UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR

PROYECTO FIN DE CARRERA

"Desarrollo de una Aplicación de Control de Seguridad basada en el modelo SERENITY en un escenario de comunicaciones inalámbricas"

Jesús Marcos Morell

Junio 2009

"Desarrollo de una Aplicación de Control de Seguridad basada en el modelo SERENITY en un escenario de comunicaciones inalámbricas"

AUTOR: Jesús Marcos Morell
TUTOR: Germán Montoro Manrique

Dpto. de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Junio de 2009

Resumen:

SERENITY (System Engineering for Security and Dependability [1]) es un proyecto europeo de investigación y desarrollo financiado por la Unión Europea. Persigue proveer de seguridad los sistemas de inteligencia ambiental ("Ambient Intelligence"), también denominados sistemas AmI. Este proyecto está siendo desarrollado por una variedad de empresas tanto españolas como europeas. El objetivo del proyecto SERENITY es proporcionar un marco para el tratamiento automatizado de la seguridad y cuestiones de fiabilidad en escenarios AmI.

El trabajo presentado en esta memoria se centra en el enfoque de SERENITY dentro de los escenarios de comunicaciones inalámbricas. Gracias a este enfoque se trabaja dentro de escenarios AmI con aplicación directa en la industria.

El escenario propuesto se centra en la prestación de acceso controlado y comunicación a través de una red inalámbrica. Esta red proporciona conexión y acceso a los recursos de un espacio de trabajo, tales como documentos internos, bases de datos o conexión tanto a Intranet como a Internet. En este proyecto la seguridad es un factor muy importante, por lo tanto hay diversos factores para evaluarla.

En una red de una compañía convencional las políticas de seguridad son asignadas normalmente de un solo modo, es decir sin incluir las características AmI o cambios dinámicos en tiempo real. Se requiere un profundo conocimiento de las características de seguridad y fiabilidad, a partir de ahora las nombramos como S&D (Secure and Dependability, Seguras y fiables), para el despliegue de la red inalámbrica y el control de acceso y así conseguir que la red inalámbrica en cuestión sea mucho más segura.

Por tanto hemos querido mejorar esta situación mediante la inclusión de las características AmI y el modelo SERENITY en el escenario de comunicaciones inalámbricas. En este escenario de comunicación AmI, se consideran varios elementos con el fin de mejorar la Seguridad y la Fiabilidad, S&D. En particular, los factores de seguridad que hemos tenido en cuenta son: la ubicación del usuario, la autenticación del mismo y la identificación del dispositivo mediante el que está accediendo.

La localización, la autenticación y la identificación de dispositivos son puntos muy importantes en las políticas de acceso del escenario, algunos recursos pueden ser o no ser accesibles dependiendo de la posición actual del usuario que está pidiendo dicho recurso, dependiendo también del tipo de dispositivo que esté utilizando y por supuesto dependiendo del usuario que esté intentando acceder al recurso. Solamente los usuarios que se encuentren en zonas seguras, estén debidamente autenticados (y tengan perfiles con suficientes privilegios) y sus dispositivos estén apropiadamente identificados (estos también con suficientes privilegios) tendrán acceso a ciertos recursos dentro del entorno.

Palabras clave:
SERENITY, inteligencia ambiental, seguridad y fiabilidad, localización Wi-fi.

Agradecimientos

Quiero agradecer en primer lugar de todo a mis padres, Jesús y Rosa, que sin ellos nada de esto podría haber sido posible. Todo el apoyo recibido en todas las aventuras que he empezado, en especial cuando dije que quería ser Ingeniero, una aventura difícil que sin embargo ahora se ve fácil.

Quiero agradecer a Víctor, a Ana Rosa a mis abuelas, la Abuela y la Nona, a mis abuelos, el Abuelo y el Lolo que les hubiera encantado ver esto, a todos mis primos y mis tíos que siempre habéis creído en mí y a toda mi familia en general por que siempre me habéis apoyado y siempre habéis confiado en mí.

A todos mis amigos del barrio Kike, Víctor, Sando, Guille, Marta, Leticia, Raúl, Saúl, Resta, Flórez, Santi, Rober, Germán, Kuko, Javi, Ana, Miguel sin vosotros esto hubiera sido mucho más difícil, sois la hostia.

A Paula, Susana, Sofía, Miriam, Estefanía, Monica y a todos aquellos que se me olvidan y que me han apoyado y han creído en mí todos estos años, gracias a todos, sin vosotros todo esto hubiera sido mucho más difícil.

Gracias Leticia por toda tu confianza y por tu apoyo incondicional, gracias por estar siempre ahí y por seguir estando ☺

Gracias a todos los nuevos amigos que hice a lo largo de este viaje y sobre todo gracias a los SuperPEPOS, gracias Gustavo, Pablo, Kiko y a la última incorporación Peter, gracias a todos por creer en mí. Gracias también a las chicas Elena y Vero. Ha sido muy importante contar con vosotros para llegar a donde me encuentro.

Gracias Mónica, Esther, Sonso, Ricardo, Beavis, Castro, Tato, Nacho, Nico, Ana, Imanol,… y a todos los que habéis hecho de esto toda una aventura imposible de repetir.

A todos los miembros de la Escuela Politécnica Superior que conseguís que la gente crezca realmente como personas, que aprendamos a la vez que vosotros. Gracias a todos los profesores de Telecomunicación, gracias Jesús Bescós, Chema, Jose Luis, Bazil, Jorge, Doroteo, Javier, Antonio, Eloy, Manuel, Daniel,… y tantos que se me olvidan, muchas gracias a todos.

Quiero agradecer lo que representa este proyecto a uno de los principales artífices, Germán Montoro, sin ti este proyecto no podría haber visto la luz. A Álvaro y a Pablo, que supusieron una parte muy importante también. Y a todos mis compañeros de Telefónica I+D, gracias Ariel, Andoni, Laura, Elena, Jose, Rocío, Laura, Oscar, Juan Carlos, Ángel, Soriano,… por vuestra inestimable ayuda y apoyo y por esos grandes partidos de pádel, y como no por los pinchos y las cañas de los viernes. Gracias a todos.

Gracias a ti Natalia por todo el apoyo que me has dado y por todo este tiempo que hemos compartido, sin ti este proyecto hubiera sido mucho más difícil de hacer. Te deseo muchísima suerte y te estoy muy agradecido por todo lo que me has dado. Gracias.

Gracias a todos los que os habéis cruzado en mi vida y ahora no mantenemos tanto el contacto, gracias Kike, Alberto, Rocío, María, Héctor, Verónica y tantos otros que se me olvidan. Gracias a todos vosotros por compartir buenos momentos y por confiar en mí.

Gracias a los jueves de cañas y gracias a todo el mundo que hace que la gente cumpla sus sueños.

Gracias a que un día hace mucho tiempo dije que quería ser Ingeniero de Telecomunicación y he tenido la posibilidad de recorrer este gran camino…, parece mentira, pero a mis veintitrés años lo he conseguido, ha sido completada una primera etapa de un gran viaje ¡Soy Ingeniero Superior de Telecomunicación!

A todos, muchas gracias.

Jesús Marcos Morell

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

1 Introducción

1.1 Motivación

Uno de los aspectos fundamentales de los nuevos escenarios de "Ambient Intelligence" es la seguridad. En este Proyecto trataremos de considerar estos problemas de seguridad y dar una visión general de las soluciones propuestas actualmente. El escenario que proponemos es un escenario AmI real donde esas soluciones son aplicadas para valorar si son efectivas dentro del mismo.

Hemos querido tener en cuenta tres características que consideramos fundamentales en las redes inalámbricas para aumentar la seguridad, como son la localización, la identificación del dispositivo y la autenticación del usuario. Si alguna de estas tres características se modifica lo lógico es que en un escenario AmI real también se modifiquen las seguridad, ya que no es lo mismo estar en una sala pública que en el despacho, no es lo mismo cambiarse de portátil y acceder con el portátil de otro y por supuesto, no es lo mismo cambiar de usuario.

Por tanto la motivación fundamental del proyecto consiste en dotar de seguridad a un escenario real de comunicaciones inalámbricas en el que las personas interactúan y no tienen por qué estar pendientes de los elementos de seguridad. Es el servidor el que se preocupa por la seguridad y los usuarios (trabajadores de empresa, estudiantes, usuarios domésticos…) pueden dedicarse a hacer sus cometidos teniendo la confianza de que están conectados a una red que vela por la seguridad de los datos y recursos a los que pueden acceder en tiempo real.

Por otro lado es lógico que no todos los recursos presentes en una red deban ser accesibles por todos los usuarios conectados a ella, por tanto en este proyecto proponemos la asignación de unos niveles de seguridad necesarios para acceder a cada recurso. En definitiva cada usuario tendrá asignado un nivel de seguridad, así como también lo tendrá cada dispositivo y cada localización. Y sólo los que posean estos tres niveles de seguridad suficientemente altos podrán acceder a los recursos en cada momento.

Por ejemplo el director de una compañía conectado en su despacho y con su portátil tendrá acceso a todos los recursos, en cambio, si trata de acceder a algún recurso protegido desde la sala de espera o desde la recepción, el sistema no deberá permitir el acceso.

En conclusión hemos desarrollado un escenario de comunicaciones inalámbricas real, con control de acceso a los recursos compartidos por medio de la seguridad proporcionada por tres factores: localización, identificación de dispositivo y autenticación de los usuarios. Proporcionando seguridad a toda la red en tiempo real y sin que los usuarios tengan que estar pendientes de lo que hacen en cada momento, ya que se el servidor está pendiente por ellos. Este escenario utiliza la tecnología desarrollada dentro

del proyecto SERENITY para integrar y comunicar las diferentes soluciones de seguridad implementadas.

1.2 Objetivos

Como objetivos de este proyecto fin de carrera podemos destacar los siguientes:

- Desarrollo de escenarios completos de comunicaciones aplicables a escenarios reales.
- Crear mecanismos para proporcionar seguridad vía localización al administrador del sistema.
- Permitir ofrecer seguridad vía identificación de dispositivos al administrador del sistema.
- Establecer un sistema de seguridad vía autenticación de usuarios al administrador del sistema.
- Aplicar el enfoque SERENITY al escenario para que la seguridad esté avalada por un proyecto europeo de sus características.
- En definitiva el objetivo fundamental de este proyecto es la realización de un escenario de comunicaciones inalámbricas seguro. Con control de acceso a los recursos compartidos mediante políticas de seguridad definidas por el administrador.

1.3 Organización de la memoria

La memoria consta de los siguientes capítulos:

2 Estado del arte

Para estructurar esta sección vamos a tratar en primer lugar el proyecto SERENITY, debido a la importancia que tiene dentro de este proyecto fin de carrera. Posteriormente introducimos el estado de las principales tecnologías utilizadas. En última instancia citamos proyectos que están relacionados y que nos pueden ayudar a comprender el entorno.

2.1 Proyecto SERENITY

El proyecto SERENITY [2] se centra en dos pilares principales: (i) en la captura de la experiencia específica de los expertos de seguridad de una manera que permita su tratamiento automatizado, y (ii) en proporcionar los medios para llevar a cabo la monitorización de la seguridad y de los mecanismos de fiabilidad en tiempo de ejecución. Todo esto se ha desplegado por medio de:

1. Un conjunto de modelos de objetos utilizados para captar los conocimientos de seguridad (llamados "S&D artefacts", S&D significa seguros y fiables en sus siglas en inglés). Utilizamos el término "S&D solution" para referirnos a un componente que proporciona un servicio de seguridad y/o fiabilidad a una aplicación. Los artefactos S&D se utilizan para representar soluciones S&D a diferentes niveles de abstracción. La representación de soluciones S&D a diferentes niveles de abstracción responde a las necesidades de su uso en las diferentes fases del proceso de desarrollo de software.
2. Un marco en tiempo de desarrollo, el "SERENITY Development-time Framework" llamado SDF, que a su vez se apoya en:
   - El desarrollo de soluciones S&D por medio de artefactos S&D. El SDF incluye los procesos y herramientas utilizadas por los expertos en seguridad para la creación de nuevas soluciones S&D. Las soluciones desarrolladas poseen información semántica relacionada con su descripción y su comportamiento operativo.
   - El desarrollo de aplicaciones seguras basadas en SERENITY. Estas aplicaciones dependen de SERENITY para garantizar el cumplimiento de sus requisitos de seguridad y fiabilidad. Dentro de estas aplicaciones se incluyen referencias a artefactos S&D. Las referencias entre las aplicaciones y los artefactos S&D se resuelven de manera que los artefactos S&D ofrecen soluciones y las aplicaciones utilizan dichos artefactos.

   El SDF se apoya en los repositorios en línea en los que se encuentran los artefactos S&D. Los expertos en seguridad utilizan estos repositorios en línea con el fin de

almacenar las soluciones S&D que desarrollan. Y los desarrolladores de aplicaciones acceden a los depósitos cuando están desarrollando aplicaciones SERENITY con el fin de buscar soluciones S&D a las que referirse en sus aplicaciones.

3. Un marco en tiempo de ejecución, el "SERENITY Run-time Framework (SRF) [3]", en realidad representa el núcleo de una aplicación SERENITY, es el responsable de la selección de la solución S&D que cumpla todos los requerimientos de la aplicación y de aportar dicha solución a la aplicación que la solicita. El SRF proporciona soporte a las aplicaciones en tiempo de ejecución, administrando las soluciones S&D y monitorizando el contexto del sistema. Las aplicaciones basadas en SERENITY se desarrollan a través de arquitecturas abiertas que son completadas en tiempo de ejecución por el SRF. En definitiva el SRF es el núcleo de una aplicación SERENITY, ya que es el encargado de gestionar las peticiones de la aplicación que solicitan las soluciones S&D.

   El proyecto SERENITY ofrece cinco artefactos S&D principales para representar soluciones S&D, que están representados en la Figura 1. Estos son:

   - Las Clases S&D representan abstracciones de un servicio de seguridad que puede ser proporcionado por las diferentes soluciones S&D. Estas soluciones S&D se caracterizan por proporcionar las mismas propiedades de seguridad y fiabilidad y tener una interfaz común. A nivel de las clases S&D la descripción de las soluciones S&D es muy simple. Contiene alguna información sobre el nombre de la solución, sus creadores, las propiedades de seguridad, y la interfaz que ofrece la solución.

   - Los Patrones S&D son descripciones detalladas que resumen las soluciones S&D. Tal como se presenta en la figura 1, cada uno de los patrones pertenece, al menos, a una clase. En este nivel de abstracción la descripción de las soluciones S&D es más detallada que en la anterior (Clases S&D). Estas descripciones contienen toda la información necesaria para la selección, instanciación, la adaptación y la aplicación dinámica de la solución de seguridad representada en el patrón. Los patrones también incluyen la descripción de los mecanismos de seguridad que representan. Por último los patrones incluyen información acerca de las restricciones impuestas por la solución Estos componentes son de lectura mecánica.

   - Los Esquemas de Integración (ISs) son un tipo especial de patrones que representan las soluciones S&D que se construyen mediante la combinación de varios patrones. Mientras que los patrones son descripciones atómicas de soluciones S&D, los Esquemas de Integración son descripciones de soluciones

complejas que vienen dadas por la combinación de varias soluciones S&D más simples.

- Las Implementaciones S&D representan los componentes que realizan las soluciones S&D. Las implementaciones S&D no constituyen realmente la solución, sólo dan su representación y su descripción. Este es el menor nivel de abstracción. A este nivel las soluciones S&D se definen en términos de la tecnología utilizada en su desarrollo y cómo hacer uso de ella.
- Por último, los Componentes Ejecutables son implementaciones reales (se corresponden con código) de soluciones S&D. El procesamiento automático de los ECs se basa en el uso de la información proporcionada por los artefactos S&D que representan la solución S&D implementada por ellos.

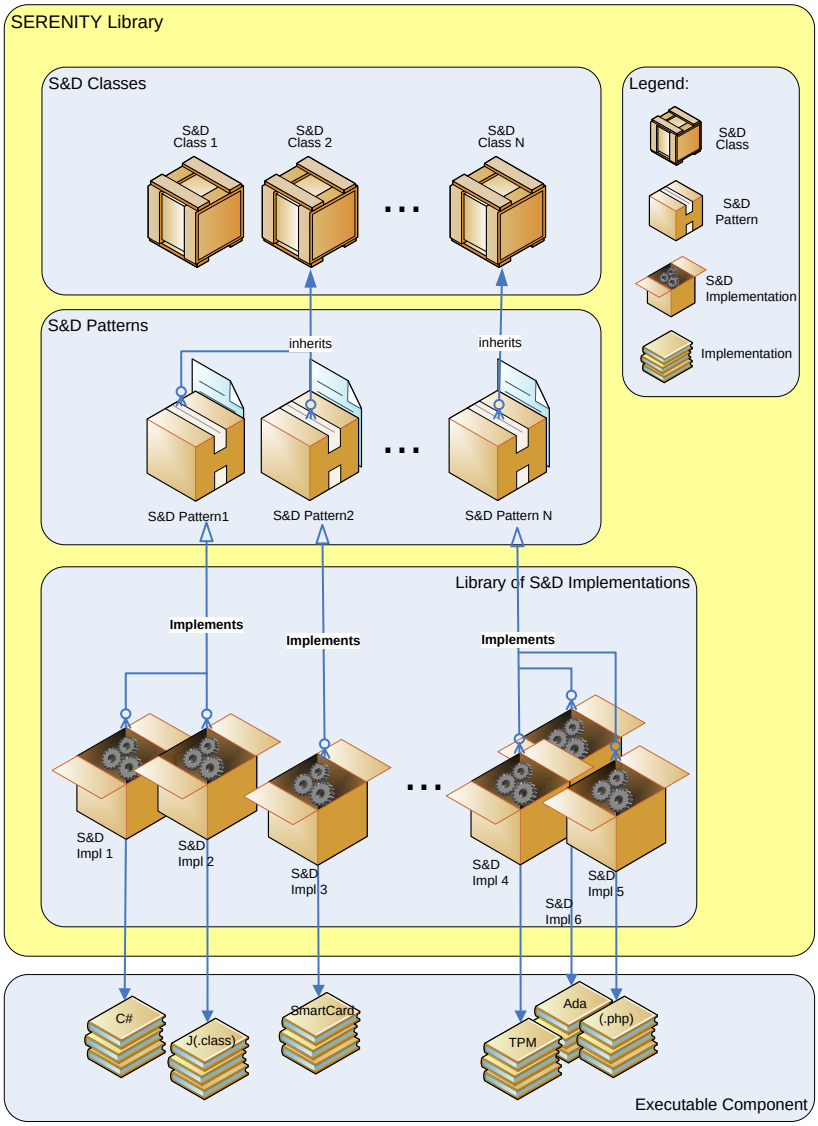


Figura 1 Artefactos SERENITY

Las soluciones S&D representan soluciones utilizando las descripciones semánticas a diferentes niveles de abstracción. Todos los artefactos S&D, menos los ECs, que se corresponden con código, son representados mediante archivos XML (ver ANEXO C). En el desarrollo de este proyecto se desarrollarán todos estos niveles de abstracción para la implementación de cada una de las soluciones S&D requeridas.

Las clases, los patrones y las implementaciones son artefactos S&D orientados al tiempo de desarrollo y los ECs son especialmente adecuados para el tiempo de ejecución. Estos artefactos S&D están organizados como una jerarquía, es decir, cada clase tiene varios patrones y cada patrón diversas implementaciones, cada una de las implementaciones se refiere a un solo EC.

Una parte también importante del proyecto SERENITY es el Servicio de Monitorización, que se encarga de verificar que los Componentes Ejecutables que están corriendo tengan una comunicación a tiempo real con la Aplicación. Esta comunicación del servicio de monitorización y la aplicación es a través del SRF (estas comunicaciones se detallan en la sección 4.2.2).

Por tanto el proyecto SERENITY nos proporciona:

1. El SRF, que es el núcleo del proyecto (ver sección 4.2.2.2).
2. El modelo de Soluciones S&D muy modularizado e intercambiable, que nos aporta una gran versatilidad al proyecto y una gran interoperabilidad.
3. El Servicio de Monitorización que informa a la Aplicación de Control de Seguridad del estado de los Componentes Ejecutables en tiempo real.

2.2 Escenarios AmI

AmI (Ambient Intelligence [4]) se refiere a entornos inteligentes que son sensibles y reaccionan a la presencia de personas. En un mundo AmI, los dispositivos trabajan para proporcionar apoyo a la gente en sus actividades diarias desde un punto de vista fácil y natural, usando información e inteligencia que se encuentra en las redes que conectas estos dispositivos. En cuanto estos dispositivos sean más pequeños y estén más conectados y más integrados en nuestro entorno, la tecnología sólo será perceptible en la interfaz del usuario.

El paradigma AmI está centrado en la interacción entre el ser humano y el ordenador cuyo diseño está caracterizado por sistemas y tecnologías que son:

- Embebidas: muchos dispositivos interconectados están integrados en el entorno.
- Conscientes del contexto: estos dispositivos pueden reconocer al usuario y su situación.
- Personalizadas: pueden ser a la medida de las necesidades del usuario.

- Adaptativas: pueden cambiar en respuesta al usuario.
- Anticipativas: pueden anticiparse a las peticiones del usuario.

2.2.1 Seguridad y fiabilidad en entornos AmI

Las aproximaciones a los entornos AmI suelen basarse en elementos de software reutilizables que resuelven un problema bajo unas condiciones de contexto determinadas, proporcionadas por un conjunto de interfaces bien definidas y una descripción asociada de su comportamiento.

En general las aproximaciones de entorno AmI están formadas por interfaces y componentes estándar. Por tanto tenemos acceso a un conjunto de capacidades especializadas, potentes y complejas. De todas maneras hay un gran coste computacional para estas aproximaciones y, por tanto, es difícil incorporar este software a dispositivos de pequeño tamaño. El potencial de uso de un Framework está en el desarrollo de servicios seguros ya que puede usar un criterio común para clasificar los requerimientos de seguridad descritos. Esto está en una estrecha relación con nuestro proyecto, debido al uso de requerimientos previamente definidos y a la necesidad de reutilización.

El concepto de patrón de seguridad ha sido introducido para ayudar al sistema a seleccionar la solución de fiabilidad o seguridad apropiada. Algunos de estos patrones de seguridad usan representaciones precisas. Por tanto lo que buscamos con estos patrones es disminuir la ya complicada tarea del programador en entornos AmI. Al definir claramente los requerimientos y a su vez los patrones de seguridad estamos reduciendo el grado de dificultad de la programación en estos entornos.

En los nuevos escenarios AmI no hay sólo aplicaciones generales, también hay aplicaciones individuales funcionando en estos escenarios o apoyándolos. Por tanto los sistemas tienen que adaptarse a cambios dinámicos de hardware y software, configuraciones de firmware, aparición y desaparición de dispositivos y componentes software. En definitiva las aplicaciones tienen que ser capaces de adaptarse dinámicamente a los nuevos entornos de ejecución. Las relaciones fiables entre componentes, aplicaciones y sus entornos no pueden continuar estando garantizadas. Por tanto la complejidad de prever todas las posibles situaciones e interacciones es muy elevada y por consiguiente la creación de soluciones que cubran los requerimientos en cuanto a seguridad y fiabilidad de los usuarios. Por otro lado en un entorno AmI global hay dispositivos, software y otras infraestructuras de comunicación que no están bajo el control de la aplicación AmI, esto nos lleva a que las aproximaciones basadas en el nivel de seguridad de la aplicación no son suficientes para proporcionar seguridad y fiabilidad a todo el entorno AmI como conjunto.

Una característica relevante de un entrono AmI es que contendrá un gran número de infraestructuras informáticas y de comunicación y un gran número dispositivos que proporcionarán nuevas funcionalidades y facilitarán la vida diaria del usuario. Estos

dispositivos poseerán una variedad de datos con diferentes requerimientos de seguridad y privacidad, por tanto tendrán que ser aplicadas distintas políticas de S&D. En este caso proteger por separado al dispositivo y a la información no es suficiente para garantizar la seguridad. En definitiva un análisis del contexto tendrá que ser evaluado para ser capaces de elegir un sistema de seguridad apropiado en tiempo real.

