Declaración de seguridad de Enigmedia App SDK 1.10.4

Versión 1.2

05-11-2014

Enigmedia SL
CIF ESB75058503

C/Portuetxe 23A - Oficina 2-6
20018 Donostia (Gipuzkoa)

Telephone: +34 943046661

Índice de contenido

1 Introducción

1.1 Referencia de la declaración de seguridad

Titulo: Declaración de seguridad de Enigmedia App SDK 1.10.4

Versión: 1.2

Autor: Enigmedia S.L.

Fecha de publicación: 05-11-2014

1.2 Referencia del TOE

Nombre: Enigmedia App SDK

Versión: 1.10.4

Desarrollador: Enigmedia S.L.

1.3 Resumen del TOE

El TOE es una SDK para dispositivos móviles orientada a las comunicaciones seguras de VoIP (Voz sobre IP) con audio/vídeo. Como tal, el TOE proporciona servicios a aplicaciones Android, de tal manera que dichas aplicaciones incluirán en su interior al TOE (entre ellas la aplicación Enigmedia App).

Esta SDK consta de un conjunto de librerías software multiplataforma que implementan la siguiente funcionalidad de seguridad:

- Funcionalidad criptográfica asociada la gestión y cifrado de las comunicaciones VoIP haciendo uso del algoritmo KVC.
- Funcionalidad criptográfica asociada a la gestión y cifrado del fichero de configuración de cada instancia del TOE, haciendo uso del algoritmo AES-CBC con clave de 256 bits.
- Importación tanto de los datos para el fichero de configuración durante la primera fase de operación del TOE como de la clave KVC utilizada para cifrar cada comunicación VoIP.
- Mantenimiento interno del ID de usuario asociado al dispositivo donde el TOE opera.
- Mantenimiento de dos canales seguros de comunicación: uno con la centralita SIP (o servidor MediaProxy en su defecto) y otro con el servidor Web del Frontend.

Con esta funcionalidad de seguridad se garantiza la protección de los siguientes elementos:

- Las comunicaciones de señalización (registro en el sistema, inicio de llamada, negociación de codecs, ancho de banda, resolución de vídeo, y otros elementos propios del protocolo SIP).
- El envío de audio y vídeo propiamente dicho.
- Los canales de comunicación auxiliares (intercambio de agenda de contactos, mensajería).
- Almacenamiento de la configuración (información necesaria para la autenticación contra los servidores del sistema y la verificación de los mismos).
- Almacenamiento de agenda, mensajes y configuración

La interfaz gráfica (GUI) de las aplicaciones que incluyen al TOE no forman parte del mismo y por tanto está fuera de la evaluación, lo que implica que no hay ninguna garantía de seguridad asociada a ellas.

1.3.1 Tipo de TOE

SDK de Android que proporciona servicios software para comunicaciones cifradas de VoIP con audio/vídeo y mensajería de texto.

1.3.2 Uso del TOE

La funcionalidad de seguridad que ofrece el TOE y mencionada en los apartados anteriores no es accedida de forma directa por los usuarios finales de las aplicaciones Android. El uso del TOE, dada su naturaleza, se hace a través de la interfaz gráfica (GUI) que en cada caso se integre con el propio TOE a la hora de crear una App final.

Generalmente, estas Apps serán idénticas desde el punto de vista funcional, de tal manera que compartirán un “core” común (el propio TOE) y la única diferencia entre ellas será el interfaz gráfico (GUI) que en cada caso se adecuará a las necesidades.

El responsable del desarrollo de las Apps de Android que hagan uso del TOE será Enigmedia SL. Cuando un cliente desee crear su propia App que integre al TOE y que tenga su GUI personalizada, contactará con Enigmedia SL, la cual proporcionará la App final (TOE+GUI) al cliente el cuestión.

Por tanto, el único usuario del TOE será el desarrollador de aplicaciones que cree la GUI y la integre con el TOE.

1.3.3 Software y hardware requerido por el TOE para su operación

Dadas las características del TOE, éste precisa de componentes adicionales tanto software como hardware para completar su operación. A continuación se presenta una descripción del entorno de operación del TOE, la cual incluye los componentes necesarios mencionados:

Componentes locales

Se consideran componentes locales a aquellos que se ejecutan dentro del propio dispositivo móvil donde el TOE está instalado.

