

Acreditación voluntaria: Cualquier permiso estableciendo los derechos y las obligaciones específicas que se otorgan a las peticiones en lo que concierne al proveedor de servicios de certificación, por el organismo público o privado encargado de la elaboración, y de la supervisión de su cumplimiento, de tales derechos y obligaciones, donde el proveedor de servicio de certificación no permite ejercitar los derechos retenidos por el permiso hasta que se recibe la decisión de la organización.

Nota: El término "acreditación" es utilizado generalmente en otra manera, significando "la acreditación de las organizaciones de la certificación que realizan la evaluación de la conformidad de productos y/o servicios".

9.3 Acrónimos

Las siguientes abreviaturas se usan en este documento:

<i>Acrónimo</i>	<i>Significado</i>
ARL	Lista de Revocación de la Autoridad
CA	Autoridad de Certificación
CIMC	Componente de Gestión y Emisión de Certificados
CP	Política de Certificado
CRL	Certificate Revocation List
CSP	Proveedor de Servicio de Certificación
HSM	Módulo de Seguridad Hardware
HW	Hardware
I/O	Entrada/Salida
It	Información Tecnológica
LRA	Autoridad de Registro Ligera
NDCCP	protocolo Near Domain Cert-Status Coverage
NQC	Certificado No-Cualificado
NTP	Protocolo de Tiempo de Red

<i>Acrónimo</i>	<i>Significado</i>
OCSP	Protocolo de Estado de Certificado "en línea"
OS	Sistema Operativo
PKI	Infraestructura de clave pública
POP	Prueba de Posesión
PP	Perfil de Protección
QC	Certificado Cualificado
RA	Autoridad de Registro
SCD	Dispositivo de Creación de Firma
SF	Función de Seguridad
SSCD	Dispositivo de Creación de Firma Segura
ST	Objetivo de Seguridad
TOE	Objetivo de Evaluación
TSA	Autoridad de Estampación de Tiempo
TSP	Protocolo de Estampación de Tiempo
TST	<i>Token</i> de Sello de Tiempo
TSS	Servicio de Estampación de Tiempo
KTS	Sistema de Confianza KeyOne
VA	Autoridad de Validación

Consideraciones sobre el fichero de licencia

Para instalar los productos de Safelayer que se encuentran en el CD-Rom de distribución, se necesita un fichero de licencia (.slx). En el fichero de licencia se incluye un fichero de *cust* que permite la ejecución de aplicaciones de Safelayer.

El software de Safelayer accede al fichero de *cust* para verificar el cumplimiento de las licencias durante la ejecución de las aplicaciones. Este *cust* es instalado durante el procedimiento de puesta en marcha en una localización apropiada. Las aplicaciones de Safelayer no se pondrán en marcha si el fichero no existe o si los datos contenidos en este fichero no permiten la ejecución de la aplicación.

Cuando arrancan las aplicaciones de Safelayer, validan todos los scripts que el producto va a utilizar. Si durante este proceso se encuentra un script con una firma inválida o no reconocida, la aplicación muestra el error de integridad y lo para.

En el fichero de *cust* se incluyen certificados raíz necesarios para verificar las firmas de código. Normalmente se incluyen dos certificados raíz:

- El primero corresponde a la firma de código de Safelayer, necesaria para verificar el código contenido en el CD-Rom de la distribución. Este certificado se incluye inicialmente en el primer *cust* liberado por Safelayer.
- El segundo es individual para cada instalación. Este certificado permite verificar que el cliente ha firmado cambios que han sido realizados en scripts del producto KeyOne, previniendo que estos cambios sean ejecutados por personal no autorizado. Este certificado será incluido en el *cust* por Safelayer para firmar cualquier script.

Cada vez que una aplicación KeyOne arranca, se verifica la firma de todos los scripts. Si un script tiene una firma inválida, un mensaje de error para la aplicación que se está utilizando. Dependiendo del fichero de licencia, éste puede permitir la ejecución de scripts sin validar la firma asociada. Esto es posible si el fichero de licencia permite ejecutar scripts con el flag `--unsecure`.

Para cumplir las garantías de seguridad EAL4+ del producto, el fichero de licencia que se use no debe permitir la ejecución de scripts con este flag. La manera para comprobar si este flag está permitido es intentare ejecutar cualquier aplicación o script con este flag. Un ejemplo es añadir el flag `--unsecure` en el fichero de inicio del servidor de KeyOne CA (ejemplo: `start C:\Safelayer\KeyOne30\KeyOneHome\keyoneserver\keyoneserver.exe -configfile ".\online/start_server.ws" --unsecure`).

Si el fichero de licencia permite la ejecución de scripts en modo inseguro, entonces el siguiente informe de error aparecerá: "Warning! KeyOne Server is running in unsecure mode.Scripts signature is not verified in any way". En cualquier caso, un mensaje de error similar al siguiente para la aplicación que se está usando: "Invalid program arguments. ERROR - Scriptor Event_ForbiddenParameter. Parameter = -- unsecure".



SAFELAYER SECURE COMMUNICATIONS, S.A.

Edificio Valreality C/ Basauri, 17 Edificio B Pl. Baja Izq. Of. B 28023 Madrid (SPAIN) Tel.: +34 91 7080480 Fax: +34 91 3076652
Edif. World Trade Center (S-4), Moll de Barcelona S/N 08039 Barcelona (SPAIN) Tel.: +34 93 5088090 Fax: +34 93 5088091

WWW.SAFELAYER.COM