Debido a su complejidad y a los elementos que están bajo control de distintos usuarios, tiene que haber mecanismos de seguridad que supervisen e identifiquen posibles ataques y decidan acciones en la medida de lo posible. Hay algunas aproximaciones existentes que pueden proporcionar soluciones adecuadas para soportar la evolución dinámica de las políticas de seguridad para mecanismos de seguridad específicos en determinados niveles operativos, como aplicación o red, éstas aproximaciones se detallan en la sección 2.5. En definitiva estas aproximaciones no pueden extenderse para soportar la evolución dinámica de mecanismos de seguridad generales, y en todo caso sus resultados serían extremadamente complicados de integrar, monitorizar y evolucionar dinámicamente en todo un entorno AmI. Por las mismas razones, no se puede esperar que las aproximaciones S&D para entornos AmI sinteticen los nuevos mecanismos S&D o las nuevas combinaciones completamente automáticas y dinámicas para estos mecanismos. Por tanto podemos resumir todos estos desafíos individuales en un gran desafío conjunto:

Para obtener S&D en entornos AmI debemos aplicar los conocimientos de los expertos en seguridad para poder llegar dinámicamente a contextos cambiantes e imprevistos. La solución intuitiva podría ser crear un sistema inteligente capaz de analizar los requerimientos y el contexto para sintetizar nuevas soluciones. Desafortunadamente, dado el estado actual de la seguridad y la inteligencia en sistemas, esta aproximación no es factible en un futuro cercano. Para poder resolver este desafío en un tiempo razonable estamos proponiendo este proyecto.

La seguridad relacionada con los problemas comunes de los sistemas puede ser clasificada en tres categorías dependiendo de si tratan confidencialidad, integridad o disponibilidad. Ahora pasamos a describir cada una de ellas.

- La confidencialidad es la propiedad por la cual la información sólo es conocida por los agentes autorizados. En el caso de los entornos AmI donde la comunicación se realice a través de redes inalámbricas la información puede ser transmitida a cualquiera que esté dentro del alcance de la red. De este modo las soluciones previstas para seguridad en redes inalámbricas también podrán ser tenidas en cuenta para los entornos AmI.

  Entre las técnicas más usadas para sistemas distribuidos podemos mencionar las basadas en encriptado y desencriptado. Para conseguir confidencialidad se usan métodos de clave pública y de clave privada.

- La integridad es violada cuando la información es alterada sin autorización. Esta definición se aplica a la información que está en un host y a la información que se transmite entre varios hosts. Por tanto tenemos que las redes inalámbricas son más vulnerables que las cableadas.

  Algunas de las técnicas más usadas en sistemas distribuidos son códigos de detección de errores, tablas Hash o firma digital. Estas técnicas también podrían ser aplicadas a los nuevos entornos AmI, y por tanto solventar los problemas de integridad.

- La disponibilidad garantiza y legitima las peticiones entre las partes autorizadas. Un posible ataque puede ocurrir cuando un usuario malicioso fuerza el colapso del sistema y provoca que el usuario autorizado sea denegado por sobrecarga del sistema. En cualquier entorno AmI donde los usuarios están conectados al host es posible evitar un ataque denegando el servicio, aunque esto pone en riesgo a su vez la disponibilidad del sistema.

2.3 Componentes Confiables Hardware

En esta sección vamos a presentar una serie de dispositivos hardware relevantes que están disponibles en el mercado, con el fin de dar una visión amplia a la parte que trata de fiabilidad vía hardware del proyecto.

Una vez acabada esta sección expondremos los motivos del hardware escogido y plantearemos también distintas opciones en torno a él.

2.3.1 Microprocesadores y chipsets

2.3.1.1 Intel Trusted eXecution Technology

La Intel Trusted eXecution Technology (Intel TXT [5]), comúnmente conocida como LaGrande, propone una serie extensiones hardware para sus procesadores y chipsets, las cuales, con el software apropiado, mejoran las capacidades relacionadas con la seguridad de la plataforma. Intel TXT proporciona una seguridad basada en hardware que ayuda a establecer un nivel de protección mayor para la información guardada, procesada e intercambiada a través del PC.

La TXT ha sido diseñada para proteger al PC antes posibles ataques software. Las características que proporciona esta tecnología son:

- Proteger la ejecución y los espacios de memoria donde se procesan datos sensibles, de forma que queden fuera del alcance del software de alto nivel.
- Sellado en el almacenamiento de claves de cifrado para establecer un escudo ante los posibles ataques.

- Mecanismo de certificación a través del cual de un sistema garantiza la correcta invocación de la tecnología TXT, así como la verificación del software que esta corriendo en el mismo.
- Protección de memoria para:
    - Mejorar la protección de los recursos del sistema.
    - Aumentar la confidencialidad y la integridad de los datos.
    - Mejorar la veracidad de las transacciones.
    - Aumentar la protección de información sensible.

2.3.1.2 vPro Technology

La tecnología vPro [6], que se encuentra en su tercera generación, integra todas las mejoras que proporciona la TXT, para implementar nuevos procesadores orientados a mejorar la seguridad de los PCs. Actualmente, trescientas cincuenta empresas del sector han comenzado a comercializar esta nueva tecnología. Tres de las más importantes son: Dell, Lenovo y Hp.

2.3.1.3 AMD

Al igual que su competencia, AMD también está desarrollando procesadores orientados a la seguridad y a la gestión de derechos digitales.

Mediante *Presidio* y sus sucesores AMD pretende proveer a los procesadores y chipsets de distintas características, tales como arranque y particionamiento seguro, y *paths* de E/S seguros.

Estos procesadores y chipsets, totalmente compatibles como el "Next Generation Secure Computing Base" de Microsoft, tienen en cuenta las recomendaciones y estándares relacionados como la computación confiable y la atestación remota.

2.3.1.4 Microsoft Palladium (security chip)

La Next-Generation Secure Computing Base (NGSCB), también conocida como Palladium, es una nueva arquitectura software que implementa algunas partes de las especificaciones propuestas por la Trusted Computing Group (TCG [7]). Sin embargo, la arquitectura propuesta por Microsoft no depende únicamente de componentes software, sino que también se basa dos dispositivos hardware especializados. Éstos son el Trusted Platform Module (TPM [8]), y una memoria físicamente separa en el procesador central. El primero de ellos, se utiliza para el almacenamiento seguro de las claves de encriptación criptográfica, mientras que la segunda se utiliza para que el acceso a los datos por parte de las aplicaciones de alto nivel se realice de forma controlada. Es decir, las aplicaciones sólo podrán acceder a los datos que les pertenezcan.

2.3.2 Dispositivos con procesador criptográfico

En lo relativo al tema de qué tipo de procesador seguro utilizar en los modelos de computación confiable y/o protegida, existen diferentes alternativas (incluso se podrían emplear varios de manera simultánea).

Por un lado, existe la posibilidad de usar la capacidad de proceso criptográfico de procesadores como el 4758 PCI de IBM y otros productos similares de la firma nCipher, o incluso se podrían emplear TPMs (Trusted Platform Modules) como coprocesadores seguros.

Por otro lado, se pueden utilizar dispositivos móviles y reemplazables, como es el caso de las tarjetas inteligentes USB como coprocesadores seguros, capaces igualmente de demostrar la aplicabilidad práctica de los modelos de Computación Confiable y/o Computación Protegida.

2.3.2.1 TPM (Trusted Plaftform module)

El TPM facilita la generación de claves criptográficas, así como la limitación de su uso. Además posee un generador hardware de números pseudo-aleatorios. También incluye características como certificación remota y almacenamiento sellado.

Por un lado, la certificación remota crea una clave hash muy difícil de falsificar, que resume la configuración hardware y software. La medida del resumen del software se decide por el programa de cifrado de los datos. Esto permite que un tercero pueda verificar que el software no ha sido cambiado. Por el otro, el almacenamiento sellado cifra los datos de tal manera que puede ser descifrado sólo si el TPM libera la clave de descifrado asociada, algo que sólo hace para el software que pueda proporcionar la contraseña que se suministra cuando el TPM fue inicialmente configurado.

Un TPM puede ser utilizado para autenticar dispositivos hardware. Dado que cada chip TPM tiene una única clave secreta RSA introducida cuando es producido, es capaz de realizar la autenticación de plataforma. Por ejemplo, puede ser utilizado para comprobar que un sistema es realmente quien dice ser.

En general, como ya se ha comentado anteriormente, llevar la seguridad a nivel de hardware, incluyendo dependencias con el software, ofrece más protección que una solución software exclusivamente, ya que ésta es más fácilmente susceptible. Sin embargo, aun cuando se utiliza un TPM, la clave sigue siendo vulnerable ya que una aplicación software obtiene la clave del TPM y la utiliza para el cifrado/descifrado.

Los componentes que forman el TPM pueden verse en la siguiente figura:

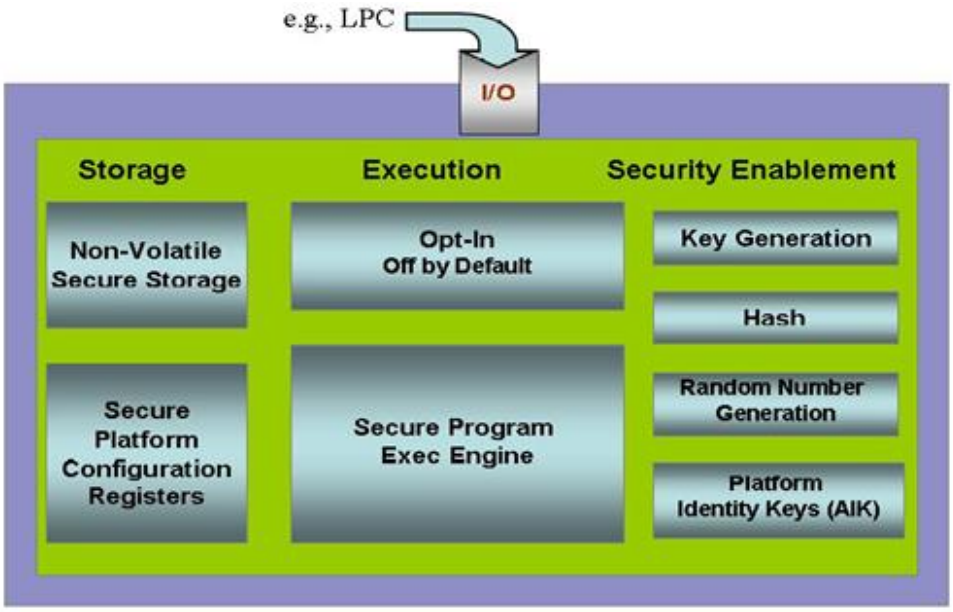


Figura 2 Componentes del TPM

La naturaleza de la criptografía basada en hardware asegura que la información almacenada en hardware está mejor protegida ante ataques externos. Existen multitud de aplicaciones, que utilicen este almacenamiento seguro de claves, que pueden ser desarrolladas. Estas aplicaciones hacen mucho más difícil el acceso a la información sobre dispositivos informáticos sin la debida autorización (por ejemplo, si el dispositivo fue robado). Si la configuración de la plataforma ha cambiado como resultado de actividades no autorizadas, el acceso a datos secretos puede ser negado y cerrado el uso de estas aplicaciones.

Sin embargo, es importante tener en cuenta que un TPM no puede controlar el software que se ejecuta en un PC. El TPM puede almacenar antes de tiempo de ejecución los parámetros de configuración, pero es misión de otras aplicaciones determinar y aplicar las políticas relacionadas con esta información.

Los Procesos que necesitan securizar secretos, tales como la firma digital, pueden hacerse más seguros con un TPM. Además, aplicaciones críticas que requieren una mayor seguridad, tales como correo electrónico seguro o la gestión de documentos segura, pueden ofrecer un mayor nivel de protección cuando se utiliza un TPM. Por ejemplo, si en el momento del arranque se determina que un PC no es digno de confianza debido a cambios inesperados en la configuración, el acceso altamente seguro a las aplicaciones puede ser bloqueado hasta que el problema se solucione (si una política se ha establecido que requiere tal acción). Con un TPM, se puede tener más certeza de que los artefactos necesarios para firmar mensajes de correo electrónico seguro no se han visto afectados por ataques de software. Y, con el uso de la certificación a distancia, otras plataformas en la

red de confianza puede hacer la determinación de hasta qué punto se puede confiar en la información de otro PC.

Estas capacidades pueden mejorar la seguridad en muchas áreas de la informática, incluido el comercio electrónico, las aplicaciones de e-gobierno, banca online, comunicaciones confidenciales y muchos otros ámbitos en los que una mayor seguridad es obligatoria. Basando la seguridad en hardware también pueden ser mejoradas. Los algoritmos criptográficos más importantes utilizados por el TPM, son RSA [9], SHA1 [10] y HMAC [11].

2.3.2.2 Procesadores SafeXcel

SafeXcel [12] es una familia de procesadores de seguridad diseñados para funcionar como coprocesadores encargados de proporcionar funciones avanzadas de seguridad. Son procesadores pensados para ser integrados en dispositivos de red y de comunicaciones como gateways, routers, servidores NAS, etc. aunque también pueden ser incluidos en otros dispositivos como lectores de tarjetas inteligentes, o dispositivos de almacenamiento.

Entre las funcionalidades de seguridad que incorporan, pueden mencionarse:

- Soporte avanzado para Virtual Private Networks (VPN)
- Soporte hardware del protocolo IPSec
- Soporte criptográfico avanzado por hardware (algoritmos DES, 3DES, AES, SHA-1, MD5, HMAC)
- Funciones de encriptado y desencriptado
- Generación de números aleatorios

En definitiva es una solución de hardware diseñada para redes y para los diseñadores de chips que buscan implementar sistemas de alto rendimiento en prevención de intrusiones dentro de los procesadores de red.

2.3.2.3 IBM Common Cryptographic Architecture (CCA)

La arquitectura IBM CCA (Common Cryptographic Architecture [13]) es una arquitectura compuesta tanto por componentes software como hardware orientado a proporcionar un soporte de seguridad criptográfica completo para aplicaciones. La Figura 3 muestra esquemáticamente la arquitectura CCA:

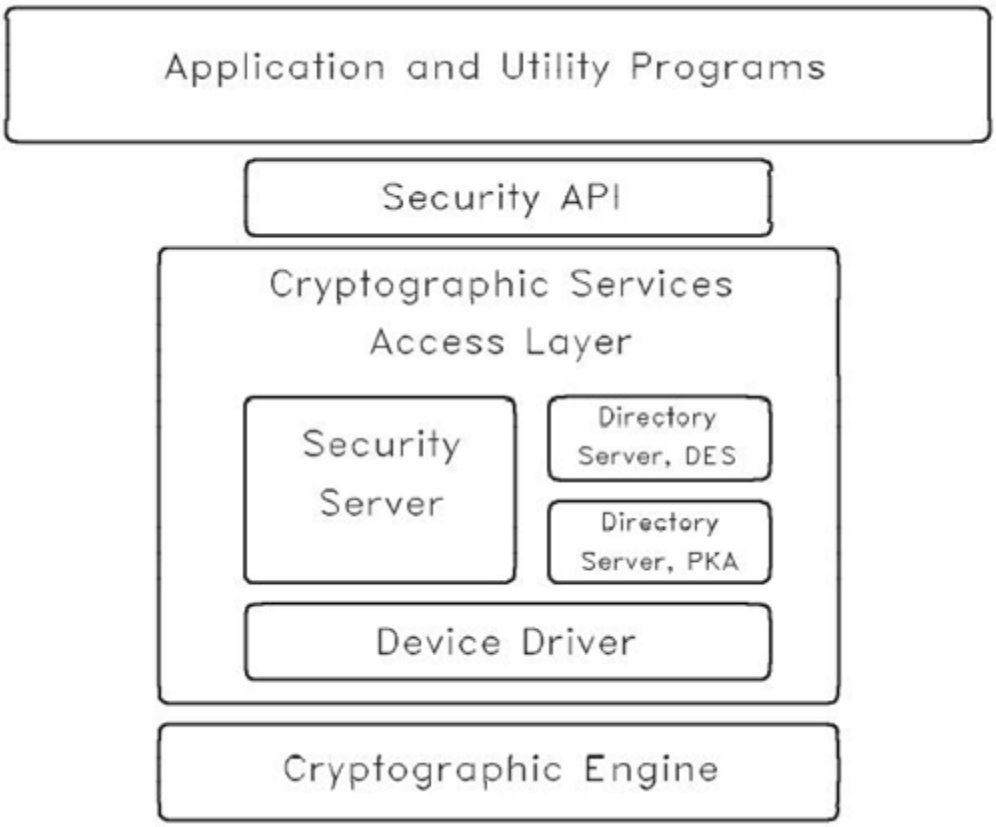


Figura 3 Arquitectura CCA

Las aplicaciones que requieren funcionalidades de seguridad utilizan una API de seguridad, que a su vez interactúa con los otros elementos software y con los componentes hardware que se encargan de proporcionar finalmente las funciones de seguridad. El framework proporciona también herramientas de ayuda al desarrollo de aplicaciones dentro de esta arquitectura.

La arquitectura IBM CCA define el conjunto de funciones criptográficas, interfaces externas y reglas de manejo de claves, que permiten la utilización de los algoritmos de encriptación estándar, tanto simétricos: DES; como asimétricos o de clave pública: PKI. Los componentes hardware implementan los distintos servicios criptográficos que definidos.

2.3.3 Componente confiable utilizado

En cuanto a componentes confiables queremos añadir que la opción que hemos visto más viable de todas las expuestas anteriormente es la del chip TPM, debido a que empieza a ser común en muchos portátiles. También comienza a existir una serie de APIs y elementos de desarrollo amplia, y creemos que es una buena opción de futuro. Actualmente no hay un uso generalizado de estos chips, pero confiamos en su evolución futura.

La idea del TPM fue desarrollada por el Trusted Computing Group (TCG). La TCG Software Stack Specification (TSS) especifica los elementos del TPM. Hay varias versiones de la TSS, actualmente se encuentra disponible la versión 1.2, que es la que usamos nosotros.

2.3.4 APIs existentes

Una vez seleccionado este chip como el más idóneo para nuestro escenario presentamos algunas de las principales APIs para el desarrollo con TPM.

2.3.4.1 La librería tpm4java

tpm4java [14] es una librería desarrollada por dos alumnos de la universidad técnica de Darmstadt, Alemania, Martin Hermanowski y Eric Tews. La librería, desarrollada en Java, permite acceder al TPM.tpm4java consta de tres capas.

- Una capa de alto nivel que proporciona acceso a la funcionalidad más común del TPM (La capa de alto nivel se define en el interfaz TssHighLevel).
- Una capa de bajo nivel que proporciona acceso a prácticamente toda la funcionalidad del TPM La capa de bajo nivel se define en el interfaz TssLowLevel).
- Un backend usado por la librería que envía los comandos al TPM.

La capa de alto nivel está implementada de tal manera que realiza su funcionalidad usando la capa de bajo nivel, por lo que todo lo que se puede hacer con la capa de alto nivel es también posible hacerlo usando sólo la capa de bajo nivel. Para accede a la capa de alto nivel el usuario debe usar el siguiente código:

```
TssHighLevel highLevel = TssFactory.getHighLevel();
```

Mientras que para usar la capa de bajo nivel debe usar este otro:

```
TssLowLevel lowLevel = TssFactory.getLowLevel();
```

2.3.4.2 TPM/J API basada en Java

TPM/J [15] es una API orientada a objetos que usa Java para acceder al TPM a bajo nivel. Fue desarrollada como parte de un proyecto de investigación en el MIT.

Esta API es intencionadamente no compatible con las especificaciones de TCG sobre TSS. Su principal intención es proveer de un API flexible orientada a objetos que sea más usable para programadores e investigadores que hagan experimentos de I+D en los casos donde las especificación de TCG no sean críticas.

Trata los comandos TPM de bajo nivel, y las estructuras de datos que responden a dichos comandos, como objetos Java. Esto permite a los programadores acceder fácilmente

a los campos de los comandos y de las estructuras de datos desde un punto de vista orientado a objetos, en vez de tener que leer bytes desde las salidas de los comandos en largas cadenas de bytes. También ofrece una implementación más modular gracias a que define cada comando y cada estructura de datos de respuesta como clases Java separadas.

TPM/J también posee clases de Java a alto nivel, que representan los conceptos de alto nivel y se construye como sesiones de autorización y sesiones de transporte. En el caso de las sesiones de autorización, por ejemplo, estas clases permiten un estado de sesión para ser guardado en un objeto de sesión. Los comandos TPM permiten cosas como encriptamiento e identificación de un modo más sistemático y modular.

Finalmente el acceso al TPM es abstraído en un objeto driver TPM. Proveyendo diferentes objetos drivers para diferentes plataformas, somos capaces de soportar múltiples plataformas.(Windows, Linux, Mac OS X y Vista) sin que el programador tenga que cambiar nada del código. Para nuestro conocimiento, TPM/J es una de las primeras API multiplataforma para TPM. Funciona bajo Linux, Windows, MAC OS X y Vista sin requerir diferentes versiones del software.

2.3.4.3 IAIK jTSS - TCG Software Stack para Java

El IAIK jTSS [16] es una implementación del TSS para Java. Todas las capas están implementadas en Java, no como en TrouSerS (ver siguiente sección) que sus capas están implementadas en C. Todavía se encuentra en las primeras fases de desarrollo y por tanto no está testado ni está completo.

Aunque todos los TPMs están basados en las especificaciones del TCG, algunos de ellos podrían comportarse de manera distinta al resto. Este software intenta abstraerse de esas pequeñas diferencias y proporcionar un comportamiento consistente independientemente del hardware TPM que haya debajo.

Una vez que IAIK jTSS esté implementado en Java portarlo entre sistemas operativos será relativamente fácil. Actualmente soporta los siguientes sistemas operativos:

- Linux (usando los drivers TPM de los núcleos recientes 2.6)
- Microsoft Windows Vista (usando Trusted Base Services)

Fijándonos en las características establecidas por el TCG para el TPM, IAIK jTSS cubre muchas partes de las especificaciones. Los TPMs versión 1.2 son completamente soportados si miramos sus cambios en la administración de recursos. Por tanto seria soportado para el TPM 1.2 que nosotros utilizamos.

2.3.4.4 API utilizada

La API que hemos escogido para su uso es la de tpm4java debido a que implementa todas las funciones del TPM y está bastante probada. También tiene un foro y unas listas de correo muy efectivas que proporcionan mucha ayuda con el desarrollo.

Por tanto después de toda esta presentación de distintas tecnologías escogemos el chip TPM y la API tpm4java para todo lo referente a la seguridad en el dispositivo.

2.4 Tecnologías de localización mediante redes inalámbricas

A pesar de las limitaciones e inconvenientes de las técnicas de localización en exteriores, éstas son suficientes para ofrecer numerosos servicios al usuario final. El reto es ahora la localización en espacios cerrados, lugares donde el GPS pierde su precisión debido a obstáculos como paredes y techos, y donde el error de los sistemas estudiados para redes celulares móviles tradicionales es inadmisible.

En la actualidad existen varias tecnologías que pueden utilizarse en interiores de edificios. Las más destacadas son Bluetooth, Wi-Fi (estándar 802.11b) y la denominada Banda Ultra-Ancha (Ultrawideband).

2.4.1 Bluetooth

Bluetooth es una tecnología diseñada para ofrecer conectividad a redes personales mediante un dispositivo móvil de forma económica. Permite conectar múltiples aparatos Bluetooth: ordenadores portátiles, PDAs, teléfonos móviles, etc., y ofrece conexión a una LAN o WAN a través de un punto de acceso. Bluetooth consigue un canal de comunicación de 721 kbps en un radio de acción de 10 metros, ampliable hasta 100 metros por medio de repetidores. La frecuencia que utiliza está entre 2,4 y 2,48 GHz, cuya gran ventaja es que es un rango de frecuencias abierto. Además, y debido a su concepción de tecnología móvil y económica, tiene un consumo de energía bajo. Para transmitir a una distancia de 10 metros emplea 1mW de potencia, mientras que para llegar a los 100 utiliza 100mW.

La aplicación práctica de esta tecnología es la posibilidad de montar redes inalámbricas en lugares donde haya dificultad para hacerlo de forma convencional, aunque hoy en día para este propósito se oferta otra tecnología, basada en los estándares del IEEE 802.11 que ofrece mayor ancho de banda y radio de conectividad. Por ello, Bluetooth se dirige más a la comunicación de dispositivos.

El sistema, denominado Red de Localización Bluetooth (Bluetooth Location Network, BLN), transmite información de la posición del terminal móvil a los servidores, sin la participación del usuario. No es objeto de restricciones debido a la pérdida de línea

de vista y funciona con dispositivos comerciales ya existentes (dispositivos con Bluetooth o terminales móviles de datos que admitan una tarjeta de expansión). El BLN está compuesto por pequeños nodos Bluetooth que establecen una topología de red espontánea con la inicialización del sistema. Puede coexistir con dispositivos Bluetooth que no son parte del sistema de localización, como impresoras y auriculares.