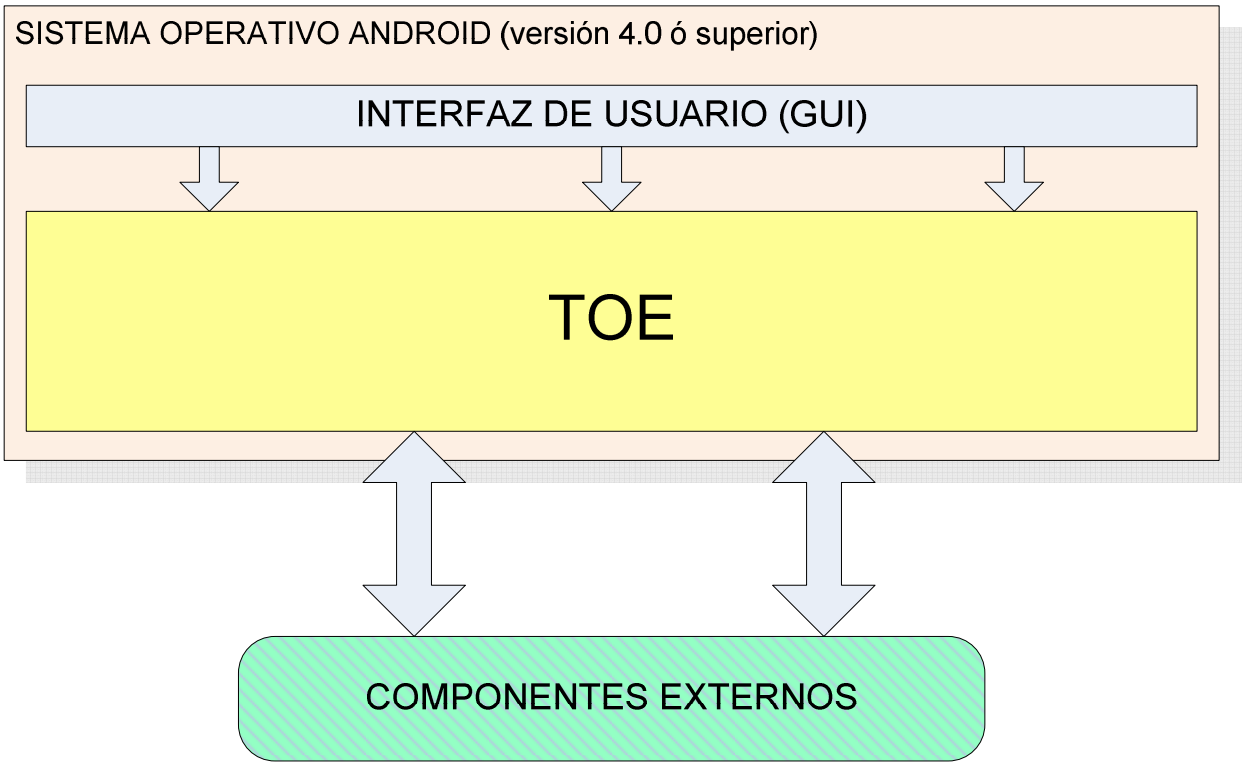

Figura 1: Componentes internos

Tal y como muestra la figura, los componentes internos son los siguientes:

- Sistema Operativo: El TOE se ejecuta sobre Android versión 4.0 ó superior (aunque existe una versión del TOE para iOS, la única versión evaluada es la que se despliega en Android). La versión de Android debe ser oficial y no encontrarse rooteada o modificada para ejecutar acciones no permitidas por defecto.
- GUI: El TOE precisa para su operación (por parte de usuarios finales del dispositivo móvil) de un interfaz de usuario, el cual será el encargado de interaccionar con las funcionalidades que ofrece el TOE.

Componentes externos

Se consideran componentes externos a aquellas entidades (hardware ó software) que ejecutan su funcionalidad fuera del propio dispositivo móvil donde el TOE está instalado.