2.4.2 Wi-Fi

Hoy en día numerosos proveedores de redes wireless están instalando sus sistemas en hoteles, cafés, aeropuertos y otros edificios. Estas infraestructuras también soportan localización de dispositivos móviles, por lo que las aplicaciones basadas en la posición para entornos de área local resultan viables.

La localización mediante redes locales inalámbricas puede llevarse a cabo de diferentes maneras. La más sencilla es la basada únicamente en el punto de acceso más cercano al terminal. Este método confunde a menudo la planta del edificio, pues es fácil que la antena más cercana a un usuario ubicado en una determinada planta sea la misma que la correspondiente a un usuario situado en una planta superior, si la posición sobre el piso es similar. Por otra parte la señal es vulnerable debido a las interferencias, lo que puede afectar, además de a la precisión, a la seguridad de la comunicación.

Existe otra propuesta muy interesante, realizada por Ekahau, compañía que comercializa un motor de posicionamiento [17] basada en el almacenamiento de medida de potencia de señal en diferentes puntos del recinto cubierto. La técnica, conocida como Wi-Fi mapping, arroja resultados más exactos que los métodos de triangulación celular, ofreciendo una precisión de 1 a 20 metros. Además, este sistema es sensible a los cambios de altura, es decir, reconoce fácilmente la planta del edificio en la que está el usuario. El proceso de entrenamiento del motor de posicionamiento Ekahau es el siguiente:

- Creación de un modelo de posicionamiento: consiste en dibujar sobre un plano del área a cubrir los puntos donde se deben tomar las medidas. Existe una herramienta de apoyo que facilita la realización de esta frase.
- Calibración del modelo de posicionamiento: se trata de recorrer el área elegida grabando muestras de la potencia de la señal en cada putno marcado en el apartado anterior. No se necesita información acerca de la localización de los puntos de acceso.
- Comienzo del seguimiento de dispositivos: mediante un gestor que ofrece la misma compañía se procede a controlar la posición de los dispositivos móviles.
- Análisis de la precisión: es posible grabar algunos datos más de prueba y analizar visualmente los vectores de error de posición y las estadísticas para encontrar áreas donde se necesitan puntos adicionales de acceso o más muestras de calibración.

2.4.3 Banda ultra-ancha

Ultrawideband (UWB [18]) es una tecnología que nació durante la década de 1960, y cuyo nombre fue acuñado por el Departamento de Defensa de los Estados Unidos en 1989. Se desarrolló para radar, localización y aplicaciones de comunicaciones. La capacidad de UWB de operar por debajo del nivel de ruido evitaba que las comunicaciones seguras pudieran ser interceptadas.

UWB emplea ráfagas de potencia mil veces más baja que las de un teléfono móvil, con duración de picosegundos, en un espectro de frecuencia amplio (3.1-10.6GHz). Estas ráfagas ocupan uno o unos pocos ciclos de portadora RF, por lo que la señal resultante tiene un ancho de banda grande. Sobre las ráfagas es posible transferir datos a velocidades de centenares de megabits por segundo. Además, la señal es relativamente inmune a la cancelación multitrayecto, ya que debido a su corta duración la señal directa va y vuelve antes de que las señales reflejadas en los obstáculos alcancen el receptor. Por ello, y porque es un sistema de baja complejidad y coste reducido, UWB resulta especialmente adecuada para aplicaciones móviles sin hilos. Puede asimismo solucionar los problemas de precisión y de seguridad de los que adolece la red Wi-Fi.

Debido a sus características, esta tecnología permite localizar los terminales móviles con un error insignificante. UWB está basada en pulsos ultracortos, de tal manera que el receptor puede determinar el tiempo de llegada con precisión de picosegundos y, por tanto, estimar la posición con precisión de centímetros. La distancia al móvil se calcula midiendo el retardo de un pulso desde que es emitido por el transmisor hasta que llega al receptor. Posteriormente, utilizando triangulación se determina con gran exactitud la posición del terminal. Si se realizan las medidas respecto a cuatro receptores diferentes, es posible saber con precisión la altura a la que está el usuario.

La Tabla 1 muestra una comparativa resumida de algunas tecnologías inalámbricas. Como se puede observar, UWB tiene una capacidad muy superior al resto, con una potencia de emisión insignificante, pero está limitada en alcance.

Tecnología	Tasa de datos (Mb/s)	Potencia (mW)	Alcance(m)	Frecuencia
Bluetooth	1-2	100	100	2.4GHz
IrDA	4	100mW/sr	1-2	Infrarrojo
UWB	100-500	1	10	3.1-10.6GHz
IEEE 802.11a	54	40-800	20	5GHz
IEEE 802.11b	11	200	100	2.4GHz
IEEE 802.11g	54	65	50	2.4GHz

Tabla 1 Comparativa de tecnologías

2.4.4 Conclusiones

Después de la presentación de los métodos de localización hemos decidido la utilización de tecnologías basadas en Wi-Fi por la disponibilidad de red. No habría que extender una nueva red para localizar a los dispositivos y eso nos facilita mucho la implantación en un escenario real.

De entre las soluciones basadas en Wi-Fi hemos decidido utilizar Ekahau porque era una tecnología que ya conocíamos y nos resultaba más sencillo utilizarla. La API era sencilla de utilizar y por tanto en un principio la veíamos más tangible que las demás.

2.5 Herramientas de localización semejantes en el mercado

2.5.1 Introducción

La rápida evolución de nuevas tecnologías de comunicación y en concreto aquellas que ofrecen comunicación a usuarios móviles de forma inalámbrica está llevando a un rápido desarrollo y despliegue de redes WLAN en múltiples entornos. Este amplio despliegue, sin embargo, presenta un importante inconveniente: la necesidad de mejorar la seguridad en la red, tanto en identificación como en acceso de usuarios y puntos de acceso.

Son varias las empresas que ofrecen soluciones de localización y aplicaciones basadas en Wi-Fi y WLAN, así como la posibilidad de utilizar los datos de localización para aplicaciones propias del usuario que se comuniquen con la aplicación de localización. El objetivo de esta sección es resumir las características básicas de algunas de las soluciones encontradas y analizar sus ventajas y desventajas, así como los puntos oscuros en la información presentada por la empresa que oferta la solución.

2.5.2 Wi-Fi WatchDog, de Newbury Networks

La compañía Newbury Networks [19] ofrece diversas soluciones, tanto a clientes finales como a desarrolladores, para mejorar la seguridad y gestión en redes WLAN, todo ello mediante la utilización de un sistema de localización patentado por Newbury, basado en triangulación. La fiabilidad y especialización de esta empresa en sistemas de localización WLAN hacen interesante el estudio de las soluciones ofrecidas por la misma que puedan ser aplicables a nuestro caso.

Los productos ofrecidos por Newbury Networks, todos ellos en el área de localización, se pueden agrupar en dos tipos:

- Soluciones finales, para que el cliente las utilice directamente sobre su red WLAN para localización. Son tres los productos ofrecidos:

- **Active Asset**: localización y seguimiento de dispositivos valiosos, mediante el uso de tags.
- **WiFi WorkPlace:** monitorización y gestión de red WLAN utilizando localización.
- **WiFi Watchdog**: detección, monitorización y securización de usuarios en redes WLAN, basado en localización.

- Soluciones para desarrolladores, que incluyen una aplicación para localización que ofrece datos de la posición de usuarios WiFi que pueden ser utilizadas por otras posibles aplicaciones que se deseen desarrollar. Dos productos de Newbury Networks se pueden considerar en este grupo:
  - **Newbury Presence Platform**: plataforma de desarrollo de SW basado en información de localización ofrecida por el software de Newbury.
  - **Newbury Location Appliance**: dispositivo tipo servidor que contiene el SW de localización, y permite añadir nuevas aplicaciones que utilicen la información de localización obtenida por el software de Newbury.

En los siguientes apartados del documento nos centraremos en el producto WiFi Watchdog, por ser el que más se acerca a la aplicación que se desea realizar y los casos de estudio.

2.5.2.1 Motivación

WiFi Watchdog es una tecnología que ofrece un grado más de seguridad en redes basadas en los estándares 802.11 para WLAN. Para ello se basa en la información de la posición de los usuarios y puntos de acceso de la red, de modo que se puede utilizar dicha información para discriminar posibles usuarios maliciosos que se encuentren en el exterior del edificio o zona a cubrir con la red WiFi. Además, WiFi Watchdog permite descubrir posibles puntos de acceso (AP) que no pertenezcan a la red corporativa y pretendan hacerse pasar por uno de ellos para obtener información de los usuarios de la red. Con todo ello, WiFi Watchdog pretende incrementar la seguridad de la red WiFi de la compañía, tanto desde el punto de vista de la red como desde el punto de vista del usuario.

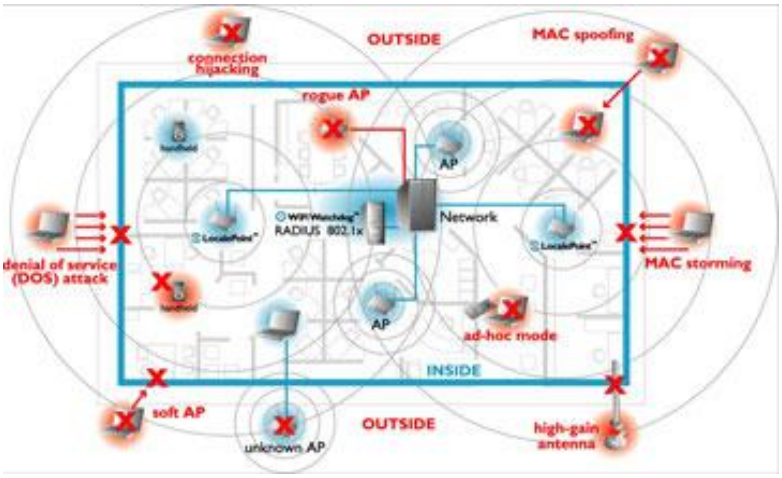


Figura 4 Casos de aplicación de Wi-Fi Watchdog

2.5.2.2 Aspectos técnicos

Este producto se basa en la utilización de sensores (usados a modo de receptores) y un software que calcula la posición de los usuarios y APs de la red mediante técnicas de triangulación. Los sensores son puntos de acceso de tipo 802.11 pasivos (en concreto APs de Cisco modificados) que recogen información de la señal de RF transmitida por los APs y usuarios en el sistema para calcular la posición por triangulación.

La precisión ofrecida con este sistema es de nivel de "zona": se discrimina a nivel de pisos y habitaciones (no se puede conocer la posición exacta dentro de una habitación, por ejemplo). Newbury indica que la posición se obtiene con una precisión de "unos pocos pies". La prueba del sistema realizada por Network Word Lab Alliance dio como resultado una precisión de unos 5 pies (unos 1.5 m).

El terminal de usuario (UT) no requiere de la instalación de ningún software (SW) o hardware (HW) adicional, lo que simplifica el despliegue de la tecnología WiFi Watchdog. Esto se logra gracias a que WiFi Watchdog basa la detección y localización únicamente en la señal radio enviada por el usuario o el AP que se desea localizar.

La localización se logra en tiempo real, de modo que la posición de cada elemento en la red se actualiza continuamente.

WiFi Watchdog está preparado para ser integrado con varios sistemas WiFi, en concreto los fabricantes soportados son: Cisco, Araba, Symbol y Trapeze. Cisco ofrece una mayor capacidad de integración, ya que se ha desarrollado una solución conjunta con este fabricante.

La solución ofrecida con WiFi Watchdog no incluye el uso de etiquetas (o tags) para la localización de dispositivos. Sin embargo, Newbury Networks también trabaja con esta tecnología de localización (por ejemplo en el producto Active Asset).

Todos los estándares actuales 802.11 (a, b y g) para WLAN son soportados.

2.5.2.3 Aplicaciones SW ofrecidas

La solución de localización y seguridad WiFi Watchdog incluye dos aplicaciones SW:

- Real Time Site Survey: permite monitorización en tiempo real del entorno bajo estudio, altas-bajas de UTs y APs y agrupación de dispositivos según el tipo. Para cada elemento de la red ofrece información de tres tipos: quién (autorizado/no autorizado), qué (tipo de dispositivo) y dónde (localización). Se pueden generar automáticamente informes con datos del historial de la red. Además, se permite incluir alertas en función de que se produzcan o no ciertos eventos (localización de un usuario en cierta zona, etc).
- RADIUS Server: ofrece servicios extra a la administración de la red WLAN, para permitir o no acceso en función de posición y dirección MAC del dispositivo. Además, permite la creación de "subredes" virtuales en función de la posición (ej, salas de reuniones, despachos, hall…).

2.5.2.4 Ventajas e inconvenientes

Debido a la característica de funcionamiento con la red WiFi, inherente del sistema, sólo dispositivos "activos" (que transmitan una señal de tipo WiFi) son detectados, lo que puede ser un inconveniente si la herramienta se quiere utilizar para controlar otros elementos (por ejemplo, personas o equipos apagados). Este problema se podría solventar incorporando otros dispositivos de detección, como etiquetas. Por otro lado, esta misma característica se puede ver como una ventaja, ya que el sistema no requiere de HW extra en los terminales de usuario o puntos de acceso preexistentes.

Como ventaja frente a otras soluciones, WiFi Watchdog ofrece una gran variedad de posibilidades en control de acceso para evitar intrusismo en la red (detección de múltiples tipos de ataques, alertas por localización en zonas prohibidas…), incluidas en el propio WiFi Watchdog. Sin embargo, la integración con posibles aplicaciones de terceros parece bastante complejo, y el desarrollo de aplicaciones que utilicen información de localización obtenida por WiFi Watchdog puede resultar difícil: WiFi Watchdog es una solución final para cliente, no para desarrolladores.

La solución requiere de una etapa de calibración, lo que implica un cierto tiempo para ponerlo en funcionamiento.

2.5.2.5 Resumen de características

Las características básicas de WiFi Watchdog se resumen en la siguiente lista:

- Reconocimiento tanto de la posición de terminales de usuarios como de puntos de acceso.
- Precisión: reconoce pisos, habitaciones y zonas (si bien la compañía que lo comercializa asegura que tiene una buena precisión en localización por coordenadas). Aproximada mente, 1.5 m.
- Permite crear alertas (de usuarios que se encuentran en zonas no permitidas, etc).
- Control de dos tipos de elementos en la red:
    - accesos indebidos de usuarios → control de acceso.
    - APs que actúan como pertenecientes a la organización y no lo son → ofrece mayor protección a los usuarios.
- Permite integración con herramientas de gestión estándar.
- Permite despliegue rápido y sencillo en redes WLAN ya instaladas, trabajando con múltiples proveedores. Además, no requiere de instalación de SW o HW extra en los equipos de usuario.
- Se soportan todos los estándares 802.11.

2.5.3 Wi-Fi Positioning System, de Skyhook Wireless

La compañía Skyhook Wireless [20] ha apostado por ofrecer a sus usuarios servicios de localización basados en rede Wi-Fi, en competencia con el sistema GPS ya implantado. La idea básica aquí es aprovechar las cada vez más numerosas redes WiFi y los equipos preparados para comunicarse a través de esta tecnología (como la gran mayoría de los portátiles y PDAs) para dar un servicio de localización de forma sencilla, rápida y barata.

El producto básico ofrecido por Skyhook Wireless se denomina WiFi Positioning System. Sus principales características se resumen a continuación.

2.5.3.1 Características básicas

WiFi Posistioning System (WPS) es un sistema de localización basado en el uso de señales WiFi para calcular de forma precisa la posición de equipos que transmitan y reciban este tipo de señal. En principio el sistema no está pensado como solución para añadir un grado más de seguridad o control de acceso, sino para ofrecer servicios de localización directamente al usuario, por lo que su aplicación en el proyecto que nos ocupa no es directa.

El uso de señales WiFi para localizar equipos ofrece diversas ventajas frente a los sistemas de localización convencionales, basados en señal GPS. En primer lugar, la mayor

parte de los terminales de usuario tipo portátiles y PDAs están dotados de trasnceptores radio para WiFi, pero pocos soportan comunicación GPS, por lo que la disponibilidad de esta tecnología para terminales de usuario es mucho menor. Además, el sistema GPS ofrece buenas prestaciones en entornos de exteriores, pero no es adecuado para entornos de interiores, donde la señal GPS apenas tiene cobertura. Los APs de redes WiFi, sin embargo, se pueden encontrar directamente en el interior del edificio, y su localización cercana al UE permite su buena recepción en entornos de interiores. Por último, los servicios basados en GPS requieren de un tiempo de cálculo de posición bastante grande, ya que es necesario comunicarse con los satélites "visibles" por el UE. Sin embargo, la localización basada en señales WiFi permite localizar equipos en menos de 1 segundo.

Según los puntos anteriores, el sistema WPS se presenta como una buena alternativa a los sistemas GPS de localización para el usuario final. Por otro lado, se observa que el enfoque es algo diferente a la idea presentada en previas tecnologías, ya que no se trata de dotar de seguridad o control de acceso a un servicio.

2.5.3.2 Aspectos técnicos

WPS basa el cálculo de posición de los equipos en el procesado de la señal WiFi detectada procedente de varios APs que se encuentran al alcance del UE. Para lograr una buena cobertura, Skyhook Wireless ha desplegado una red de puntos de acceso en varias ciudades estadounidenses. El equipo de usuario debe instalarse un software cliente para obtener la posición a partir de las señales WiFi detectadas. Existen dos posibles configuraciones, en función de la carga computacional y la memoria de que se disponga en el UE:

- Device centric: todo el SW necesario se encuentra en el UE (algoritmo de cálculo de posición y base de datos de localización de los APs). El propio terminal, por tanto, calcula su posición a partir de la base de datos de APs conocidos. Esta opción es la mejor para dispositivos que no necesariamente están conectados a la red (p.e. PDAs), ya que no requiere conectividad a la red.
- Network centric: la base de datos se encuentra en la red, de modo que sólo el cliente de localización se ejecuta en el UE. El servidor de localización facilita la posición al cliente a través de una red de datos (por ejemplo a través de WLAN). Este modo de operación es adecuado para equipos que se pueden asumir conectados a una red (p. e. un portátil).

El sistema WPS es adecuado para localización en interiores y áreas metropolitanas, ya que no requiere de visión directa entre UE y APs.

La cobertura ofrecida por el sistema no es global, ya que depende directamente de que existan APs en la base de datos, en la zona donde se encuentra en UE. A fecha de 7 de Noviembre de 2008, la red de localización de Skyhook Wireless cubría áreas que

representaban el 65% de la población en USA, y planean extenderse al 70% y a la mayoría de la población canadiense a finales de año. Sin embargo, aún no parece que hayan empezado a dar servicio en Europa. Se puede ver las áreas de cobertura en la Figura 5.

Figura 5 Cobertura actual del sistema WPS

La precisión ofrecida por esta tecnología es de aproximadamente 20 m, según la compañía que lo ofrece. Por tanto podemos ver que no es una tecnología muy precisa en comparación con otras anteriormente presentadas.

La integración del sistema con aplicaciones de terceros parece bastante directa, ya que WPS incluye varias APIs y un sistema de desarrollo (SDK) para utilizar la información de localización ofrecida por WPS en otras aplicaciones que estén basadas en este tipo de información. Un ejemplo interesante es el uso de información de localización integrado en Skype, la aplicación de voz sobre IP y mensajería instantánea mundialmente conocida; esta solución conjunta ha dado lugar a Skype E911 Plug-in (Beta), con información de dónde se encuentra el usuario con el que estamos hablando o chateando (ver Figura 6).

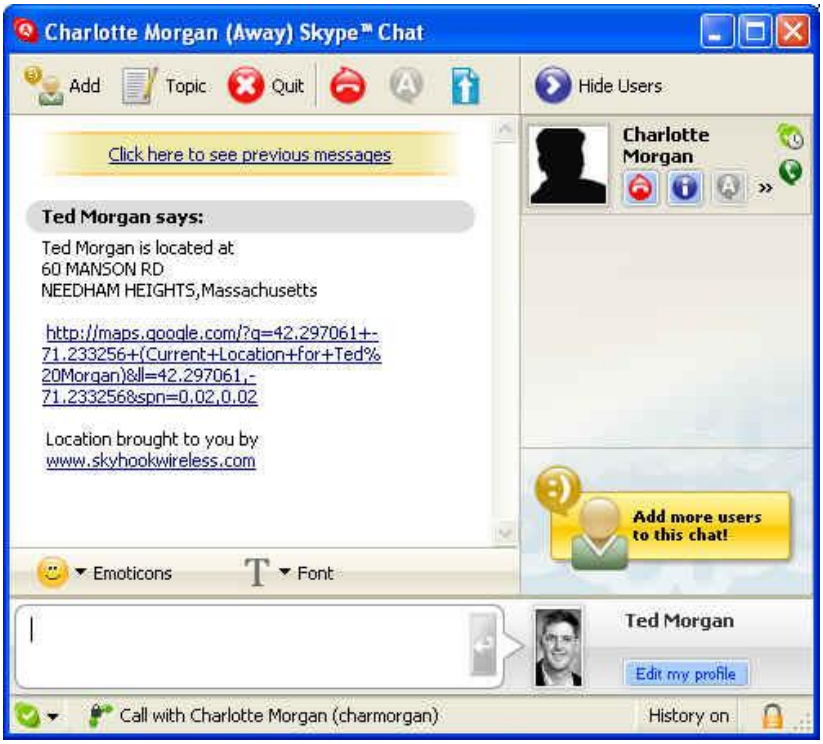


Figura 6 Aplicación Skype E911, con información de localización

2.5.3.3 Ventajas e inconvenientes

La principal ventaja de esta tecnología frente a otros sistemas es que no requiere de ningún HW adicional, bastando con poseer en el UE de una tarjeta WiFi. Como en el caso de otras tecnologías, esta ventaja puede a su vez volverse desventaja cuando se trata de localizar dispositivos que no se encuentran activos. En este caso no bastaría con utilizar tarjetas activas que transmitan señal WiFi, ya que además es necesario ejecutar un software cliente en el terminal de usuario para, como mínimo, capturar la información sobre qué APs son visibles y el nivel de señal recibida.

El sistema es bastante sencillo, y el SW que se debe ejecutar tiene pocos requisitos, lo que simplifica el uso de esta tecnología para localización. Además, no requiere de etapa calibración. Sin embargo, no está enfocada a la localización centralizada de múltiples dispositivos en una red, sino más bien a dar servicios a un usuario final, lo que la hace menos adecuada a la aplicación aquí perseguida.

Como ventaja frente a otras tecnologías presentadas, WPS ofrece bastantes facilidades para poderse integrar con aplicaciones de terceros, gracias al SDK con diversas APIs.

2.5.3.4 Resumen de características

Las características básicas de WiFi Watchdog se resumen en la siguiente lista:

- Tecnología basada en señal WiFi ofrecida por una serie de APs cuya posición es conocida (se encuentran en una base de datos).
- Entornos de aplicación: Indoor-outdoor, indoor.
- No requiere ningún HW extra, es todo basado en SW.
- Facilidad de integración con otras aplicaciones, gracias a APIs (SDK) para desarrolladores.
- Sólo ofrece cobertura en Estados Unidos (65% de la población) y próximamente en Canadá.
- Precisión aproximada: 20 m.
- No se especifica si funciona para todos los estándares 802.11 (es presumible que sí).

2.5.4 Conclusiones

Se han estudiado dos soluciones de localización presentes en el mercado actual, si bien existen otras muchas opciones con mayor o menor aplicabilidad al caso que nos interesa. En general, se ha observado que las soluciones de localización ofrecidas son cada vez más y con mayores posibilidades.

En cuanto a las soluciones analizadas, cabe decir que la primera de ellas (WatchDog de Newbury Networks) presenta un gran número de opciones en cuanto a la seguridad ofrecida al cliente, permitiendo localizar no sólo equipos de usuario sino también posibles puntos de acceso falsos. El software ofrecido es muy completo y con múltiples opciones (alertas, control por zonas…), pero parece poco abierto, lo que lo hace algo difícil de integrar con otras aplicaciones de terceros. Una opción interesante sería estudiar la solución, también de Newbury, llamada *Location Appliance*, pensada para desarrolladores.

La segunda solución analizada (WiFi Positioning System, de Skyhook Wireless), si bien presenta una idea interesante al tratar la localización como un servicio para el cliente, basado en dar cobertura utilizando puntos de acceso WiFi en vez de señal GPS, no es aplicable a nuestro caso, ya que sólo se puede utilizar en zonas en las que exista cobertura de Skyhook, que por el momento se restringe a Estados Unidos.

Entonces sólo nos quedaría compararnos con la primera opción con la que podemos establecer que nuestro sistema es más barato, open source, y posee mucha versatilidad. A favor de su opción parece que está bastante probada y tiene el respaldo de una compañía. Nosotros preferimos nuestro modelo de seguridad basado en tres factores independientes que conjuntamente nos dan mucha más fiabilidad.

2.6 Relación con otros proyectos

El proyecto SERENITY se encuentra relacionado con otros proyectos, entre los que se encuentran los que se indican a continuación:

2.6.1 BUGYO

Es un proyecto [21] CELTIC que define un marco de seguridad e implementa una maqueta para medir, documentar y mantener los niveles de garantía de seguridad (assurance) de servicios basados en sistemas de telecomunicaciones.

El marco de seguridad proporciona guías, métodos así como aplicaciones software, que aseguran toda la seguridad que puedan. Gracias a este marco el proyecto desarrolla un sistema de ayuda a mantener un cierto nivel de garantía de seguridad en todos los niveles. Este sistema tiene certificaciones y elementos que permiten comprobar su garantía de seguridad.

2.6.2 SEGUR@

Es un proyecto [22] CENIT que tiene como objetivo generar un marco de confianza y seguridad para el uso de las TICs en la e-Sociedad. Partiendo de la situación actual, se realizará un amplio trabajo de investigación en tecnologías básicas aplicables a la seguridad de la información, tales como biometría, algoritmos criptográficos, algoritmos avanzados de detección de intrusiones, etc. Se complementará con la definición de arquitecturas y protocolos que permitan la aplicación de estas tecnologías básicas a la consecución del objetivo del proyecto.

El problema de la seguridad y la confianza en la Sociedad de la Información es un problema muy complejo que necesita un enfoque necesariamente multifactorial. Segur@ cubre un amplio abanico de ámbitos y tecnologías. Los principales objeticos de este proyecto son los siguientes:

- Explorar y desarrollar el concepto de “Redes confiables”, donde cada elemento de la red es capaz de comunicar a otro su nivel de seguridad; es decir, el grado de confianza que es capaz de ofrecer, permitiendo que la red le configure un determinado nivel de servicio en función de ello para garantizar la seguridad del resto.
- Identidad digital robusta y al alcance de todos. Se pretende el desarrollo de soluciones integrales de garantía de la identidad y la privacidad de implantación solidaria, de forma que se mantengan los niveles de confianza independientemente de las entidades que las desplieguen. Se desea que estas soluciones estén guiadas por dos vocaciones: El “propósito general”, lo que les debería permitir ser implantadas para proteger cualquier tipo de servicio; y el “amplio estrato de

difusión", con el fin de poder llegar a cualquier actor y dotarle de capacidad para eliminar el fraude de identidad y el menoscabo de la privacidad, de acuerdo a su percepción de la sensibilidad del servicio que desea proteger y los recursos que tiene para ello.

- Sistemas de monitorización, detección y resolución temprana de problemas y amenazas en la red basados en actuaciones colaborativas entre todos los elementos y entidades presentes en ella. Las soluciones aportadas en este ámbito propondrán mecanismos específicos e integrales para la búsqueda de amenazas en la red y la implicación de todos los actores a la hora de su resolución.

2.6.3 €-Confidentia (EUREKA/ITEA)

Es un programa europeo de investigación [23] cuyo objetivo es desarrollar una plataforma que permita asegurar aplicaciones y ofrecer un entorno seguro de ejecución para dispositivos móviles y ordenadores.

El objetivo fundamental del proyecto €-Confidentia es establecer una plataforma segura y fiable para aplicaciones multiservicio que protejan a los usuarios finales de los diversos peligros que se pueden encontrar en el ciberespacio. La plataforma es aplicable a múltiples plataformas móviles como teléfonos móviles o portátiles.

2.6.4 mVIA

Es un proyecto [24] que resuelve de manera transparente para el usuario la conectividad del vehículo tanto a Internet, como con otros vehículos o infraestructura, con independencia del medio para realizarlo (wi-fi, wimax, umts, etc.).

m:Vía pretende ampliar los objetivos desde la perspectiva de las comunicaciones utilizando las redes y tecnologías móviles o inalámbricas, para seguridad vial, aportando valor añadido como información, entretenimiento, soporte a vehículos, confort y optimización del tráfico. En definitiva, m:vía quiere conseguir tecnología y conocimiento para poder mejorar el transporte por carretera, convirtiendo las vías y los vehículos en un entorno inteligente.

Figura 7 Proyecto m:Via

m:Vía quiere conseguir tecnología y conocimiento para poder mejorar el transporte por carretera, convirtiendo las vías y los vehículos en un entorno inteligente

2.6.5 CVIS (Cooperative vehicle infrastructure Systems)

Es un proyecto europeo [25] de cooperación entre vehículos. Trata sobre las tecnologías necesarias para comunicar vehículos entre sí (C2C), así como con las redes externas (C2R y C2W). Se obtiene también información relevante sobre el guiado y la ruta optima al destino en tiempo real.

Es un proyecto nuevo que no está muy desarrollado de momento, pero que puede tener una gran repercusión en un futuro no muy lejano.

2.6.6 Conclusión

Una vez presentados todos estos proyectos podemos decir que nuestra propuesta cubre un campo más o menos específico que no se ha visto en ninguno de los proyectos mencionados en esta sección.

Por tanto podemos establecer que nuestro proyecto puede aportar nuevas condiciones de seguridad en escenarios AmI. Nosotros creemos que la investigación en este tipo de escenario es muy importante y representa una oportunidad de poder modelar escenarios que se comporten de una manera casi real, como es nuestro caso.

En definitiva vemos con claridad como nuestro proyecto fin de carrera, y en mayor medida el proyecto europeo SERENITY serán considerablemente importantes en cuanto a escenarios AmI nos referimos.

3 Diseño

El entorno general en el que este proyecto tiene lugar es en un escenario de comunicaciones inalámbricas. Los requerimientos de seguridad varían frecuentemente en este escenario en donde las variaciones de los usuarios y los dispositivos son comunes. Las redes inalámbricas son cada vez más rápidas, especialmente en interiores de empresas, en edificios oficiales, en aeropuertos… como una alternativa mucho más flexibles que las redes clásicas de cable. En este entorno las diferentes zonas determinan necesidades de seguridad distintas dependiendo de diversos factores. La red que usamos en este escenario es Wi-Fi.

En una compañía la seguridad varía dependiendo de donde nos encontremos, así en la cafetería o en el hall de entrada no se podrá acceder a los registros internos de la compañía. Como ya hemos mencionado en el estado del arte (apartado 2.4, Tecnologías de localización) hay diversos métodos para localizar dispositivos. En nuestro caso utilizaremos el Ekahau Positioning Engine.

Los dispositivos que nos disponemos a localizar son dispositivos portables con adaptador Wi-Fi, PDAs teléfonos móviles, portátiles…

3.1 Motivación

Con este escenario de comunicación pretendemos mostrar los retos de la seguridad en un entorno Wi-Fi con dispositivos móviles.

Como ya hemos mencionado, la vulnerabilidad de las redes inalámbricas es mucho mayor que la de las redes convencionales. Para garantizar la seguridad en este entorno debemos poner requerimientos de seguridad muy fuertes para poder proteger adecuadamente la red y dar a los usuarios un entorno seguro. Cualquier entorno que requiera de la actuación de los usuarios para garantizar la seguridad en todo el escenario será muy vulnerable en cuanto a seguridad, ya que los usuarios suelen ser reacios a la instalación de software en sus máquinas. Esto podría desembocar en un entorno inseguro para todos los usuarios. Por tanto en este punto entroncamos con los escenarios AmI, los cuales ya hemos descrito en el estado del arte, ver sección 2.2. Normalmente el usuario final debe ser considerado un punto débil en cuanto a seguridad se refiere.

La arquitectura y los conceptos de patrones propuestos por SERENITY son apropiados para este escenario.

3.2 Descripción del escenario

Este escenario consiste en una red Wi-Fi interior en un edificio o una oficina. Nos centramos en redes que requieran un cierto nivel de seguridad, donde ciertos recursos o

cierta información sólo son accesibles para algunos usuarios. Las redes Wi-Fi están experimentando una evolución muy grande y están apareciendo numerosos ejemplos del escenario que nosotros estamos proponiendo, ya sea edificios oficiales, embajadas, hospitales… Debido a este potencial de aplicación exponemos la necesidad de utilizar los conceptos SERENITY.

Suponemos que cuando se instala una red inalámbrica el objetivo que se está buscando es ofrecer un método de comunicación más flexible, independiente de la localización. Esta red permite al usuario acceder a múltiples recursos remotos y a diversa información en cualquier momento y en cualquier lugar donde exista cobertura de red. El usuario puede cambiar de localización sin la necesidad de estar conectando y desconectando un cable de red. Lo malo de esta flexibilidad es que se puede volver contra la propia red, ya que no es capaz de diferenciar las zonas desde las que se conectan los usuarios y los distintos niveles de seguridad que sería aconsejable que están zonas poseyeran.

La localización puede ser utilizada como un parámetro para mejorar la seguridad en las redes Wi-Fi, poniendo distintos niveles de seguridad en zonas diferentes. Por tanto los usuarios están localizados en áreas distintas, las cuales tienen asignado un nivel de seguridad dependiendo de varios factores. Los usuarios podrían cambiar de área en tiempo real, y nuestro sistema debería ser capaz de reconocer estos cambios y tomar medidas para garantizar la seguridad de las comunicaciones y de los accesos.

Hasta ahora sólo hemos estado valorando la opción de cambio de localización sin mencionar en ningún caso la seguridad del dispositivo, y la identidad del usuario que intenta acceder a través de éste dispositivo. Por tanto ahora continuamos con la definición del escenario agrupando todos los requerimientos.

A partir de ahora tenemos en cuenta los tres factores de seguridad que se tienen en cuenta:

- Localización: usando la información proporcionada por un Sistema de Localización Interior (ILS, Indoor Location System, por sus siglas en inglés), en nuestro caso utilizamos Ekahau.
- Identidad del usuario: los usuarios deben autenticarse para poder iniciar sesión, ya sea como visitantes o como empleados.
- Identidad de dispositivo: cada unidad puede identificarse a sí misma. En nuestro caso esta identificación se produce vía TPM (ver apartado XX), garantizando la seguridad del dispositivo.

3.3 Lista de requisitos S&D del escenario

Los requisitos que aplican a todo el sistema son los siguientes:

- Una red Wi-Fi proporcionada por la compañía A debería estar disponible desde el lugar donde los empleados deseen conectarse.
- Los equipos de los empleados deben tener conectividad Wi-Fi, esto es, los dispositivos de los empleados deben ser capaces de conectarse a la red Wi-Fi de la compañía A.
- Debe haber un sistema de localización interior, ILS, instalado en la compañía A. Debe también abarcar toda la superficie que ocupa la empresa. En nuestro caso este sistema va sobre la red Wi-Fi, por tanto no necesita una instalación previa.

Estos son los requisitos S&D (Secure & Dependability) que hemos tenido en cuenta para el escenario:

- La información enviada a través de la red Wi-Fi debe estar apropiadamente asegurada para evitar que pueda ser captada por terceros interesados.
- El sistema ILS debe estar disponible como poco durante las horas en que la red Wi-Fi está operativa en la compañía. Además debe estar garantizado el funcionamiento del servidor ILS.
- El servidor ILS debe proporcionar información con, al menos, un cierto nivel de acierto para el área restringida. Este nivel debe poder ser configurado en el sistema para poder garantizar un cierto nivel de confianza para la información de localización.
- Debe haber una política de seguridad definida por el usuario, la localización y el dispositivo, y debe estar guardada en un repositorio o en un servidor. Este repositorio debería estar apropiadamente asegurado para evitar que accedan usuarios no permitidos. De hecho solamente la Aplicación de Control de Seguridad debería tener acceso al repositorio de política de seguridad.
- El equipamiento de acceso debe tener el hardware y el software requerido para permitir que sea localizado e identificado. Este hardware/software debe estar protegido para evitar intentos de fraude.
- El HW/SW del dispositivo del usuario necesario para la localización y la identificación debería ser proporcionado por la compañía de un modo seguro, para que tanto la empresa como el usuario puedan confiar en él.
- El equipamiento de los empleados debe tener un hardware y/o un software que lo identifiquen unívocamente. La información provista debe ser tangible y segura. Por tanto la identificación del dispositivo debe garantizar que el dispositivo es realmente el que dice ser. El equipamiento de los visitantes podría tener o no el hardware/software apropiado.

- El adaptador Wi-Fi en el equipo del usuario debería soportar el protocolo estándar de seguridad que viene definido en SERENITY.
- Los recursos a los que se necesita acceder deben estar disponibles y operativos.
- La Aplicación de Control debe controlar el acceso a los recursos protegidos.
- En el caso de que se trate de información, ésta debe estar adecuadamente archivada para garantizar su confidencialidad y su integridad.
- El terminal del empleado debe disponer de las capacidades para permitir la autenticación del usuario dependiendo de su nivel de seguridad, como está definido por la Aplicación de Control de Seguridad y por SERENITY.
- El empleado debe ser previamente registrado en la Aplicación de Control de Seguridad, así como el dispositivo. Así aseguramos una correcta identificación y autenticación.
- Debe haber reglas definidas para la identidad de los empleados y su perfil. Estas reglas deben ser accesibles en la Aplicación de Control de Seguridad. EL perfil de usuario está relacionado con los usuarios en bas al nivel de seguridad que tienen cada uno de ellos, en nuestro caso definimos tres niveles: visitante, becario y trabajador.
- El visitante debe poseer las credenciales generales para conectarse como invitado de la compañía. La compañía debe garantizar que esta información es fiable.
- El ILS debe ser capaz de detectar que un usuario desautorizado no está dentro del área que abarca el escenario.
- Debe haber un nivel de seguridad asociado a las localizaciones de fuera del escenario. La información de estas zonas debe estar almacenada en el repositorio o en un servidor seguro.

3.4 Arquitectura

En esta sección vamos a describir la arquitectura necesaria para resolver todos los requisitos que acabamos de plantear. Esta arquitectura proporciona una descripción del escenario.

3.4.1 Visión general

La Figura 8 nos muestra la arquitectura lógica para los requisitos previamente mencionados. Los componentes principales son recursos asegurados de la compañía, a los que pueden acceder los usuarios, los dispositivos de los usuarios, los dispositivos de la

compañía, los sensores, la red de la empresa y los diferentes servicios y aplicaciones, incluyendo la Aplicación de Control de Seguridad para controlar el firewall.

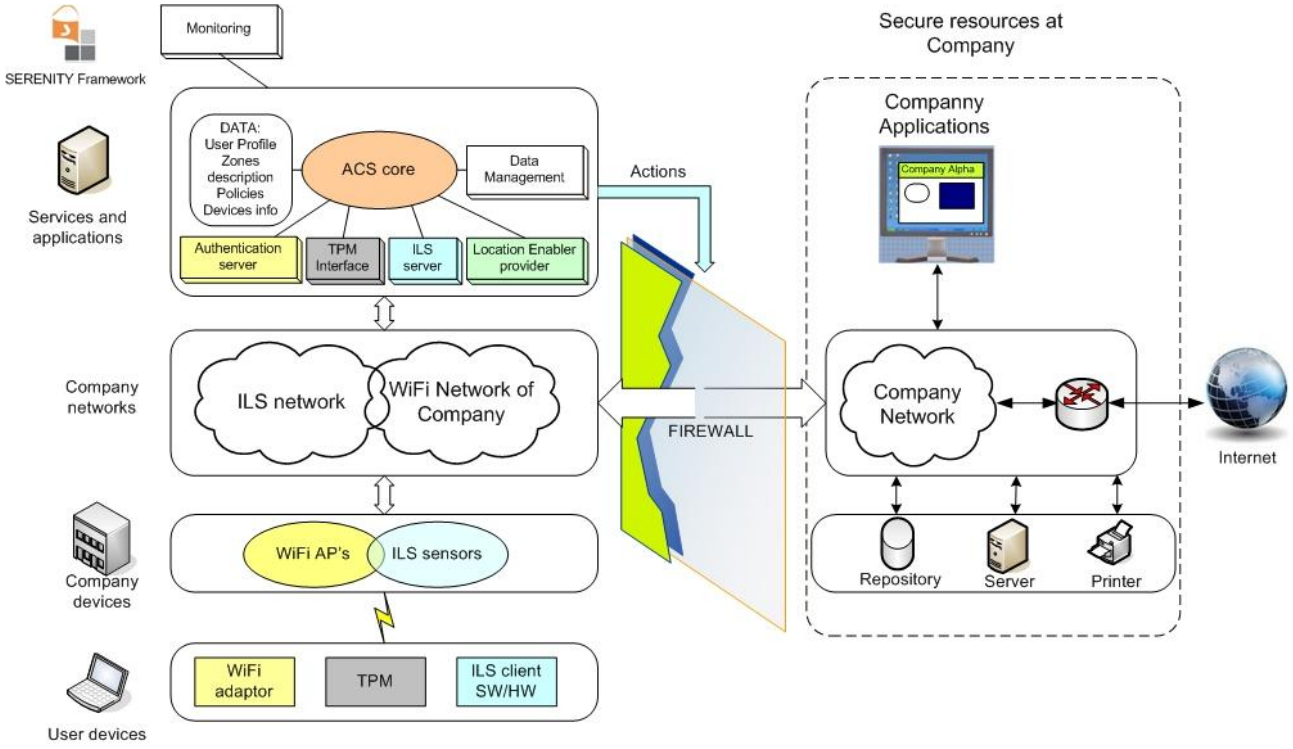


Figura 8 Visión general del escenario

3.4.2 Descripción de los elementos de la arquitectura

3.4.2.1 La aplicación de control de seguridad

Es la encargada de controlar el proceso de autenticación en la red. Toma las decisiones de permitir o no el acceso a cualquier recurso. Sus decisiones están basadas en las políticas definidas por el administrador, que fijan niveles mínimos de seguridad para cada recurso en base a tres parámetros: identificación del usuario, la zona en la que se encuentre y la identificación de la unidad a través de la cual está accediendo.

3.4.2.2 Indoor Location System, ILS. Sistema de Localización Interior

Proporciona información de la localización de todos los dispositivos que están conectados a la red de la empresa. Conceptualmente el ILS es un módulo diferente de la red Wi-Fi y su objetivo es proporcionar información acertada sobre la localización de las unidades basada en la información recogida por los sensores ILS. Puede estar basada en diferentes tecnologías. Como ya hemos mencionado, nosotros empleamos Ekahau, utilizado en redes Wi-Fi.

3.4.2.3 Firewall

Es el elemento activo para controlar los accesos a los recursos. Están definidas reglas que determinan qué recursos están accesibles y para qué dispositivos. La aplicación modifica las reglas del Firewall dinámicamente basándose en los elementos ya mencionados anteriormente (localización, identificación de usuario e identificación de dispositivo).

3.4.2.4 Red Wi-Fi

Es la red inalámbrica corporativa, tiene cobertura en todo el recinto de la compañía.

3.4.2.5 Red ILS

Es una red que comprende diversos sensores ILS que transmiten información al servidor ILS. Conceptualmente es una red distinta de la usada para las comunicaciones del usuario, pero en este caso empleamos la misma red Wi-Fi de la compañía debido a que, como ya hemos mencionado, nuestro sistema ILS está basado en señales Wi-Fi. Por tanto la Red ILS está embebida dentro de la red Wi-Fi de la compañía.

3.4.2.6 Sensores ILS

Son los dispositivos que determinan la información de localización de las unidades presentes en su área de cobertura. Están distribuidos por toda la compañía. Para nuestro caso, los sensores son los propios puntos de acceso de la red de la compañía.

3.4.2.7 Puntos de acceso

Proporcionan acceso a la red de la compañía. También tienen la función de sensores ILS, como ya hemos mencionado en el anterior punto.

3.4.2.8 Recursos de la compañía

Cualquier recurso accesible en la red de la compañía: aplicaciones corporativas, impresoras, acceso a internet, acceso a determinadas bases de datos…

3.4.2.9 Dispositivos de usuario

Cualquier dispositivo usado para conectarse a la red de la compañía. En nuestro caso es necesario que estos dispositivos instalen un software cliente para que se les pueda localizar e identificar vía TPM. Por tanto es recomendable, aunque no obligatorio, que dispongan de un chip TPM para que puedan ser identificados.

3.4.3 Funcionamiento de la arquitectura

El funcionamiento general del escenario puede dividirse en tres grandes apartados:

1. Conectarse a la red.
2. Activar la identificación de dispositivo y la localización.
3. Acceso a un recurso.

3.4.3.1 Conectarse a la red

Lo primero que tiene que hacer un usuario que quiera conectarse a la red Wi-Fi de la compañía (ponemos una compañía con una red Wi-Fi como caso general, a partir de ahora hablaremos de empresa o compañía) es activar su adaptador Wi-Fi y seleccionar el SSID de la compañía (para los visitantes, la empresa debería proporcionarles esta información). La red no permite conectarse a usuarios que no estén autenticados, en este caso los empleados tienen credenciales corporativas (usuario y password) y los visitantes usan credenciales comunes proporcionadas por la empresa (el mismo usuario y password para todos los visitantes).

Adicionalmente el canal inalámbrico está protegido por WPA [26], que es el método de acceso para redes inalámbricas propuesto por la Wi-Fi Alliance [27] y adoptado por el IEEE, estándar 802.11i, para solucionar problemas de seguridad observados en los métodos de encriptación basados en WEP. WPA y WPA2 están basados en el estándar 802.1X y el protocolo EAP para proporcionar una autenticación más segura. Hay dos métodos de uso considerados para WPA: personal y empresarial. En esta implementación el método usado es el empresarial, ya que está adaptado a una compañía con múltiples dispositivos de usuario.

Cuando el usuario activa su adaptador Wi-Fi se intenta establecer una conexión con la red Wi-Fi de la compañía. Para ello se envía un mensaje de autenticación por EAP sobre RADIUS [28]:

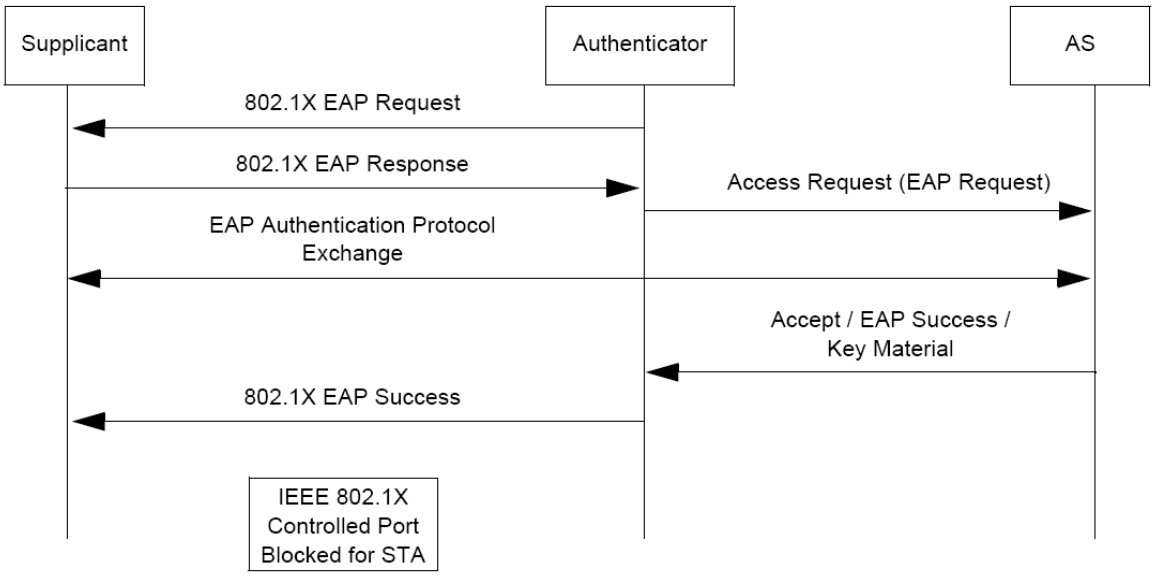


Figura 9 Recorrido del mensaje para autenticación basado en EAP-RAIDUS

Después de una autenticación correcta de RADIUS el usuario está conectado a la red corporativa. La parte que concierne a RADIUS es donde está ubicada la solución de autenticación de usuarios, digamos que esta solución integra la identificación en base a dicho protocolo y la consulta a la base de datos para confirmar esta autenticación y el nivel de seguridad.