Para establecer comunicaciones VoIP cifradas haciendo uso del TOE es necesario disponer, por un lado, de un segundo TOE instalado en otro dispositivo, y por otro lado, de un conjunto de servidores que proporcionan diferentes servicios. Estos servidores se pueden dividir en dos grupos: Frontend y Backend. La siguiente figura muestra esta división:

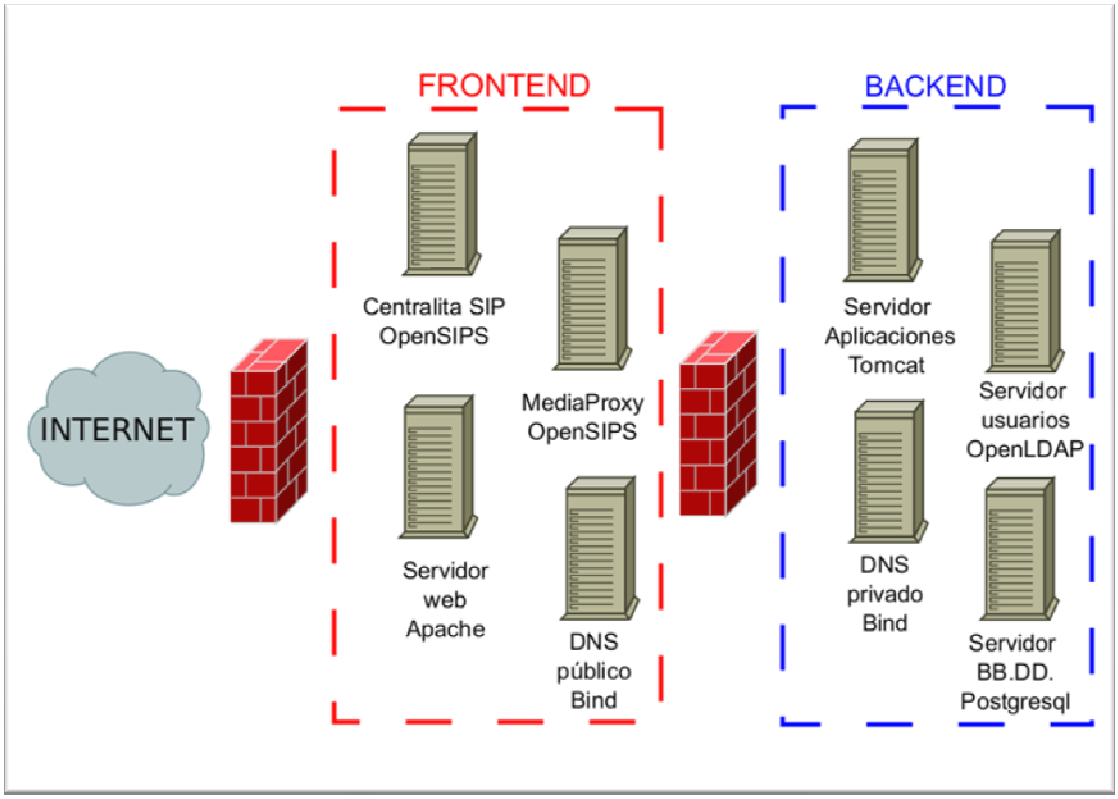

Figura 2: Componentes externos

- Servidores del Frontend:
  - Centralita SIP: Es un servidor OpenSIPS 1.9.1. Las comunicaciones de señalización se harán sobre una conexión TLSv1.0, con CipherSuite openssl HIGH. Esta centralita es la encargada de generar las claves para cifrar los datos de audio y vídeo utilizando un generador de aleatoriedad Simtec Entropy Key certificado con FIPS 140-2. El servidor OpenSIPs también se conecta al servidor de BBDD y al servidor de usuarios que se encuentran en el backend para realizar tareas relativas a gestión de usuarios y sesiones. Se hace uso de certificados con clave RSA de 4096 bits para la autenticación mutua entre servidor y clientes.
  - MediaProxy: Es un servidor OpenSIPS 1.9.1 con el módulo MediaProxy 2.6.1 activo. Se encarga de la distribución del contenido de audio y vídeo cuando la comunicación directa entre clientes no es posible.
  - Servidor web: Es un servidor Apache 2.2.22. Se encarga de ofrecer una serie de servicios web securizados para funciones auxiliares del sistema. Hace de frontal (a modo de proxy) del Backend para consultas por parte del TOE. Se usa autenticación mutua entre el servidor y los clientes, con certificados x.509 de 4096 bits.
  - DNS público: Bind 9.8.4.
- Servidores del Backend:

- Servidor de aplicaciones: Tomcat 7.0.28, donde corre una aplicación propia de gestión de usuarios que a su vez se conecta tanto con el servidor de base de datos para gestión de sesiones como con el servidor de usuarios para gestión de usuarios finales.
- Servidor de usuarios: OpenLDAP 2.4.31.
- DNS privado: Bind 9.8.4.
- Servidor BBDD: PostgreSQL 9.1.14

A continuación se presenta un esquema de la comunicación que se establece cuando dos terminales se comunican haciendo uso del TOE y su entorno:

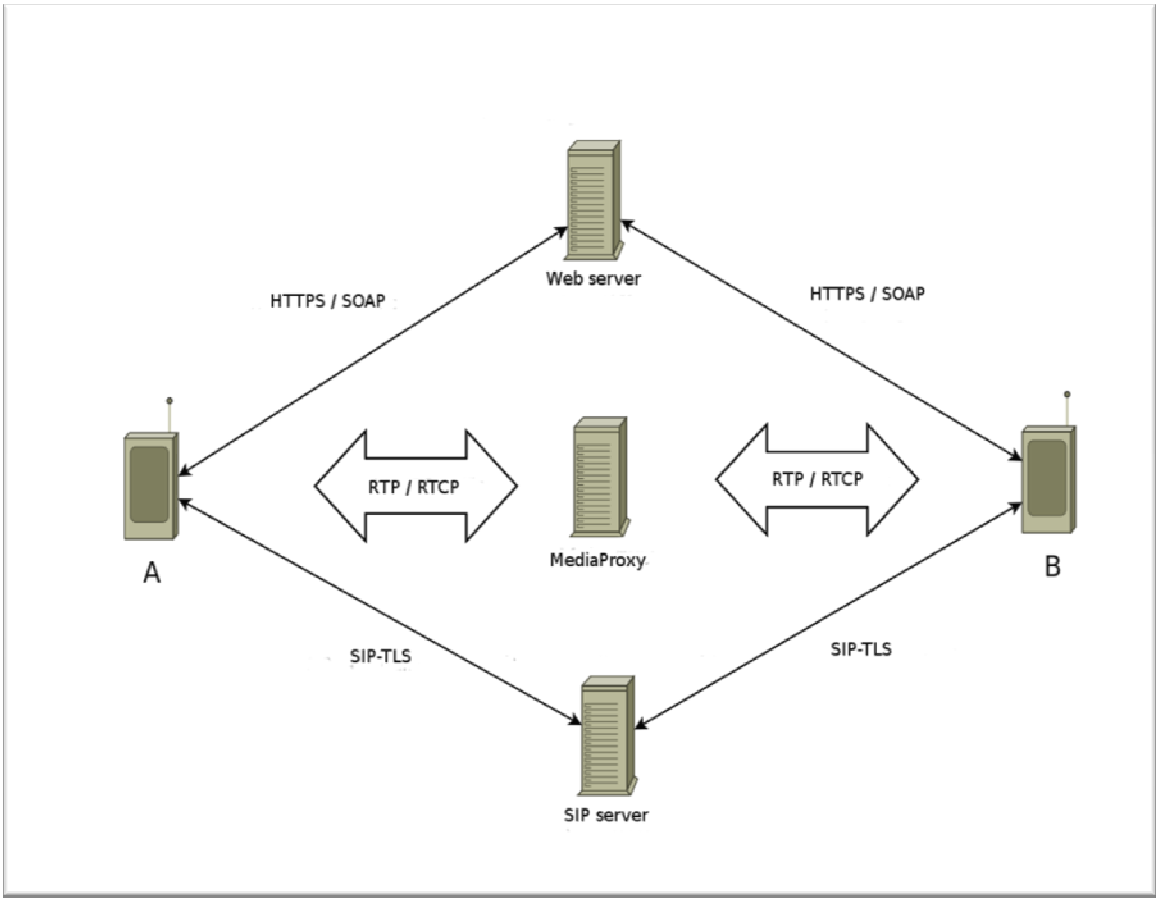

Figura 3: Esquema de comunicación

1.4 Descripción del TOE

1.4.1 Ámbito físico del TOE: Componentes

El TOE es una SDK para dispositivos móviles que es utilizada por aplicaciones Android. Los usuarios del TOE son los desarrolladores que crearán GUIs que, al integrarlos con el TOE, darán como resultado dichas aplicaciones Android.