3.4.3.2 Activar la identificación de dispositivo y la localización.

Una vez que el usuario consigue la conexión Wi-Fi, el siguiente paso es localizarlo e identificar los dispositivos a través del cual está conectado. Para los dispositivos se requiere que exista un software previamente instalado, este paso podemos obviarlo si presuponemos que los dispositivos de la empresa están listos para usarse en la red corporativa (hay disponible una guía de instalación en el anexo I para más detalles).

Para permitir la identificación del dispositivo y la localización es necesario que la unidad del usuario disponga de un hardware y un software que la hagan localizable e identificable. Detallamos estos aspectos a nivel lógico y funcional obviando los detalles de implementación:

- Localización: necesita software instalado en el dispositivo, pero no necesita ningún tipo de hardware adicional.
- Identificación: está basada en TPM, por tanto el dispositivo debería tener este chip. En el caso de que no lo tenga, la Aplicación de Control de Seguridad identificará el dispositivo en base a la MAC y le asignará, por tanto, un nivel bajo de seguridad en cuanto a identificación del dispositivo.

Un visitante que desee permiso para usar la red inalámbrica deberá instalarse un software que le proporciona la compañía. Este software comprende el cliente del ILS para que pueda ser localizado. La empresa también le proporciona un nombre de usuario y un password, que es común para todos los visitantes, cuyo perfil asignado es el de visitante. En general los visitantes no tendrán acceso nada más que a Internet, ya que no tienen ningún dispositivo asignado y su perfil de usuario tiene el nivel de seguridad más bajo. En cuanto a nivel de seguridad por localización, será el que esté asignado a la zona donde se encuentren.

3.4.3.3 Acceder a un recurso

Una vez que el usuario está autenticado en la red y tiene conexión Wi-Fi, puede intentar acceder a cualquier recurso disponible en la red corporativa. El sistema posee la información de localización de los usuarios conectados en tiempo real, por tanto la Aplicación de Control de Seguridad cambia los permisos de los usuarios si éstos cambian de localización. Los pasos de esta parte pueden resumirse en los siguientes:

1. Un usuario intenta acceder a un recurso a través de la red Wi-Fi.
2. Un paso de autenticación ha tenido lugar previamente para que el usuario tenga acceso a la red Wi-Fi. Entonces la información del usuario está disponible en la ACS.
3. El sistema de localización proporciona información acerca de la localización del usuario. Si no hay información de la localización, el usuario será marcado como “no localizable” y no se podrá garantizar su acceso a los recursos.
4. La Aplicación de Control es informada por el sistema de localización cuando un usuario cambia de zona. En estos casos la Aplicación cambia los permisos del usuario a tiempo real.
5. Para actualizar los permisos de los usuarios la Aplicación demanda la identificación del dispositivo. Esto genera una petición al TPM y éste a su vez proporciona sus datos de identificación a la Aplicación.
6. Teniendo en cuenta la información de localización, la identidad del usuario y la identificación del dispositivo, la Aplicación de Control permite el acceso, o no, a los recursos cuyos requisitos de seguridad sean menores o iguales que los del usuario. Esto provoca que la ACS se comunique con el firewall para actualizar los permisos del usuario.
7. Si un usuario intenta acceder a un recurso al que no puede acceder, el firewall no le permite el acceso. La configuración inicial del firewall es totalmente cerrada por defecto, sólo se proporciona acceso a recursos a los usuarios que cumplen con los requisitos de seguridad.

3.5 Planteamiento de la arquitectura

En esta sección se presenta la implementación del sistema. Aquí se plantean los principios para la implementación del escenario y detalles del desarrollo.El escenario del proyecto lo podemos ver en la Figura 10.

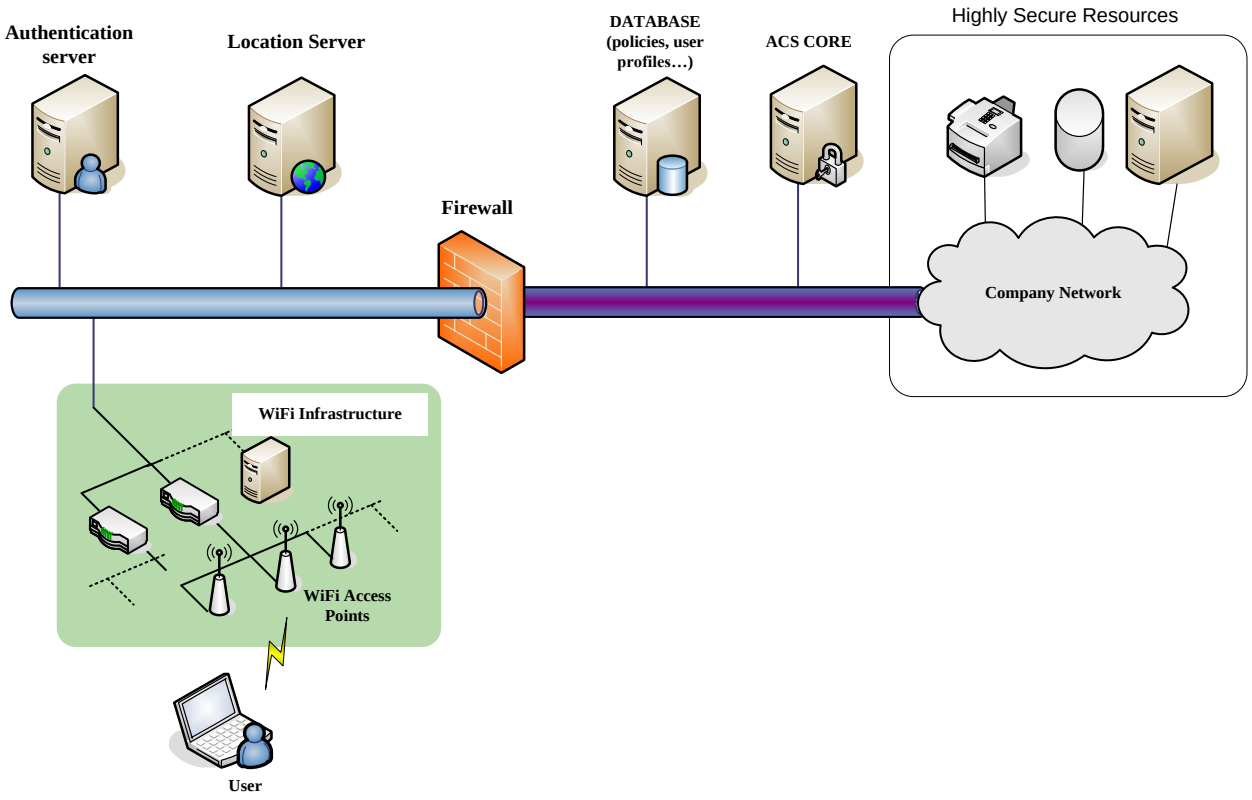


Figura 10 Vista general del escenario

Se compone de:

- Authentication Server: funciona con una implementación open source RADIUS para la autenticación.
- Location Server: dispone del software de Ekahau para localización, como ya hemos mencionado anteriormente.
- Database: Una base de datos SQL en la que se la información referente a los recursos y a los requisitos necesarios para acceder a ellos..
- ACS: es el "cerebro" del escenario, es el que posee toda la información y en base a ella otorga permisos para los recursos, o no.
- Los recursos pueden ser cualquier elemento conectado a la red, desde carpetas compartidas hasta impresoras o bases de datos.

- Firewall: implementación open source que está corriendo en una máquina que dispone de comunicación con el ACS.

Una red Ethernet cableada conecta todos estos componentes entre sí. La red inalámbrica está conectada a la red principal a través del Firewall. El firewall divide la red en dos partes: la parte protegida por él mismo y la parte "expuesta".

La tecnología de localización está basada en triangulación de señales Wi-Fi y la calidad de la calibración en un paso anterior. De este modo el ILS usa las señales de los puntos de acceso para determinar la localización de un dispositivo. Los puntos de acceso y los sensores ILS son los mismos en nuestro escenario.

Si miramos cómo debe ser usado el dispositivo del usuario, vemos que cualquier dispositivo con Wi-Fi y un chip TPM es válido. Ahora mismo en el mercado hay muchos ordenadores portátiles que disponen de ambas características. También es predecible que las PDAs y los teléfonos móviles tendrán chip TPM en un futuro. Si no dispusiera de un chip TPM también se puede usar en la red de este escenario, sacrificando el indicador de seguridad relacionado con la identificación del dispositivo.

3.6 Casos Supuestos

A continuación se muestra una serie de supuestos que resumen los posibles casos que se pueden dar en el presente sistema:

- Caso 1: Un empleado accede a internet desde un área restringida.
- Caso 2: Un empleado accede a un recurso muy protegido.
- Caso 3: Un empleado accede a recursos usando un dispositivo no asignado.
- Caso 4: Un empleado cambia su localización.
- Caso 5: Un visitante se conecta a internet desde un lugar público.
- Caso 6: Un usuario no autorizado intenta acceder a Internet usando la red Wi-Fi de la compañía A.

3.6.1 Personajes

Ahora que ya conocemos los factores que tendrá este escenario nos disponemos a presentar a todos los personajes que se dan cita en los casos supuestos.

- Pedro: es un project manager que trabaja para la compañía A, una compañía de publicaciones. Trabaja en la sede de la empresa, situada en el centro de la ciudad en un edificio con muchas oficinas en distintas plantas. Esta empresa instaló hace un tiempo una red Wi-Fi para ofrecer conexión inalámbrica a la red interna de la compañía y también para proporcionar servicios de internet para toda la empresa.

- Juan: es un cliente de Pedro. A Juan le gusta llevar su PDA para poder administrar su trabajo diario, y mira si hay conectividad especialmente si se encuentra fuera de su oficina, para poder llevar correctamente la gestión de sus proyectos.
- Rosa: trabaja en una compañía que se encuentra en el mismo edificio que la compañía A. La empresa de Rosa no tiene acceso a internet pero Rosa no tiene mucho trabajo y quiere chatear con algunos amigos. Rosa tiene un teléfono móvil con conectividad Wi-Fi y algunos conocimientos sobre la configuración de redes inalámbricas.

3.6.2 Procesos soportados

3.6.2.1 Caso 1: Un empleado accede a Internet desde una localización restringida

Pedro está en su oficina, en la tercera planta. Utiliza el ordenador portátil, en vez de un ordenador de sobremesa, desde que necesita viajar mucho, por tanto normalmente está conectado a la red Wi-Fi de la empresa. Él quiere examinar unas páginas Web en Internet. En primer lugar, cuando su dispositivo móvil se conecta al Wi-Fi tiene que introducir su nombre de usuario y contraseña. Después su dispositivo se comunica con nuestra Aplicación de Control de Seguridad, la cual pide la localización al servidor ILS (Indoor Location System). El ILS determina la situación de Pedro, o más exactamente de su dispositivo. Además el ACS comprueba que Pedro tiene asignado el dispositivo que está utilizando. Una vez pasado este proceso el ACS permitirá a Pedro acceder a las páginas Web que desee, suponiendo que la política de la empresa permita ver contenidos Web a los trabajadores debidamente identificados, tanto ellos como sus dispositivos, y debidamente localizados.

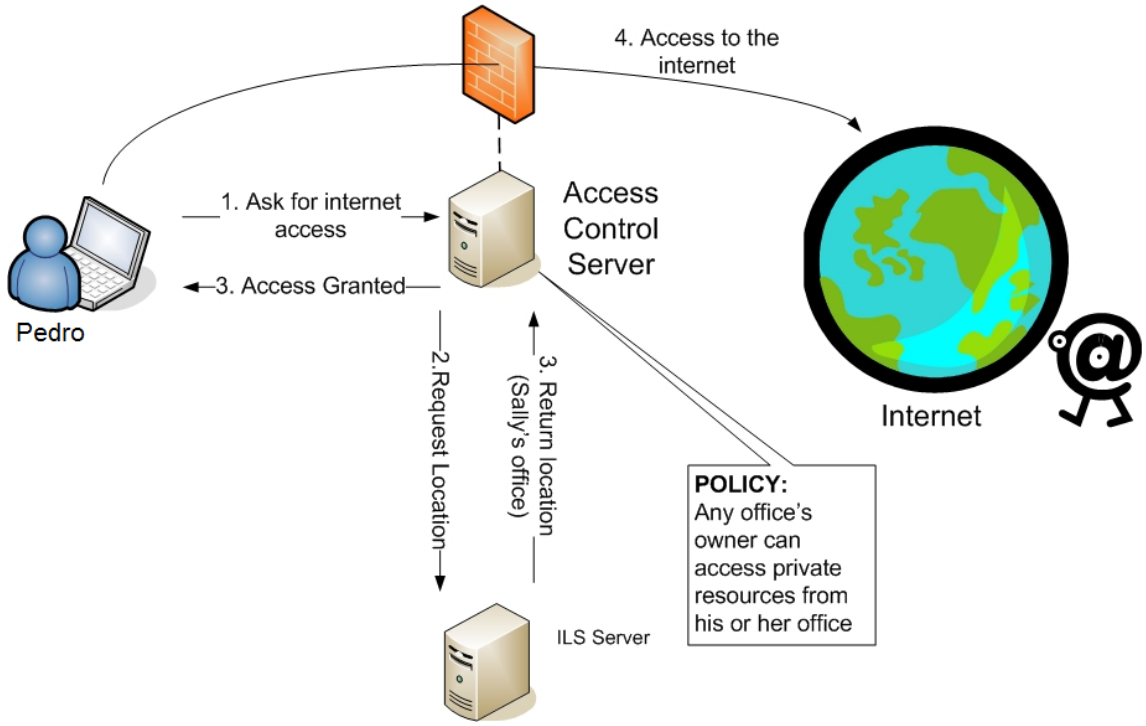


Figura 11 Acceso desde una localización restringida.

3.6.2.2 Caso 2: Un empleado accede a un recurso usando un dispositivo que no tiene asignado

El portátil que la empresa dio a Pedro no tiene batería y se le olvidó el cargador en casa. Un compañero le ha dejado su portátil para enviar un email. Cuando se conecta, él proporciona sus credenciales, pero en este caso este dispositivo no le está asignado. La política en este caso dice que solo unos recursos básicos pueden ser usados o accedidos: Internet y email. Pedro no podrá acceder a otros recursos privados o a impresoras, pero le sirve para enviar un email.

3.6.2.3 Caso 3: Un empleado cambia su localización

Debido a que ha venido un nuevo patrocinador para distribuir los libros que están publicando, Pedro ha organizado una reunión con los responsables de algunas empresas. La reunión tiene lugar en una de las salas de reuniones que la empresa A tiene en el edificio principal, en la segunda planta. Pedro coge su ordenador portátil por si necesita consultar alguna información durante la reunión. Ha olvidado el informe que imprimió, pero como dispone del portátil podrá acceder a la información aunque la sala de reuniones no sea tan segura como su despacho. Los responsables de las otras compañías pueden conectarse a la red Wi-Fi, pero como no son empleados de la compañía A, no pueden identificarse como usuarios con acceso a recursos privados, pero pueden acceder con el rol de invitados. Por tanto, gracias al enfoque SERENITY la ACS selecciona los métodos que

permiten a los invitados tener acceso a internet y a su vez les veta el acceso a los recursos privados de la compañía.

3.6.2.4 Caso 4: Un visitante se conecta a Internet desde una localización pública

Mientras tanto Juan, un cliente que iba a reunirse con Pedro cuando acabara la reunión con los responsables de las empresas, llega a la sede de la compañía A. Como Pedro está ocupado en la reunión con los responsables, Juan es llevado a una sala de espera cerca del hall. Mientras el está esperando, recuerda que tiene que leer un importante email de un compañero de trabajo, por tanto usa su PDA para conectarse a la red Wi-Fi de la compañía y así poder leer este email. En esta ocasión también gracias al enfoque SERENITY nuestra ACS evalúa el contexto de la situación y deduce que Juan se encuentra en un área que puede ser accedida por el público en general y por tanto Juan no podrá consultar los recursos privados.

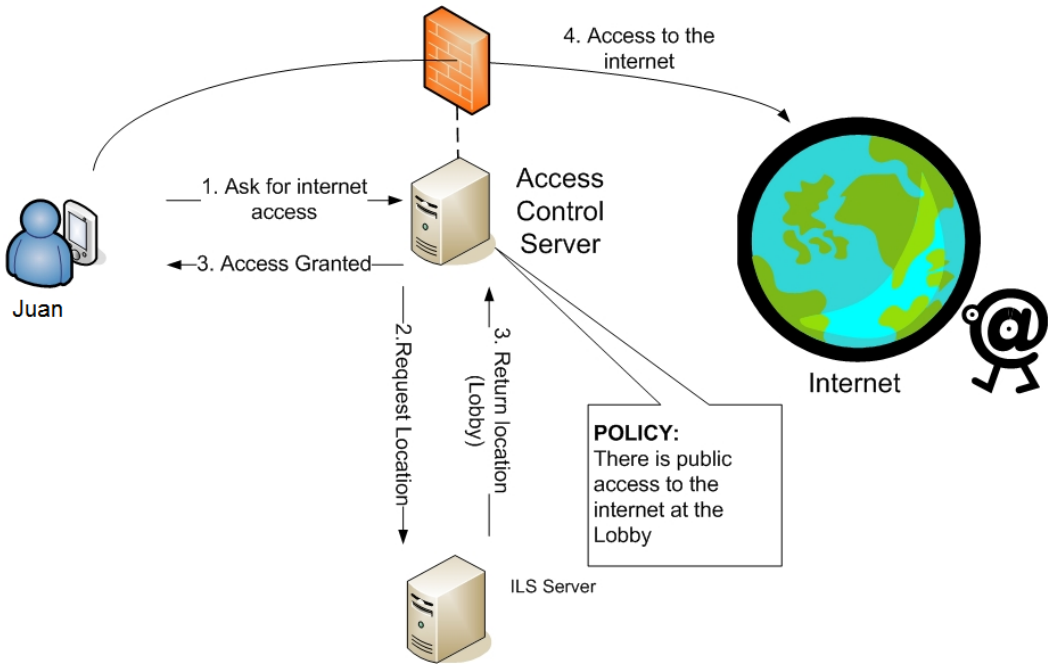


Figura 12 Un visitante accede desde una localización pública

3.6.2.5 Caso 5: Un usuario no deseado intenta conectarse a Internet usando la red Wi-Fi de la compañía A

Después de la reunión Juan y Pedro se van a tomar un café para tener un pequeño descanso. Mientras tanto Rosa, que trabaja para otra compañía en la cuarta planta del mismo edificio, intenta acceder a la red Wi-Fi de la compañía A. Quiere chatear con algunos amigos pero en su empresa no hay acceso a internet. En este caso nuestra ACS

detecta que el usuario está intentando acceder desde una zona en la que no está la compañía A y por tanto este usuario no podrá acceder a la red por ningún motivo.

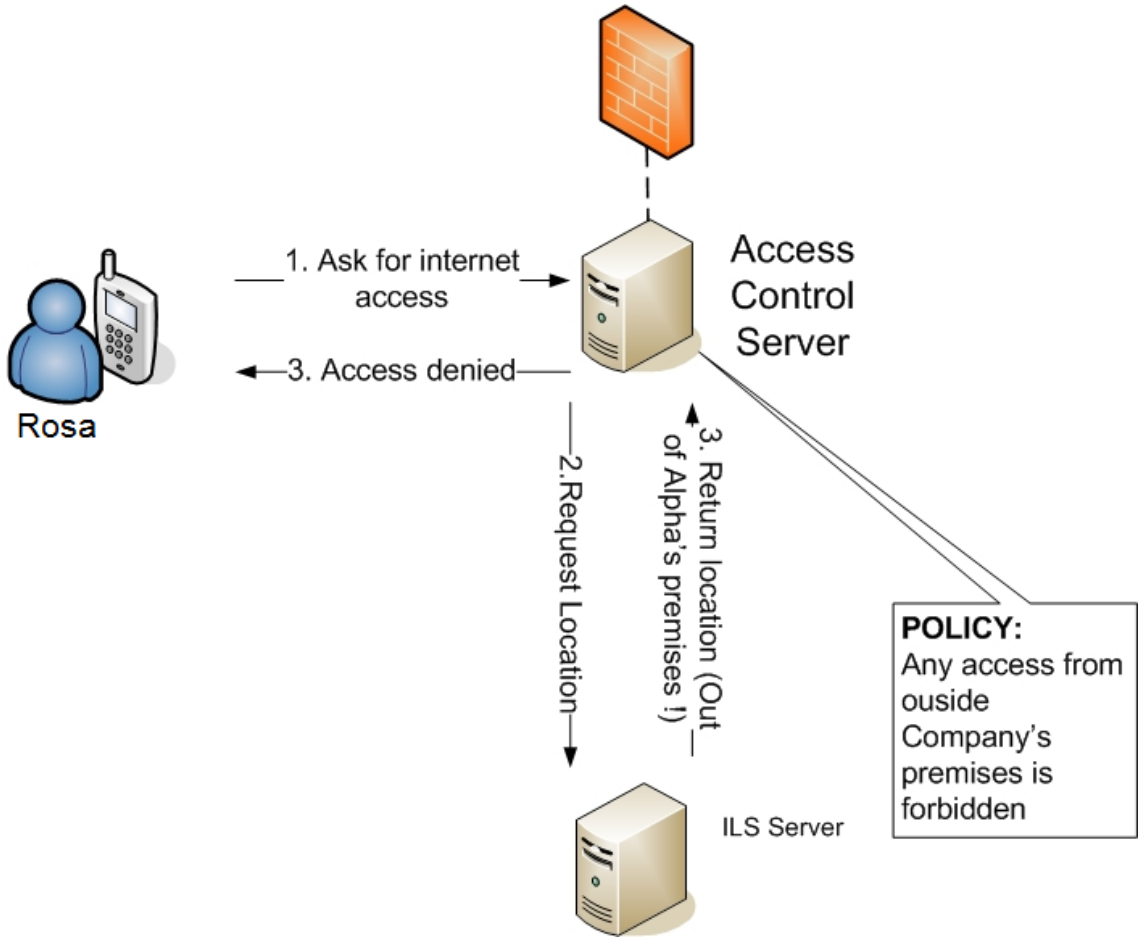


Figura 13 Acceso desde una localización restringida

3.7 Trabajo realizado

En el escenario consideramos situaciones relacionadas con la seguridad en la red. En todas ellas tenemos uno o más usuarios intentando acceder a recursos a través de la red inalámbrica desde una localización específica. Los usuarios son primero autenticados asociándose su identidad a un perfil de usuario. Un Servidor de Control de Acceso ("ACS" en sus siglas en inglés) decide si acepta o deniega el acceso teniendo en cuenta los siguientes aspectos:

- Localización: usando la información proporcionada por un ILS (Indoor Location System).
- Identidad del usuario: los usuarios son autenticados para iniciar sesión y pueden ser adecuadamente clasificados en ciertos perfiles, como administrados, empleado, visitante…

- Identificación de dispositivo: cada unidad es identificada usando TPM (Trusted Platform Module).

Teniendo en cuenta los requisitos, hemos diseñado el siguiente prototipo donde ha sido adoptado el modelo SERENITY, ver siguiente figura:

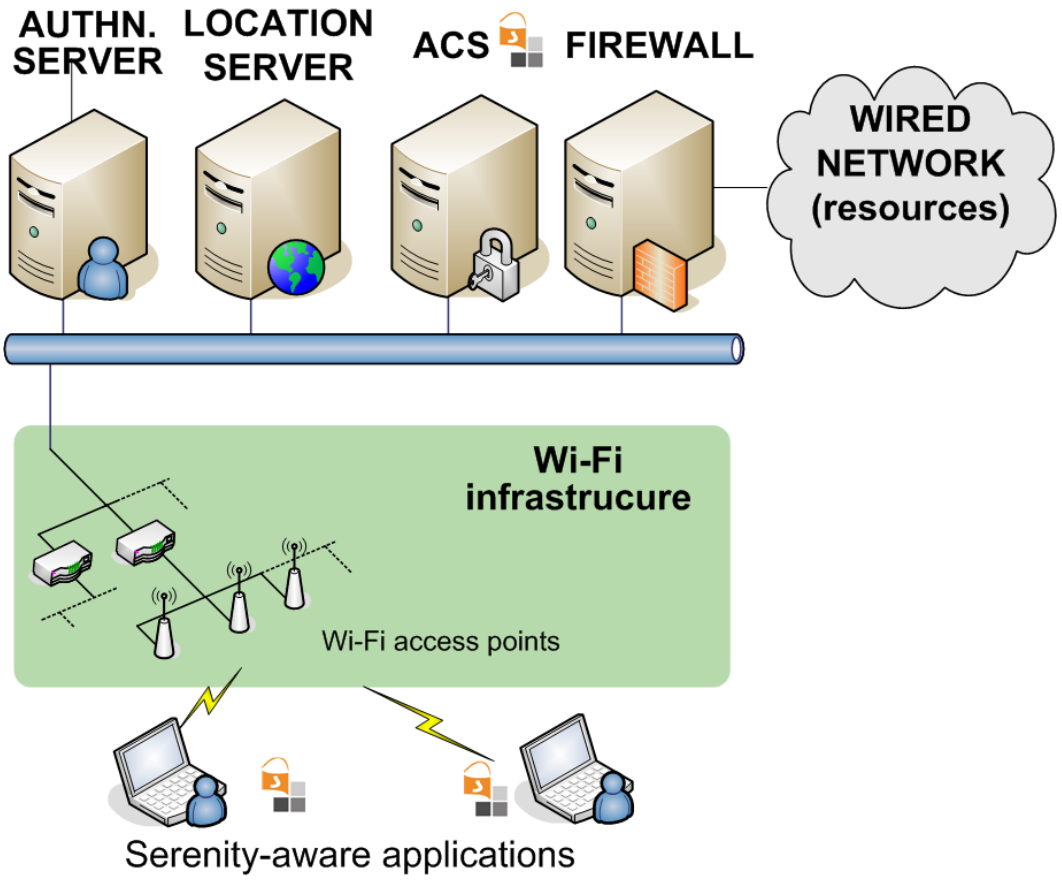


Figura 14 Elementos del proyecto

En esta figura pueden reconocerse los principales factores presentes en el escenario del proyecto.

En resumen, los artefactos S&D utilizados en el escenario son:

- Patrón de identificación de dispositivos basado en TPM: este patrón representa un mecanismo para identificar una unidad basado en la tecnología TPM. Una unidad con dispositivo TPM activo puede identificarse mediante métodos criptográficos usando el TPM con ese propósito. El TPM provee varias funciones criptográficas basadas en hardware que permiten identificar al dispositivo de forma segura.
- Patrón de localización: la solución presentada por este patrón proporciona niveles de seguridad específicos para cada una de las zonas dentro de las áreas

controladas. El patrón selecciona un perfil de seguridad dependiendo de la zona en la que se encuentre el usuario.

- Patrón de autenticación del usuario: este patrón se encarga de autenticar al usuario y asignarle un perfil de seguridad.

Para dar una utilidad a todos estos artefactos, y como parte fundamental del proyecto, hemos desarrollado una aplicación de control basada en SERENITY que, mediante la utilización de todo lo dicho anteriormente, implementa las funcionalidades que ponemos como objetivos en este escenario. A continuación hacemos un listado de los elementos que componen el escenario, especificando aquellos que han debido de implementarse para desarrollar el proyecto:

- La infraestructura Wi-Fi es la principal red del escenario. Adicionalmente esta red es la base para la localización.
- Para el servidor de autenticación se ha desarrollado una solución S&D que procesa las peticiones de conexión, y que asigna una valor de seguridad una vez que el usuario está autenticado correctamente. La parte fundamental de esta solución es el Componente Ejecutable que está desarrollado en Java con la ayuda de una API para conexiones SQL. Usamos el protocolo EAP-RADIUS para la autenticación.
- Para el servidor de localización se ha desarrollado una Solución S&D. La parte fundamental de esta solución es el desarrollo del EC que se está programado en Java. Usará el apoyo del software de Ekahau.
- Para la identificación de dispositivos se ha desarrollado también una solución S&D cuya parte fundamental es el desarrollo de dos ECs. En este caso necesitamos un EC que esté en el servidor haciendo las peticiones de identificación y otro EC, situado en el cliente, que se encarga de identificar el dispositivo del cliente de forma segura y fiable gracias al chip TPM. En esta solución observamos que es fundamental la presencia de dos instancias del SERENITY Runtime Framework (SRF, se puede ver la explicación en la sección 4.2.2.2), una el servidor y otra en el cliente.
- El Servidor de Control de Acceso (ACS) que implementa el núcleo de las funcionalidades del sistema: controla la autenticación de usuarios, identificación de dispositivos y la localización. El elemento fundamental del ACS es la Aplicación de Control de Seguridad que hemos desarrollado en base al modelo SERENITY, por tanto precisa de una instancia del SRF. Todo el desarrollo de la Aplicación de Control es en Java, se comunica con los distintos elementos gracias a diversas APIs. Los cambios en la seguridad podrían resultar en una reconfiguración dinámica de las reglas del filtro del firewall por parte de

la Aplicación de Control. Todo esto podemos verlo con mucho más detalle en la sección de imple.

- El Firewall separa a la red inalámbrica del resto de la red (cableada) y es configurado dinámicamente por el ACS. Para la implementación del Firewall utilizamos una máquina virtual de Línux, con un debian instalado, lo configuramos mediante iptables. La comunicación entre la Aplicación de Control y el Firewall corre a cargo de la Aplicación de Control y está programado en Java gracias a una API.
- Servidor DHCP, para otorgar las direcciones ip dentro de la red local. También está alojado en una máquina virtual debian y está configurado mediante unos comandos que tiene debian para poder utilizar una máquina como servidor dhcp.
- Integración del Servicio de Monitorización. Es un módulo aparte que se encarga de monitorizar las soluciones implementadas para que la Aplicación de Control tenga constancia del funcionamiento de los Componentes Ejecutables a tiempo real.
- Integración de todos estos elementos en un escenario plenamente operativo con capacidad de gestión a tiempo real gracias a la aplicación de los conceptos SERNEITY y los conceptos presentados en los escenarios AmI.

En definitiva observamos que todos estos puntos son los objetivos a desarrollar en el pfc. Todos ellos se programarán en Java, como ya hemos mencionado. La parte de clases, patrones e implementaciones S&D será realizada en XML, como también hemos mencionado anteriormente. El SRF está proporcionado por SERENITY, y el software de localización por Ekahau, se hará uso de este software desde Java gracias al uso de diversas APIs.

Todos los aspectos a desarrollar e integrar se detallan en el siguiente apartado de implementación, donde describiremos cada una de las partes por separado y el trabajo realizado en cada uno de los elementos presentes en el proyecto.

4 Implementación

4.1 Descripción general

En esta sección vamos a tratar la siguiente información:

- El proceso de implementación, así como la unión de todos los elementos no implementados.
- La arquitectura.
- El entorno implementado.
- Información relativa al proceso de implementación.

Este proyecto se centra en la interacción de los usuarios y sus dispositivos con una red Wi-Fi corporativa, todo el conjunto reacciona a tiempo real ante variaciones en el entorno o en los usuarios.

En nuestro caso el escenario se centra en tres parámetros que afectan a las consideraciones sobre seguridad:

- Identidad: cada usuario tiene asignado en la base de datos un perfil de usuario que le otorga un cierto nivel de seguridad dependiendo de su posición en la empresa.
- Localización: como ya hemos ido describiendo a lo largo de esta memoria, la localización es un aspecto fundamental para la seguridad. Cada una de las zonas de la empresa tendrá un nivel de seguridad diferente.
- Identificación del dispositivo: también lo hemos ido mencionando a lo largo de esta memoria. Conocer el dispositivo a través del cual está accediendo cada usuario es un factor también muy importante. Cada dispositivo tendrá asignado en la base de datos un nivel de seguridad.

4.1.1 Descripción del escenario

4.1.1.1 Introducción

La arquitectura está basada en las redes Wi-Fi que puede haber en cualquier negocio medianamente grande, donde conviven la red de cable con la red inalámbrica. La parte inalámbrica de la red está conectada a través de un firewall, ya que los usuarios inalámbricos son en general más vulnerables en cuanto a seguridad. En la Figura 15 se puede observar cómo está planteado todo el escenario.

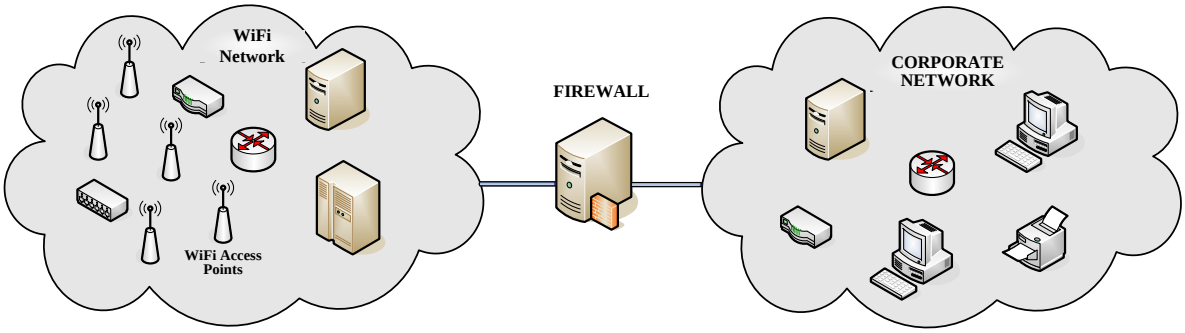


Figura 15 Entorno global

4.1.1.2 Componentes y arquitectura

La Figura 16 describe la arquitectura empleada.

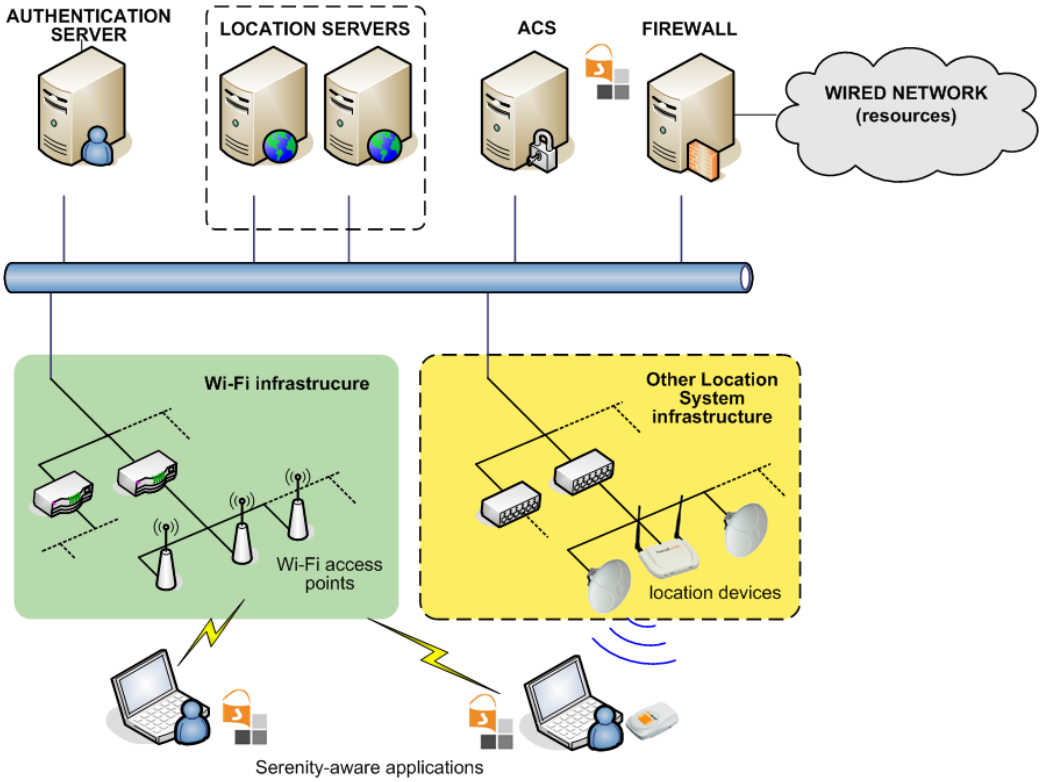


Figura 16 Arquitectura del escenario

En esta figura podemos reconocer todos los elementos principales presentes en el escenario (en una primera instancia podemos observar las entidades que utilizan SERENITY, se puede distinguir en ellas el icono SERENITY):

- La red Wi-Fi es la red protegida, que requiere los servicios de autenticación, localización e identificación de dispositivos. Está compuesta por puntos de acceso, hubs y por un servidor DHCP. Adicionalmente es la base para el sistema de

localización, que utiliza la triangulación de las señales Wi-Fi, como ya hemos mencionado anteriormente.

- Abajo a la derecha podemos observar cómo hay representado otro sistema de localización alternativo. Este sistema es una propuesta futura para proporcionar al menos dos soluciones SERENITY referidas a la localización del dispositivo. En este proyecto no hemos desarrollado esta otra solución, está representada para mostrar que el esquema modular de todo el escenario permite cambiar las soluciones sin modificar todo el conjunto. Si quisiéramos que la ACS utilizara otro sistema de localización habría que implementar otro patrón del modelo SERENITY. Y para su utilización bastaría con añadir este nuevo patrón a la base de datos de patrones del SRF.

- El Servidor de Autenticación usa el protocolo EAP-RADIUS. Procesa las peticiones de los usuarios y les asocia un perfil mediante una consulta a la base de datos. Si no figuran en la base de datos no proporciona ningún perfil y devuelve un código por el que la Aplicación de Control reconoce que no está identificado. Este servidor envía a la Aplicación de Control tanto el nombre del usuario como el valor de seguridad asignado a su perfil.

- El Servidor de Localización proporciona la localización de todos los usuarios conectados y envía esta información a la Aplicación de Control. Hemos utilizado una API proporcionada por el fabricante para desarrollar este elemento. El código desarrollado comunica con el servidor de Ekahau y consulta la zona en donde se encuentran los usuarios, posteriormente se asigna el valor de seguridad de la zona y se envía, tanto la zona como el valor de seguridad, a la Aplicación de Control.

- Los dispositivos de los usuarios pueden ser portátiles o cualquier unidad portátil, véase móviles, PDAs… En este proyecto los dispositivos deberían disponer de un chip TPM, aunque no es estrictamente necesario, para poder identificarlos. Como vemos en la Figura 16, los dispositivos de los usuarios poseen una aplicación cliente SERENITY y una instancia del SRF. El software desarrollado para el cliente procede a enviar, mediante un socket previamente fijado, una serie de paquetes donde se comunica en primer lugar la MAC del dispositivo y posteriormente se procede con el Remote Attestation (protocolo descrito en la sección 4.2.2.4.2.3). Por tanto, para que haya una identificación correcta hay dos instancias del SRF, una en el cliente y otra en el servidor. Además tienen que estar cargados los ECs de TPMServer y TPMClient por la Aplicación de Control y por la Aplicación cliente respectivamente.

- El ACS, Access Control Server, es donde se encuentra principalmente la Aplicación de Control de Seguridad, que a su vez implementa las funcionalidades del núcleo del sistema. Controla el firewall para permitir el acceso, o no, a los

recursos basándose en los perfiles de usuario, en su localización y en la identificación del dispositivo. Esta aplicación es totalmente una aplicación SERENITY donde está corriendo una instancia del SRF. El SRF es el responsable de la selección y aprovisionamiento de la solución, o las soluciones, que demande la aplicación. Por tanto el SRF proporcionará los tres ECs que demandará nuestra Aplicación de Control, y estos a su vez cargan los servidores de localización y de autenticación y el server para el Remote Attestation.

- El Firewall separa la red inalámbrica del resto de la red. Nuestro Firewall está desarrollado en una máquina aparte, Linux, y se comunica con la Aplicación de Control vía SSH. La máquina es Linux para poder implementar el Firewall con *iptables*. La Aplicación de Control modifica la reglas *iptables* del firewall dinámicamente dependiendo de las variaciones en alguno de los tres factores de seguridad.

4.1.1.3 Entorno de implementación

La mayoría de la implementación de este proyecto son desarrollos en Java, por tanto se puede considerar prácticamente independiente del sistema operativo. Sin embargo en el caso del software de localización se requiere que el sistema operativo sea MS Windows. La siguiente tabla proporciona información acerca de los detalles de implementación de los componentes del prototipo.

Componente	SO	Software
Servidor de Autenticación	Windows / Linux / Otros	FreeRADIUS
Servidor de Localización	Windows	Commercial location software
Bases de datos	Windows / Linux / Otros	PostgreSQL y MySQL
Aplicación de Control	Independiente del SO	Desarrollo en Java
Firewall	Linux	Comunicación SSH e *iptables*

Tabla 2 SO y software requerido

Podemos observar como el software es código abierto, menos el software de la localización.

4.1.2 Integración con el SRF

4.1.2.1 Ubicación y utilidades del SRF en la Aplicación de Control

Nuestro escenario del proyecto tiene dos instancias del SRF, como hemos apuntado en la descripción de la arquitectura:

- Aplicación de Control de Seguridad: está encargada de tomar decisiones de acceso, computar diversas variables que recibe de los Componentes Ejecutables cargados (localización, perfil de usuario e identificación de dispositivo) y una de las características principales es la adaptación dinámica a los cambios, como el cambio de zona. La Aplicación de Control es una aplicación SERENITY que solicita las soluciones S&D al SRF.
- Dispositivos de clientes: hay una instancia del SRF presente en la conexión con el dispositivo del cliente. Este SRF es capaz de establecer una negociación con el SRF presente en el servidor principal. La base de datos del SRF del cliente es lógicamente mucho menor que la que está presente en el SRF del servidor, por tanto el SRF del cliente es una instancia más ligera que el del ACS.

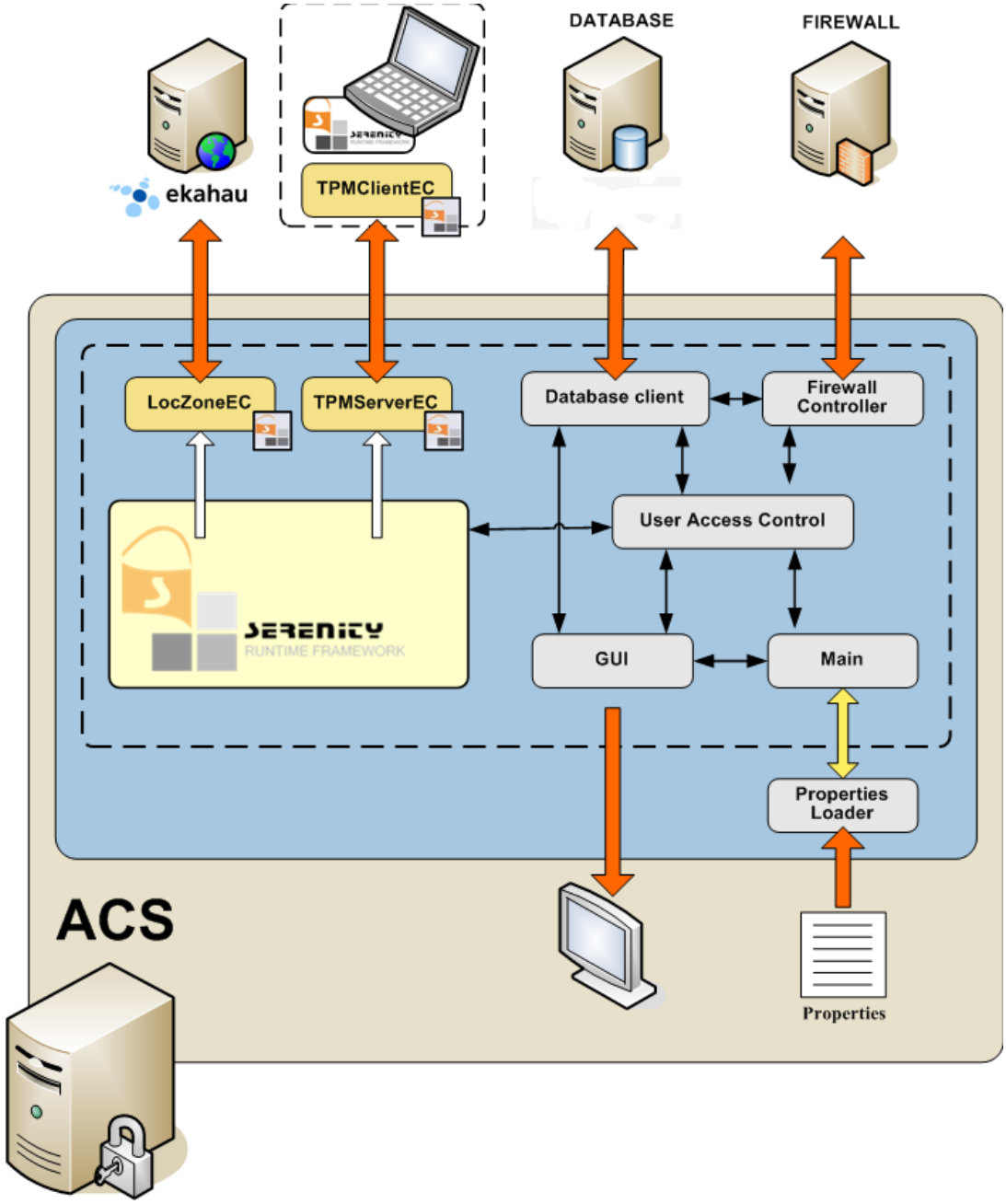


Figura 17 Arquitectura de la Aplicación de Control

En Figura 17 se describe la arquitectura de la Aplicación de Control de Seguridad. La relación con el SRF es la siguiente: la Aplicación le solicita soluciones y el SRF escoge los Componentes Ejecutables (ECs) que cumplen los requisitos citados por la Aplicación. Estos ECs interactúan con otros elementos como bases de datos o el Servidor de Localización. También podemos observar que el SRF está presente en el dispositivo del usuario, podemos ver también el Componente Ejecutable cliente (llamado *TPMClientEC*) y la relación con el Componente Ejecutable presente en la Aplicación (*TPMServerEC*). Los dos Componentes Ejecutables forman parte de la misma solución S&D y del mismo

patrón con dos roles, con una implementación distribuida. El tercer Componente Ejecutable no se muestra por razones de claridad en la ilustración, pero estaría situado dentro de la Aplicación de Control y con comunicación con la base de datos.

4.1.2.2 Funciones del SRF en el proyecto

Las funciones del *SERENITY Runtime Framework,* SRF, son las siguientes:

- Enlace: en cada petición el SRF busca la mejor Solución S&D para una serie de requisitos. El SRF escoge un Componente Ejecutable que se corresponda con los requisitos que demanda la aplicación. El SRF carga este Componente Ejecutable para que la aplicación pueda utilizarlo. El SRF comprueba todas las soluciones que están presentes en su base de datos. En nuestro caso tenemos tres Artefactos S&D cargados en la base de datos del SRF de la Aplicación de Control que se corresponden con los tres Componentes Ejecutables. Estos Artefactos S&D cargan el patrón de localización, el de identificación de dispositivo y el de autenticación, que a su vez se refieren a sus respectivos Componentes Ejecutables (*LocZoneEC*, *TPMServerEC*, *IdentityEC*). En el caso del SRF del usuario sólo habrá un Artefacto S&D que se corresponde con la identificación de dispositivo, que cargará el Componente Ejecutable *TPMClientEC*.

- Negociación: esta característica es necesaria cuando hay patrones distribuidos. En nuestro caso el patrón distribuido es el de identificación de dispositivo, como ya hemos mencionado, que está presente en el usuario y en el servidor. Por tanto es necesario que se sincronicen para llevar a cabo la identificación.

- Servicio de Monitorización y mecanismos de reacción: cuando el contexto cambia podría ser posible que alguna de las soluciones seleccionadas ya no sea apropiada. Los Patrones S&D cargados cuentan con sus propias reglas de monitorización para detectar cambios, y los Componentes Ejecutables tienen implementados un Capturador de Eventos que generan los eventos que son mandados al Monitor. El uso que le damos en nuestro proyecto es básico, debido a que el servicio de Monitorización no tiene capacidad suficiente para operar con eventos complejos. En nuestro caso el Monitor es empleado en el Componente Ejecutable de Localización para comprobar si el usuario detectado no supera un cierto criterio de calidad de la señal recibida. En ese caso el Servicio de Monitorización envía un mensaje a la Aplicación de Control a través del SRF para que el administrador sepa cómo está funcionando el sistema a tiempo real.