Por tanto, los componentes de que consta el TOE y que se hacen llegar a los usuarios son:

Elemento	Formato	Nota
TOE	Ficheros de código fuente C y JAVA	El TOE se proporciona al desarrollador a modo de conjunto de librerías
Especificación funcional del TOE	PDF y HTML	Está formada por los siguientes documentos: - Doxygen Enigmedia App SDK, v1.7 - libSRTP Overview and Reference Manual, v1.3 - Web Services Client Android, version 1.10.4

1.4.2 Ámbito lógico del TOE

Desde el punto de vista del ámbito lógico del TOE, se cuentan con las siguientes características de seguridad:

Seguridad en las comunicaciones

Las comunicaciones de señalización (registro en el sistema, inicio de llamada, intercambio de claves, negociación de codecs) hacen uso de autenticación mutua a través de TLSv1.0. Se utiliza una cipherSuite Openssl High con certificados de 4096 bits y cifrados con 3DES mediante una passphrase de 40 bytes. Estos certificados son únicos por usuario final y son generados y cifrados en servidor haciendo uso de una passphrase aleatoria. Una vez generados son enviados a la aplicación que integra al TOE durante el uso del asistente de configuración inicial a través de un canal TLSv1.0. De esta forma no sólo se cifra la comunicación sino que también se autoriza y autentica al usuario final, permitiendo tener en todo momento un control de los usuarios finales registrados, así como la caducidad y revocación de certificados.

Además de las comunicaciones de señalización, se llevan a cabo comunicaciones con un servidor web Apache desplegado en el Frontend. Las comunicaciones con este servidor se realizan a través de HTTPS.

Almacenamiento cifrado

El almacenamiento de la configuración, agenda e historial de llamadas se realiza mediante el cifrado de un fichero de configuración usando AES CBC de 256 bits. La clave AES de 256 bits utilizada para el cifrado y descifrado de dicho archivo es generada mediante key derivation a partir de un PIN que el usuario final de la aplicación que integra al TOE introduce durante la instalación.

El TOE ofrece facilidades de almacenamiento seguro de los mensajes, mediante SQLite y un cipher AES CBC de 256 bits cuya clave se genera aleatoriamente durante el proceso de instalación y se almacena en el fichero de configuración.

VoIP con algoritmo KVC

Las comunicaciones VoIP se cifran haciendo uso del algoritmo propietario KVC y con un autenticación y comprobación de integridad HMAC-SHA1 de 80 bits en cada paquete de datos enviado.

Importación segura de datos

Tanto la configuración de la aplicación una vez que un usuario final se ha dado de alta, como la clave del algoritmo KVC utilizada en cada comunicación VoIP son importadas de manera segura en el TOE, a través de los canales seguros de comunicación.

La generación de las claves del algoritmo KVC se realiza en servidor, usando para ello hardware de generación de entropía certificado como FIPS 140-2.

Gestión de la identidad del usuario

Una vez que el usuario final se ha dado de alta en los servidores, su identidad se almacena en el dispositivo desde el que se ha dado de alta, así como su configuración.

Borrado de seguridad

Las claves asociadas a los algoritmos criptográficos que implementa el TOE son borradas de memoria una vez utilizados.

NOTA: La versión de OpenSSL integrada en el TOE es la 1.0.1k

2 Declaraciones de conformidad

2.1 Conformidad respecto a la norma CC

Esta Declaración de Seguridad cumple con lo indicado en la norma Common Criteria versión 3.1, Parte 2 release 4, y Parte 3 release 4, para un nivel de evaluación equivalente EAL1.

2.2 Conformidad respecto a Perfiles de Protección

Esta Declaración de Seguridad no declara el cumplimiento de ningún Perfil de Protección.