4.1.2.3 Integración del SRF en cada situación

- Integración del SRF dentro del Access Control Server, ACS: el núcleo del escenario es la Aplicación de Control que implementa toda la lógica por la cual se

permite el acceso, o no, a los recursos protegidos. Para conectar la Aplicación de Control con el SRF, y por consiguiente con los tres Componentes Ejecutables encargados de la seguridad, usamos llamadas al SRF (creando conexiones SSL). Estas llamadas, y la comunicación con el SRF en general, tanto por parte de la Aplicación de Control como por parte de los Componentes Ejecutables se hacen a través de APIs, una para la Aplicación y otra para los Componentes Ejecutables, *SERENITYApplicationDevSupportLib* y *SerenityECDevSupportLib* respectivamente.

- Integración del SRF en los dispositivos de usuario: la aplicación cliente también necesita una manera de comunicarse con el SRF, para poder comunicarse con el *TPMClientEC*. En definitiva la manera es la misma que para la Aplicación de Control, a través de unas APIs que hemos mencionado al final del punto anterior.

4.1.3 Descripción de los Patrones S&D

A continuación se van a describir los tres patrones S&D utilizados. Estos son el patrón de identificación, el patrón de localización y el patrón de autenticación.

4.1.3.1 Patrón de identificación de dispositivos vía TPM

Este patrón ofrece la posibilidad de proporcionar un nivel de seguridad de un dispositivo previamente identificado. Este nivel de seguridad ha sido fijado previamente cuando hemos aprovisionado la base de datos. El método de identificación se realiza mediante el protocolo Remote Attestation. El chip TPM que se encuentra en el dispositivo cliente es el que identifica unívocamente a dicho dispositivo de una manera criptográfica muy fiable.

Remote Attestation es un protocolo de autenticación open source. En nuestro caso no hemos encontrado ninguna implementación funcional y por tanto hemos tenido que implementarlo. La implementación es en Java y tiene más funcionalidades que las empleadas para este proyecto.

La atestación se refiere al protocolo de comunicación existente entre un servidor y un cliente cuándo el primero quiere verificar si el cliente que se encuentra en un estado seguro. El TPM está fabricado con una clave pública/privada que se encuentra a nivel de hardware, llamada Endorsement Key (EK). La parte pública de la clave está certificada por una autoridad de certificación, (CA, Certification Authority). Cada TPM tiene una EK única. Usando la parte privada de esta EK, el TPM puede firmar confirmaciones de la fiabilidad del sistema. Un ordenador remoto puede verificar que estas confirmaciones han sido firmadas por un TPM fiable.

Ese es el fundamento del Remote Attestation, pero observamos que si cada TPM tiene una única Endorsement Key nos sirve para identificar unívocamente a cada

dispositivo que tenga un chip de estas características. En este proyecto no verificamos la integridad del dispositivo conectado, aunque la implementación hecha permitiría esta característica en el caso de que algún administrador la solicitara.

Más adelante describiremos el funcionamiento detallado del Remote Attestation, cuando detallemos cada uno de los elementos desarrollados en el proyecto (sección 4.2.2.4.2.3).

4.1.3.1.1 Requisitos cubiertos

De los requisitos necesarios para el escenario presentados en la sección 3.3, vemos como se cubre el siguiente con el desarrollo del patrón de identificación de dispositivos vía TPM:

- El equipamiento de los empleados debe tener un hardware y/o un software que lo identifiquen unívocamente. La información provista debe ser tangible y segura. Por tanto la identificación del dispositivo debe garantizar que el dispositivo es realmente el que dice ser. El equipamiento de los visitantes podría tener o no el hardware/software apropiado.

4.1.3.2 Patrón de localización

Ahora pasamos a describir el patrón de localización que ya hemos ido introduciendo anteriormente. El objetivo de este patrón es proporcionar la localización de dispositivos en determinadas zonas controladas. En una red Wi-Fi de una oficina podemos encontrar distintas zonas con distintos uso públicos y privados. Por tanto hay que diferenciar los distintos niveles de seguridad en cada uno de esos espacios. En definitiva el patrón debe proporcionar un valor de seguridad dependiendo de la zona en la que se encuentre el dispositivo localizado y la zona en la que se encuentra dicho dispositivo, así como la localización exacta en coordenadas en el caso de que la Aplicación de Control desee mostrar un mapa de la ubicación.

El Componente Ejecutable de localización pregunta al servidor Ekahau la zona en la que se encuentra el dispositivo en cuestión, lo identifica en base a la MAC, y el servidor Ekahau responde con la localización. En el caso en el que la aplicación pida la localización exacta el Componente Ejecutable pregunta al servidor Ekahau las coordenadas de localización. Hay cuatro elementos que forman parte de la implementación:

- Aplicación de Control.
- Sistema de Localización, Ekahau RTLS es el que hemos empleado.
- Puntos de acceso.
- Dispositivo del usuario.

La Figura 18 nos muestra la interacción entre los distintos actores.

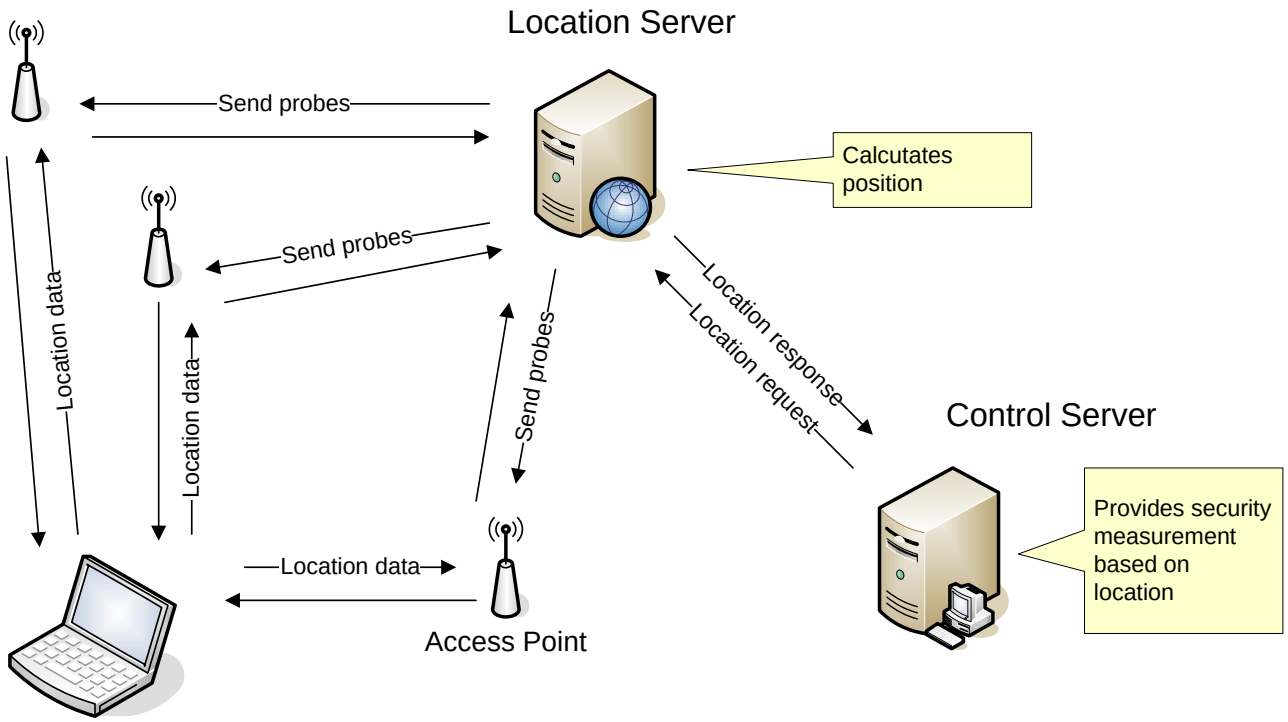


Figura 18 Solución de localización

Una vez que el Componente Ejecutable tiene los datos de la localización, éste sólo tiene que consultar en la base de datos para asignarle un valor de seguridad.

4.1.3.2.1 Requisitos cubiertos

De la lista de requisitos presentados en la sección 3.3, este patrón cubre los siguientes:

- El sistema ILS debe estar disponible como poco durante las horas en que la red Wi-Fi está operativa en la compañía. Además debe estar garantizado el funcionamiento del servidor ILS.
- El servidor ILS debe proporcionar información con, al menos, un cierto nivel de acierto para el área restringida. Este nivel debe poder ser configurado en el sistema para poder garantizar un cierto nivel de confianza para la información de localización.
- El equipamiento de acceso debe tener el hardware y el software requerido para permitir que sea localizado e identificado. Este hardware/software debe estar protegido para evitar intentos de fraude.
- El ILS debe ser capaz de detectar que un usuario desautorizado no está dentro del área que abarca el escenario.

- Debe haber un nivel de seguridad asociado a las localizaciones de fuera del escenario. La información de estas zonas debe estar almacenada en el repositorio o en un servidor seguro.

4.1.3.3 Patrón de autenticación de usuarios

Como los patrones anteriores, éste proporciona nivel de seguridad basado en los perfiles de usuario. Cada uno de los usuarios en la base de datos tiene asignado un perfil de seguridad que a su vez tiene asignado un valor, un nivel de seguridad. Para los visitantes todos tienen el mismo perfil y tiene el nivel de seguridad mínimo.

Perfil de Usuario	Nivel de Seguridad
Visitante	0
Becario	1
Trabajador	2

Tabla 3 Perfiles de seguridad

En la Tabla 3 podemos ver un ejemplo de cómo establecer los niveles de seguridad. Estos datos figuran en la base de datos PostgreSQL del Componente Ejecutable. En la base de datos están los nombres de los respectivos usuarios y sus perfiles de seguridad, así como los perfiles de seguridad asociados a los niveles de seguridad (tal y como podemos ver en la Tabla 3).

4.1.3.3.1 Requisitos cubiertos

De la lista de requisitos presentados en la sección 3.3, este patrón cubre los siguientes:

- El visitante debe poseer las credenciales generales para conectarse como invitado de la compañía. La compañía debe garantizar que esta información es fiable.
- El terminal del empleado debe disponer de las capacidades para permitir identificación y autenticación del usuario dependiendo de su nivel de seguridad, como está definido por la Aplicación de Control de Seguridad y por SERENITY.

4.2 Descripción detallada de los elementos desarrollados en el proyecto

4.2.1 Introducción

Ya hemos visto los elementos por los que está compuesto el escenario. En la Figura 19 se puede observar los elementos globales que hemos implementado para la consecución del proyecto. En los siguientes apartados iremos analizando uno a uno los elementos y los métodos de programación empleados en su desarrollo. Del mismo modo se mostrará el empleo de APIs o de elementos preexistentes, como puedan ser el SRF o el servicio de Monitorización.

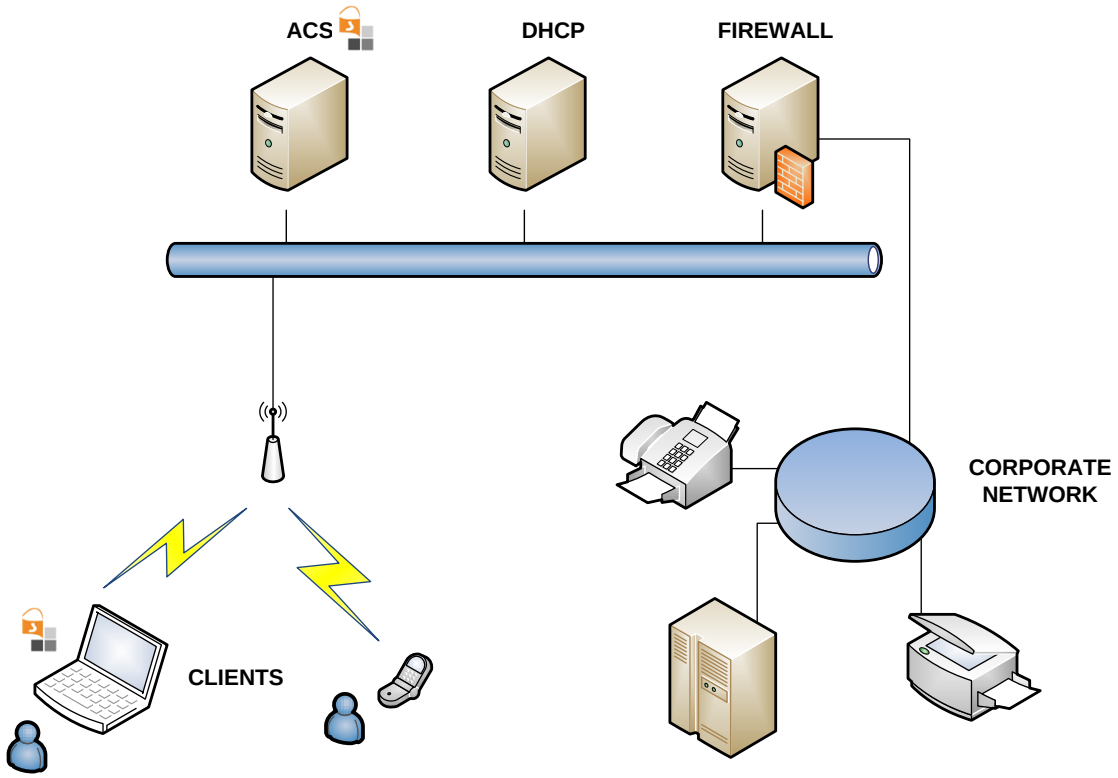


Figura 19 Visión global del proyecto

Una vez presentado el escenario desde el punto de vista del programador, los bloques funcionales observados en la Figura 19 son los que hemos tenido en cuenta en última instancia para la consecución del proyecto.

4.2.2 ACS

Empezamos con el Servidor de Control de Acceso. Es la parte más importante del proyecto y acompañaremos todas las descripciones de elementos con sus respectivas figuras que ilustren el funcionamiento de todos los elementos. En la Figura 20 podemos ver todos los elementos que componen el ACS:

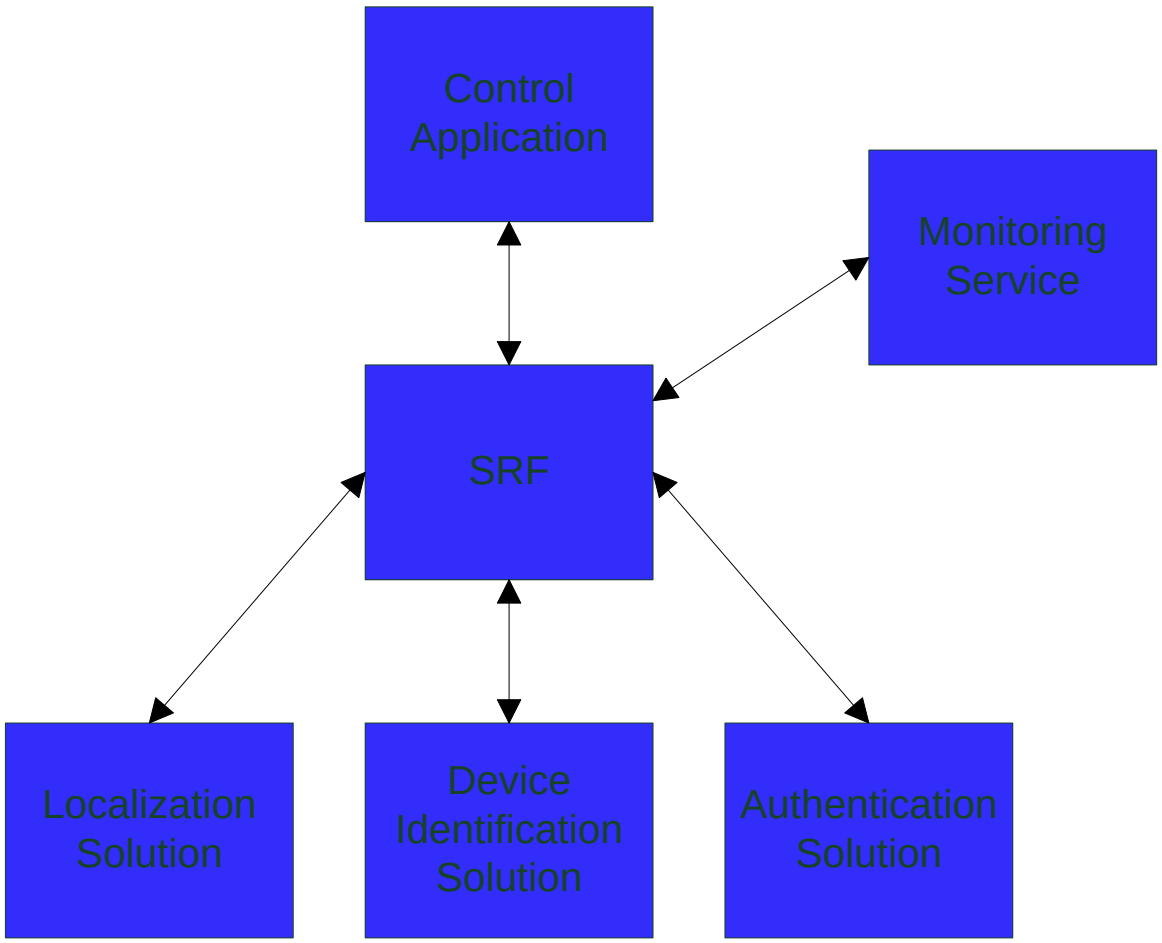


Figura 20 ACS: Conexiones

Las líneas de doble dirección simbolizan las conexiones reales existentes entre los distintos módulos. Como se puede observar todo va conectado al SRF, que es el que organiza las conexiones entre los distintos elementos. En cambio en la Figura 21 podemos ver cómo el flujo de información no se corresponde con las conexiones vistas en la Figura 20, ya que el SRF realmente enlaza estos elementos.

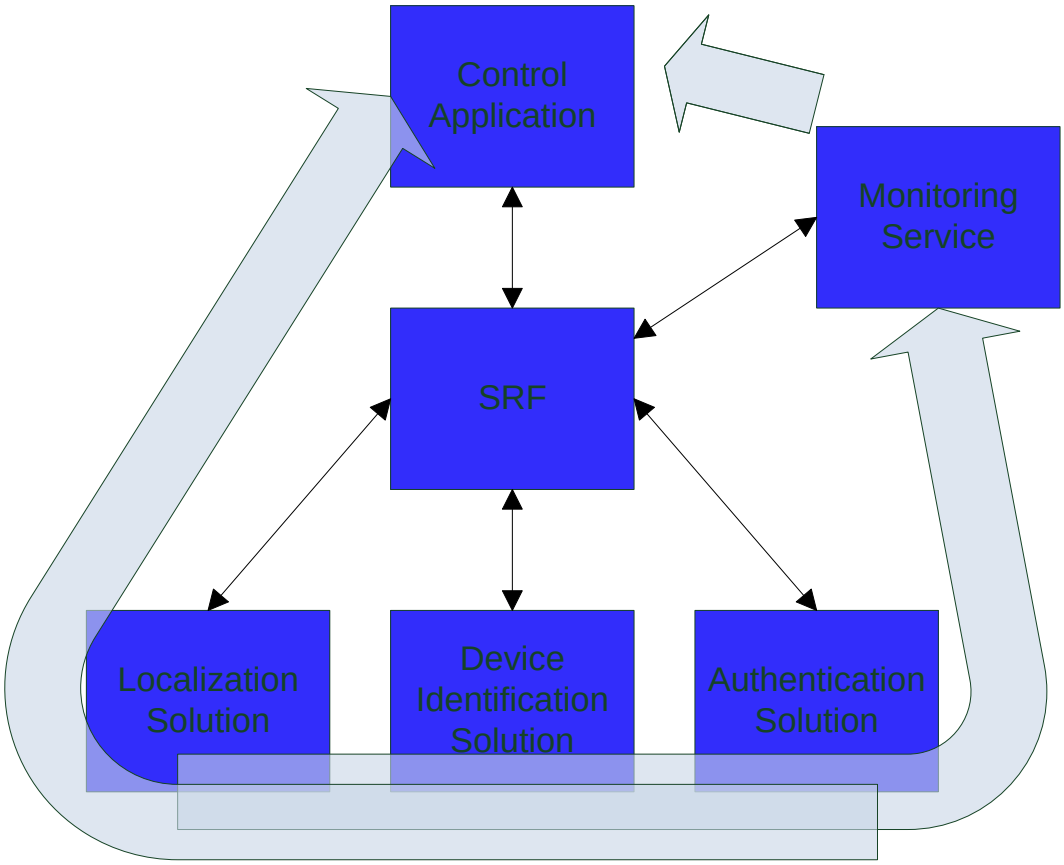


Figura 21 ACS: Flujo de información

Ahora podemos ver, en la Figura 21, cómo la información va desde las soluciones a la Aplicación de Control, y desde las mismas al Monitoring service y a su vez a la Aplicación. Por tanto podemos decir que el flujo de información es periférico.

Ahora pasamos a describir cada uno de los elementos presentes en la Figura 20 y en la Figura 21.

4.2.2.1 Control Application

Como hemos ido diciendo a lo largo de la memoria es la aplicación que controla todos los aspectos relacionados con la seguridad, así como el Firewall. Las peticiones de carga de las soluciones las hace al SRF y éste le proporciona las distintas soluciones. También recibe información del servicio de monitorización para saber en qué estado se encuentran las soluciones cargadas por el SRF. Toda la Aplicación de Control está desarrollada en Java y tiene un interfaz gráfico para facilitar la interacción con el administrador. Usa bases de datos en MySQL para almacenar los datos.

En la aplicación de Control se pueden diferenciar tres hilos de ejecución principales a los que hemos denominado NetClientsManager, AccessMeasurer y FirewallControl.

- NetClientsManager: Este hilo es el encargado de descubrir a cada uno de los clientes conectados a la red de acceso, así como almacenar la información necesaria de cada cliente.
- AccessMeasurer: La tarea principal de este hilo es la de realizar las medidas oportunas de seguridad, identificación de dispositivo, identificación de usuario e identificación de la zona de acceso, para cada uno de los clientes administrador por NetClientsManager.
- FirewallControl: Este hilo tiene la tarea de actualizar las reglas del firewall, dando acceso a determinados recursos a aquellos clientes cuyas medidas de seguridad se lo permitan.

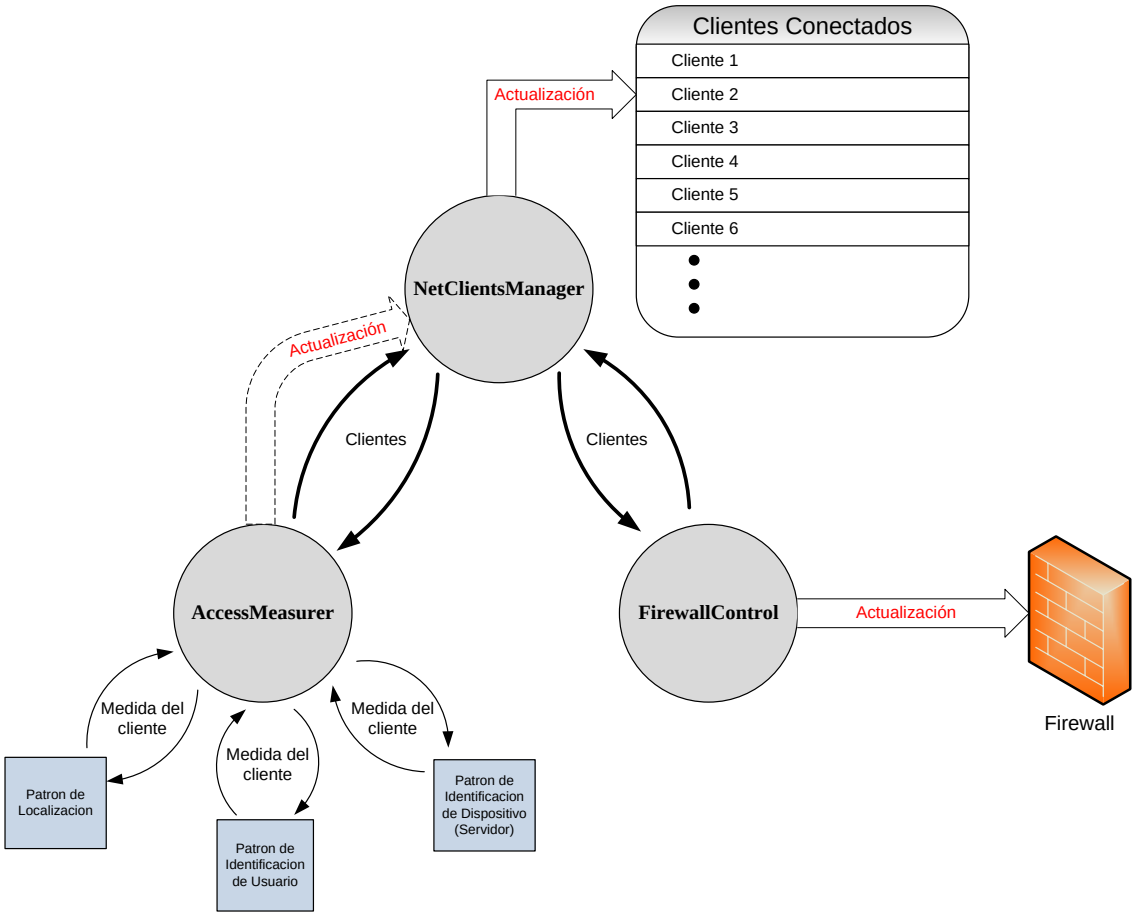


Figura 22 Hilos de ejecución en la Aplicación de Control

Tal y como vemos en la Figura 22 los hilos de ejecución AccessMeasurer y FirewallControl se nutren de la información proporcionada por NetClientsManager, en concreto de los clientes que este tiene almacenados. AccessMeasurer además actualiza el estado de los clientes con las medidas de seguridad de cada uno de ellos.