3 Objetivos de seguridad para el entorno operacional

OE.INSTALLATION	El entorno es el encargado de verificar la identidad del usuario a través de una confirmación con envío de SMS, generación de la configuración de la App en sus servidores e inserción de los datos del usuario final en el sistema (LDAP) una vez verificado.
OE.SYNC	El entorno permite el uso de canales seguros entre la aplicación y los servidores, con los que sincronizar los datos de usuario final.
OE.SIGNAL_ENCRYP	El entorno proporciona un servidor de señalización SIP, con capacidad para validar a los usuarios finales mediante certificados.
OE.MEDIA_ENCRYP	Es el entorno el que genera las claves del algoritmo KVC que más tarde se van a usar para cifrar la voz y el vídeo y las distribuye a las dos partes que van a establecer una comunicación o llamada.

4 Definición de componentes extendidos

No se definen componentes extendidos.

5 Requisitos de seguridad del TOE

5.1 Requisitos funcionales de seguridad

La funcionalidad de seguridad del producto se limita a satisfacer los requisitos funcionales, según la parte 2 de CC v3.1 R4 siguientes:

Class FCS	
FCS_CKM.1 Cryptographic key generation. 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 TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [**generación de clave AES usando key generation a partir de un PIN de usuario final**] and specified cryptographic key sizes [**256 bits**] that meet the following: [**ninguno**].
FCS_CKM.4/AES Cryptographic key destruction, 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 TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [**escritura de ceros**] that meets the following: [**none**].
FCS_CKM.4/KVC Cryptographic key destruction, 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 TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [**escritura de ceros**] that meets the following: [**none**].
FCS_COP.1/AES Cryptographic operation. 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 TSF shall perform [**Cifrar, descifrar**] in accordance with a specified cryptographic algorithm [**AES-CBC**] and cryptographic key sizes [**256 bits**] that meet the following: [**none**].
FCS_COP.1/KVC Cryptographic operation. 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 TSF shall perform [**Cifrar, descifrar**] in accordance with a specified cryptographic algorithm [**algoritmo propietario KVC**] and

	cryptographic key sizes [**252 bits**] that meet the following: [**none**].
Class FDP: User data protection	
FDP_ITC.1 Import of user data without security attributes. Requires that the security attributes correctly represent the user data and are supplied separately from the object.	
FDP_ITC.1.1	The TSF shall enforce the [**none**] when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.1.2	The TSF shall ignore any security attributes associated with the user data when imported from outside the TOE.
FDP_ITC.1.3	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: [**sólo se importará la clave del algoritmo KVC cuando se establezca una nueva comunicación VoIP con otro dispositivo**].
FDP_ITC.2 Import of user data with security attributes. Requires that security attributes correctly represent the user data and are accurately and unambiguously associated with the user data imported from outside the TOE.	
FDP_ITC.2.1	The TSF shall enforce the [**none**] when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.2.2	The TSF shall use the security attributes associated with the imported user data.
FDP_ITC.2.3	The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.
FDP_ITC.2.4	The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.
FDP_ITC.2.5	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: [**sólo se reciben los datos para el fichero de configuración tras la validación vía SMS**].
Class FIA: Identification and authentication	
FIA_ATD.1 User attribute definition, allows user security attributes for each user to be maintained individually	
FIA_ATD.1.1	The TSF shall maintain the following list of security attributes belonging to individual users: [**userID**].
Class FTP: Trusted path/channels	
FTP_ITC.1/SIP Inter-TSF trusted channel. Requires that the TSF provide a trusted communication channel between itself and another trusted IT product.	
FTP_ITC.1.1	The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or

	disclosure.
FTP_ITC.1.2	The TSF shall permit [***the TSF, another trusted IT product***] to initiate communication via the trusted channel.
FTP_ITC.1.3	The TSF shall initiate communication via the trusted channel for [**realizar una llamada VoIP cifrada**].
FTP_ITC.1/WEB Inter-TSF trusted channel. Requires that the TSF provide a trusted communication channel between itself and another trusted IT product.	
FTP_ITC.1.1	The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2	The TSF shall permit [***the TSF, another trusted IT product***] to initiate communication via the trusted channel.
FTP_ITC.1.3	The TSF shall initiate communication via the trusted channel for [**obtención de los datos para el fichero de configuración así como consulta y gestión de contenidos albergados en servidores desplegados en el Backend**].

5.2 Dependencias de los requisitos funcionales de seguridad

La siguiente tabla muestra las dependencias existentes entre los requisitos funcionales de seguridad incluidos en el apartado anterior, y el modo en que éstos han sido satisfechos. Para aquellas dependencias no satisfechas, se proporciona una justificación de tal hecho.

SFR	Dependencias	Satisfecha
FCS_CKM.1	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4	SÍ (FCS_COP.1/AES) SÍ (FCS_CKM.4/AES)
FCS_CKM.4/AES	[FDP_ITC.1or FDP_ITC.2 or FCS_CKM.1]	SÍ (FCS_CKM.1)
FCS_CKM.4/KVC	[FDP_ITC.1or FDP_ITC.2 or FCS_CKM.1]	SÍ (FDP_ITC.1)
FCS_COP.1/AES	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	SÍ (FCS_CKM.1)

	FCS_CKM.4	SÍ (FCS_CKM.4/AES)
FCS_COP.1/KVC	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	SÍ (FDP_ITC.1) SÍ (FCS_CKM.4/AES)
FDP_ITC.1	[FDP_ACC. or FDP_IFC.1] FMT_MSA.3	NO (Ver Justificación_1) NO (Ver Justificación_2)
FDP_ITC.2	[FDP_ACC.1 or FDP_IFC.1] [FTP_ITC.1 or FTP_TRP.1] FPT_TDC.1	NO (Ver Justificación_3) SÍ (FTP_ITC.1/WEB) NO (Ver Justificación_4)
FIA_ATD.1	Ninguna	-
FTP_ITC.1/SIP	Ninguna	-
FTP_ITC.1/WEB	Ninguna	-

Justificación_1

Para importar la clave KVC el TOE no ejercita ninguna política de control de flujo/acceso, pues la conexión a través de la que se obtiene la clave es una conexión confiable, y no existen perfiles diferentes dentro del TOE en base a los cuales tratar dicha clave de uno u otro modo. En todos los casos, la clave obtenida se utilizará para cifrar la comunicación que en ese momento se ha establecido con otro terminal haciendo uso del TOE.

Justificación_2

Dado que la clave KVC es un atributo efímero asociado a una única conexión, no se lleva a cabo una inicialización estática del valor de la misma, dado que una vez la conexión termina, la clave es borrada en memoria y sus recursos son liberados.

Justificación_3

Para importar el los datos del fichero de configuración el TOE no ejercita ninguna política de control de flujo/acceso, pues la conexión a través de la que se obtienen los datos para el fichero de configuración es una conexión confiable, y no existen perfiles diferentes dentro del TOE en base a los cuales tratar dicho fichero de uno u otro modo.

Justificación_4

El responsable último de la custodia del fichero de configuración es el propio TOE por lo que no es necesaria una gestión distribuida de la integridad de dicho fichero.

5.3 Requisitos de garantía de seguridad

El desarrollo y evaluación del TOE se realizará conforme al siguiente nivel de garantía: EAL1

A continuación se incluye la lista de requisitos de garantía asociados a EAL1 por referencia a los códigos del catálogo de la parte 3 de Common Criteria version 3.1 R4.

Assurance Class	Assurance components
ADV: Development	ADV_FSP.1 Basic functional specification
AGD: Guidance documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
ALC: Life-cycle support	ALC_CMC.1 Labelling of the TOE
	ALC_CMS.1 TOE CM coverage
ASE: Security Target evaluation	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST introduction
	ASE_OBJ.1 Security objectives for the operational environment
	ASE_REQ.1 Stated security requirements
	ASE_TSS.1 TOE summary specification
ATE: Tests	ATE_IND.1 Independent testing - conformance
AVA: Vulnerability assessment	AVA_VAN.1 Vulnerability survey

5.4 Justificación de los Requisitos de garantía de seguridad

La garantía de seguridad deseada para este tipo de TOE según las exigencias del mercado es la proporcionada por el nivel de evaluación equivalente EAL1.

5.5 Dependencias de los Requisitos de garantía de seguridad

Todas las dependencias en los requisitos de garantía de seguridad se han satisfecho.

6 Especificación Resumida del TOE

A continuación se indica, para cada uno de los requisitos funcionales de seguridad, el modo en que el TOE lo implementa.

FCS_CKM.1

La creación de la clave del algoritmo AES-CBC-256 que se utiliza para cifrar/descifrar el fichero de configuración parte de una proceso de derivación de clave que tiene como origen un PIN introducido por el usuario final de la aplicación que integra el TOE. Para ello se efectúa una llamada al método *EVP_BytesToKey()* de OpenSSL pasando como parámetros los datos obtenidos del dispositivo a modo de array de bytes. El mencionado método devuelve un array de 32 bytes, el cual se transforma en la clave AES de 256 bits utilizada para el cifrado/descifrado del fichero de configuración.

FCS_CKM.4/AES

La destrucción de la clave del algoritmo AES-CBC-256 que se utiliza para cifrar/descifrar el fichero de configuración se realiza utilizando el método *EVP_CIPHER_CTX_cleanup()*, el cual es el encargado de borrar el contexto creado y utilizado para el cifrado/descifrado. Este borrado de contexto implica la eliminación de la clave que se utilizó para la finalidad indicada.

FCS_CKM.4/KVC

El borrado de la clave utilizada para el algoritmo KVC de cifrado de comunicaciones VoIP se realiza de forma activa, escribiendo ceros en la variable que alberga la clave, justo antes de liberarla. Esta funcionalidad descansa en la librería SRTP de Cisco, la cual está integrada dentro del TOE.

FCS_COP.1/AES

El TOE incluye la implementación del algoritmo AES en modo CBC con tamaño de clave de 256 bits. Dicha implementación incluye tanto el cifrado como el descifrado de datos, y es efectuado por métodos de OpenSSL.

FCS_COP.1/KVC

El TOE incluye la implementación del algoritmo propietario KVC, el cual lleva a cabo el cifrado/descifrado de las comunicaciones VoIP. Se trata de un algoritmo de clave simétrica de 252bits, el cual se basa en la implementación de la librería *libsrtp* de Cisco.

FDP_ITC.1

Cuando un TOE lanza una llamada VoIP, la centralita SIP genera una clave aleatoria haciendo uso de un generador de entropía certificado FIPS 140-2. Dicha

clave se entrega al TOE en los paquetes INVITE y ACK. Toda esta comunicación se lleva a cabo a través del canal seguro que implementa la conexión TLS entre TOE y servidor. Una vez que la clave es recibida por el TOE, éste la importa y la usa temporalmente mientras la llamada en curso se mantiene activa.

FDP_ITC.2

Para poder registrarse en la aplicación el usuario final necesita disponer de un certificado válido con el cual conectarse al webservice de configuración, para lo cual se hace uso de un canal seguro TLS. A través de este webservice el usuario final debe introducir una identificación en dos pasos, bien a través de un email o bien utilizando un número de teléfono válido.

Una vez realizada la identificación en dos pasos, y haciendo uso del mencionado canal seguro, el TOE descarga los datos de configuración. Estos datos son importados por el TOE, el cual los almacenará cifrados.

FIA_ATD.1

De entre la información que se almacena en el fichero de configuración personalizado por usuario final, es el propio identificador de usuario.

FTP_ITC.1/SIP

Para poder registrarse o hacer peticiones al servidor SIP se realiza una autenticación mutua basada en los certificados de cliente y servidor con OpenSSL TLSv1 Ciphersuite HIGH. Una vez establecido ese canal seguro, el usuario debe autenticarse con la passphrase que tiene almacenada en el fichero cifrado.

Los paquetes RTP y RTCP que corresponden al contenido y estadísticas de la llamada se cifran y autentican antes de su envío mediante el método *srtp_protect()* de *libSRTP* (utilizando el cipher KVC) y se descifran y verifican su autenticidad mediante el método *srtp_unprotect()*.

FTP_ITC.1/WEB

Para poder registrarse o hacer peticiones al servidor HTTPS se realiza una autenticación mútua basada en los certificados de cliente y servidor con OpenSSL TLSv1 Ciphersuite HIGH. Una vez establecido ese canal seguro, el usuario final debe autenticarse con la passphrase del servidor de usuarios LDAP que tiene almacenada en el fichero cifrado. La comunicación a través del canal seguro se hace a través de Web Services (implementados en Java).