4.2.2.1.1 NetClientsManager

Como se ha explicado antes, NetClientsManager es el encargado de administrar los clientes existentes en la red.

La tarea de descubrimiento de clientes se realiza mediante multidifusión, esta estrategia permite el envío eficiente de mensajes sobre cada enlace de la red de una vez.

Los clientes, para poder recibir dichos mensajes, deben estar asociados a los que se denomina "grupo multicast". Un grupo multicast tiene asociado una dirección de red determinada.

Según la versión actual del protocolo de internet, IPv4, se reservan las direcciones de tipo D para la multidifusión. Las direcciones de tipo D son aquellas en las cuales los 4 bits más significativos son '1110' (224.x.x.x). El trafico multicast tiene lugar en el nivel de transporte UDP. Consiste en definitiva en el envío de datagramas UDP.

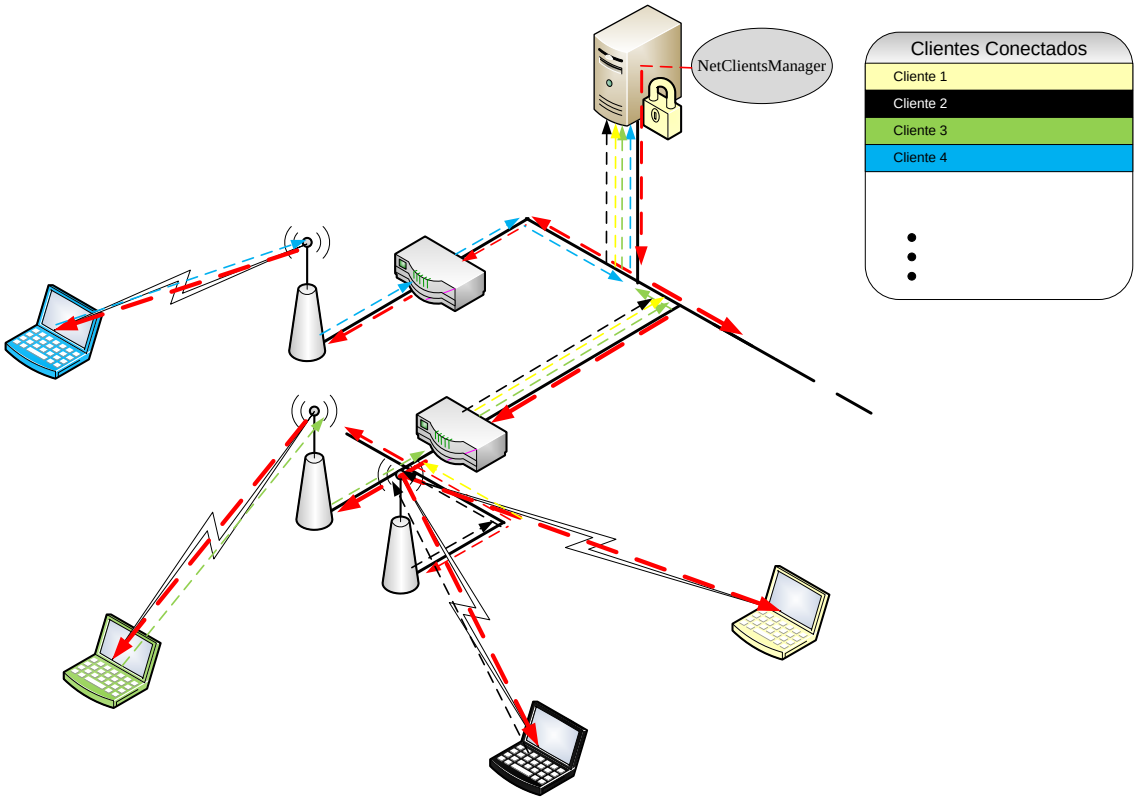


Figura 23 Flujo de los datagramas multicast en la red

En la Figura 23 se puede apreciar cómo se transmiten los datagramas multicast, indicados por flechas punteadas rojas. Los datagramas de difunden por toda la red, la infraestructura de red de acceso, hasta llegar a cada uno de los clientes.

Cada cliente, al recibir un datagrama multicast responde con otro datagrama, en este caso unicast, con destino la aplicación de control, el origen del datagrama multicast recibido.

De esta manera el NerClientsManager puede tener una tabla actualizada de cada uno de los clientes que pertenezcan a la red de acceso. De la siguiente manera:

Cliente	IP	MAC	Ultima Respuesta
Cliente 1	192.70.1.1	11:11:11:11:11:11	t_1
Cliente 2	192.70.1.2	22:22:22:22:22:22	t_2
Cliente 3	192.70.1.3	33:33:33:33:33:33	t_3
Cliente 4	192.70.1.4	44:44:44:44:44:44	t_4
●			
●			
●			

Tabla 4 Tabla de clientes

Se puede observar en la Tabla 4 que además de contener una lista de clientes, también contiene una serie de propiedades de cada uno de ellos. En esta fase, las propiedades establecida por NetClientsManager son las siguientes:

- IP- Dirección IP de cliente, es recogida del datagrama unicast recibido por la aplicación de control
- MAC: Dirección MAC del cliente, este dato es enviado como un dato dentro del datagrama unicast, enviado por el cliente.
- Ultima Respuesta: Es una marca de tiempo que indica la última vez que se ha recibido una respuesta del correspondiente cliente.

El hilo de ejecución NetClientsManager no solo debe descubrir los clientes existentes en la red, también debe tener actualizada la tabla de clientes. Para conseguir esto, los datagramas multicast se envían periódicamente cada cierto tiempo (Tm), recibiendo respuesta de los clientes para cada uno de ellos, actualizando así el campo de Ultima Respuesta de los clientes antiguos y añadiendo los nuevos clientes.

Al mismo tiempo, NetClientsManager debe borrar de la tabla aquellos clientes que considera muertos, es decir, que ya no están presentes en la red de acceso. Para ello, se establece un umbral de tiempo de tiempo de respuesta máximo (Tc), tiempo de caducidad, para el que se considerará que el cliente ya no está presente.

$$Si(t_{actual} - t_i > Tcaducidad) \rightarrow Cliente\ _i\ Muerto$$

NetClientsManager está compuesto a su vez por otros tres hilos de ejecución:

- El propio NetClientsManager: Se encarga de comprobar periódicamente la existencia de algún cliente muerto para proceder con su borrado de tabla de clientes.
- ClientsDiscoverer: Este hilo se encuentra en todo momento atento a la recepción de algún datagrama proveniente de algún cliente para así proceder con la actualización del tiempo de respuesta, en caso de ser un cliente ya existente, o con la inserción de un nuevo cliente en la tabla de clientes.
- Multicaster: La tarea de este hilo es la de enviar periódicamente datagramas multicast a la red de acceso con el objetivo de que los clientes conectados lo reciban y envíen por su parte la respuesta.

En el siguiente diagrama de tiempos podemos ver cómo se comporta cada una de estas ejecuciones para conseguir el objetivo explicado.

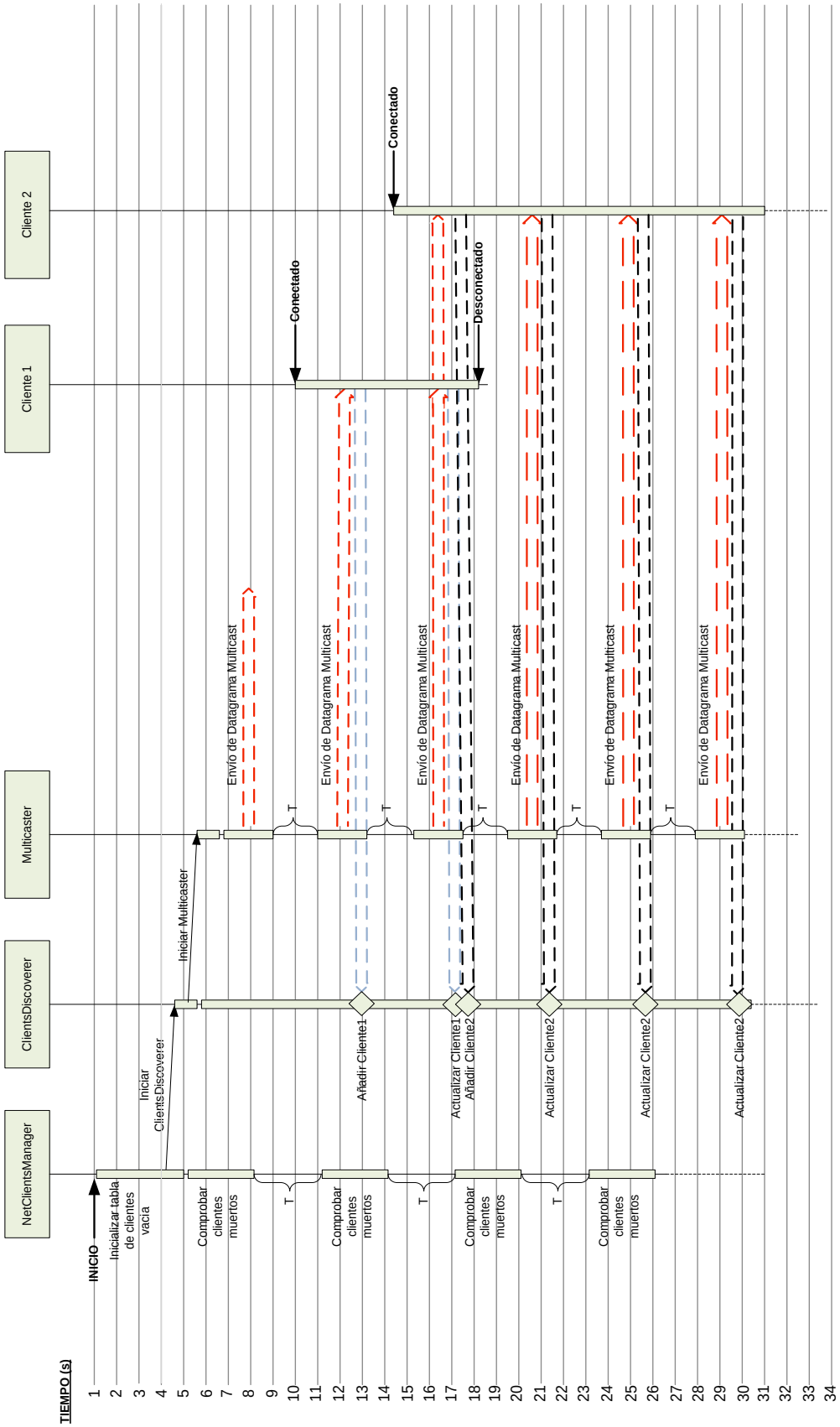


Figura 24 Esquema de tiempos de la Aplicación de Control

Observando la Figura 24 podemos deducir los cambios que sufre la tabla de los clientes detectados en el sistema, teniendo en cuenta que el tiempo de caducidad es de 6 segundos:

- t=1s: Creación de la tabla, inicialmente vacía.
- t=13s: Inserción del Cliente1 en la tabla. Ultima Respuesta= 13.
- t=17s: Actualización de la marca de tiempo del Cliente1. Ultima Respuesta= 17.
- t=18s: Inserción del Cliente2 en la tabla. Ultima Respuesta= 18.
- t=21s: Actualización de la marca de tiempo del Cliente2. Ultima Respuesta= 21.
- t=24s: Borrado del Cliente2. Se cumple 24-17 > 6.
- t=26s: Actualización de la marca de tiempo del Cliente2. Ultima Respuesta= 26.
- t=30s: Actualización de la marca de tiempo del Cliente2. Ultima Respuesta= 30.

4.2.2.1.2 AccessMeasurer

La tarea de este proceso, como se puede deducir del propio nombre, es la de realizar medidas de cada uno de los clientes conectados a la red. Estos son obtenidos del proceso explicado anteriormente, NetClientsManager, que contendrá una lista actualizada de los clientes presentes.

Cuando hablamos de medidas de seguridad nos estamos refiriendo a los valores de la seguridad que nos proporciona cada uno de los tres patrones implementados:

- Medida de Identidad de usuario.
- Medida de Identidad de dispositivo.
- Medida de Localización.

El resultado final de una medida es un número entero positivo o cero, el cual indica el nivel de seguridad adquirido por la propiedad ambiental medida. Cuando mayor sea la medida, más seguro se considera dicha propiedad.

De esta manera, completaremos la tabla de clientes generada por NetClientsManager con tres campos más, de la siguiente manera:

CLIENTES						
Cliente	IP	MAC	Ultima Respuesta	Medida de Usuario	Medida de Dispositivo	Medida de Localizacion
Cliente 1	192.70.1.1	11:11:11:11:11:11	t_1	x	x	x
Cliente 2	192.70.1.2	22:22:22:22:22:22	t_2	x	x	x
Cliente 3	192.70.1.3	33:33:33:33:33:33	t_3	x	x	x
Cliente 4	192.70.1.4	44:44:44:44:44:44	t_4	x	x	x
•						
•						
•						

Tabla 5 Tabla de clientes (II)

En la Tabla 5 las X corresponden con el número resultado de la medida realizada por este proceso.

Como hemos empleado el proyecto SERENITY esto quiere decir que esta aplicación se despreocupa totalmente de soluciones de seguridad, es decir la Aplicación de Control pide las soluciones al SRF y es éste el que se las proporciona. Con este planteamiento logramos que el proyecto sea más seguro, dinámico y escalable.

La primera tarea de este proceso es la de pedir las soluciones necesarias al SRF. Tenemos distintas maneras de pedir la solución:

- Pedir un S&DClass: estos significa pedir la solución en su sentido más amplio. Poniendo un ejemplo, para la solución de localización, pedir una S&DClass denominada "LocalizationMeasurer". Será el SRF el que decida cuál es el S&DPattern, de entre los disponibles, óptimo para la situación concreta.
- Pedir un S&DPattern: sería una petición más concreta. Siguiendo con el ejemplo anterior, podemos pedir un S&DPattern denominado "EkahauMeasurer", que pertenece la clase "LocalizationMeasurer", e implementa un patrón de medición de seguridad de localización mediante Ekahau. A su vez este S&DPattern podría tener varias implementaciones, el SRF elegiría la que considerase más estable según las condiciones en las que se encuentre.
- Pedir un S&DImplementation: esta petición es la más concreta de todas, significa pedir una implementación en concreto de un S&DPattern. Esta no es la opción más recomendable ya que no dejaríamos que fuera SERENITY el que escogiera la implementación que más se adecuara a los requisitos de seguridad. Seríamos nosotros los que pediríamos una Solución concreta.

En nuestro caso, y hemos explicado en la sección 3.2, hemos desarrollado tres patrones, uno por cada medida de seguridad. Por lo tanto será indiferente el tipo de petición realizada ya que todas derivarán en la misma S&DImplementation. El resultado final es el mismo componente ejecutable (EC).

En el siguiente esquema podemos ver el proceso completo, desde la petición S&D hasta la comunicación entre la aplicación y el componte ejecutable, para cada uno de los patrones solicitados:

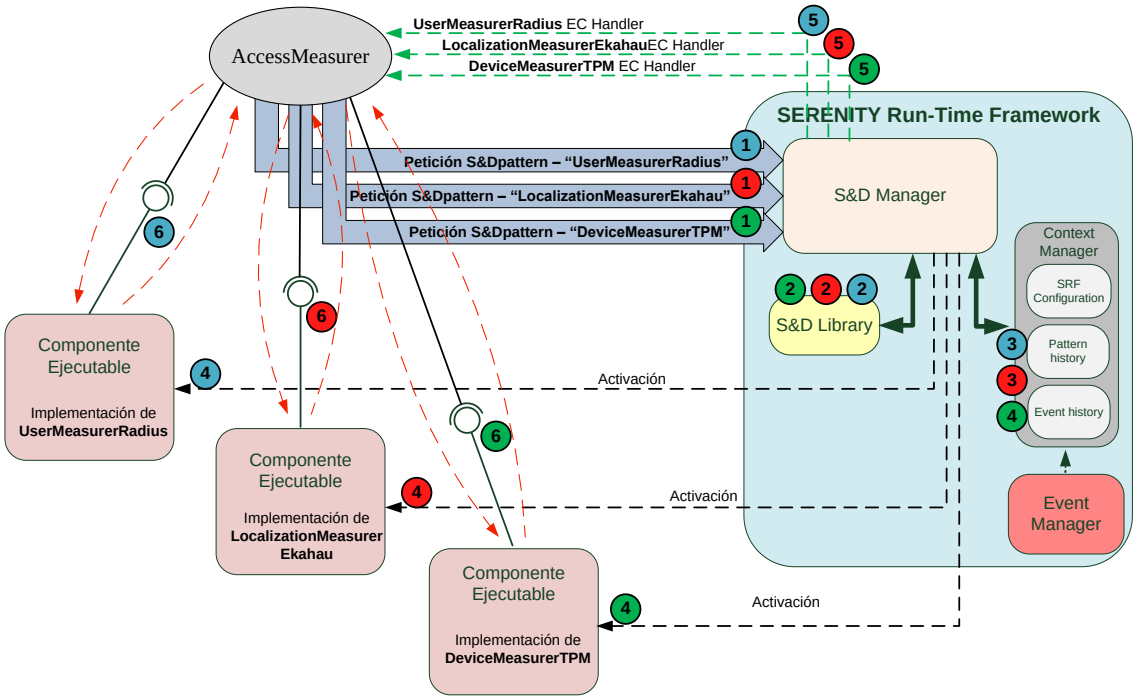


Figura 25 Proceso de carga de patrones S&D

Este proceso es llevado a cabo en el arranque del AccessMeasurer, su objetivo es proveerse de los patrones de seguridad necesarios:

- Para la solución de medida de Usuario se hace la petición de un patrón S&Dpatter denominado UserMeasurerRadius.
- Para la solución de medida de Localizacion se hace la petición de un patrón S&Dpatter denominado LocalizationMeasurerEkahau.
- Para la solución de medida de Dispositivo se hace la petición de un patrón S&Dpatter denominado DeviceMeasurerTPM.

En cada una de las peticiones se realizan, de manera resumida, las siguientes acciones:

1. La aplicación envía la petición S&DPattern del patrón necesario al Serenity RunTime Framework, recibida por el componente S&DManager.
2. El SRF comprueba la disponibilidad del S&DPattern solicitado en el S&DLibrary, obteniendo todas las S&DImplementations relacionadas con la petición.
3. Con ayuda del componente Context Manager se deduce cuál de las S&Dimplementations es la óptima para el contexto actual, se evalúan el historial de patrones, el historial de eventos y la propia configuración del SRF.
4. Una vez obtenido la S&Dimplementation óptima, se produce la activación del componente ejecutable que implementa la solución solicitada.
5. El SRF le da el mando del componente ejecutable a la aplicación, por medio de lo que se conoce como ECHandler.

6. Obtenido el ECHandler, la aplicación ya puede comunicarse con el componente ejecutable.

La comunicación con un componente ejecutable se realiza mediante lo que se denomina callOperation, una llamada de operación. La funcionalidad principal de un patrón de estilo "Measurer" es la de realizar la medición de un factor de seguridad (localización, autenticación de usuario o identificación de dispositivo), con lo que están provistas de una callOperation llamada getSecurityLevel. la respuesta del componente ejecutable es un numero entero positivo o cero que representa el nivel de seguridad de que dispone el usuario..

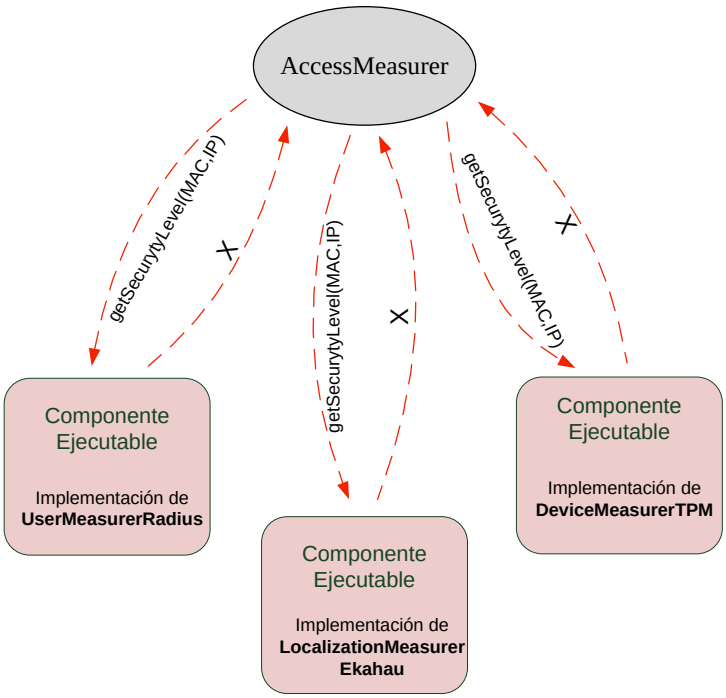


Figura 26 Comunicación con los Componentes Ejecutables

Como se puede ver en la Figura 26, AccessMeasurer hace uso de los componentes ejecutables mediante la callOperation getSecurityLevel, para obtener la medida de seguridad para un cliente dada su dirección IP o dirección MAC.

Las tres mediciones se hacen periódicamente para cada uno de los clientes encontrados en la tabla de clientes conectados generada por NetClientsManager.

Un componente ejecutable puede tener más de un callOperation para dar funcionalidad extra a la aplicación, aunque no sean en sí una operación de seguridad. Sirva como ejemplo una callOperation del patrón UserMeasurerRadius denominada "getUserName" para obtener el nombre del usuario que corresponde con el cliente conectado.

En apartados siguientes se explica detalladamente el desarrollo de cada uno de los patrones utilizados por la aplicación así como todas las callOperation disponibles para cada uno de ellos.

El Serenity RunTime Framework lleva una monitorización de cada componente ejecutable instanciado, gracias al Servicio de Monitorización, ver apartado 4.1.2.2 . Esto quiere decir que cuando el sistema detecta un error en algún componente ejecutable, una violación de una regla de monitorizacion, el SRF puede parar el EC correspondiente, reemplazarlo por otro según crea conveniente, o simplemente comunicárselo a la aplicación, en nuestro caso el sistema empleado es el de comunicar a la Aplicación de Control sin parar el Componente Ejecutable.

4.2.2.1.3 FirewallControl

Este proceso se encarga de actualizar las reglas de firewall para cada uno de los clientes, teniendo en cuenta las medidas determinadas por el proceso AccessMeasurer. Este hilo se nutre de la tabla de clientes conectados.

Lo que se pretende con la actualización de las reglas de firewall es dar acceso a determinados recursos localizados en la red interna, la red protegida. Un recurso puede ser un host o un conjunto de hosts. Además, un recurso puede estar limitado también a determinados protocolos, como en los casos en los que el recurso es un servicio. Un protocolo está definido por uno o más puertos, UDP o TCP.

Cada recurso tiene asociado unos niveles mínimos de seguridad que debe de cumplir un cliente para poder tener acceso a él. Estos niveles de seguridad se se basan en las medidas de las propiedades ambientales, ya mencionadas, en las que se encuentra un cliente: nivel mínimo de usuario, nivel mínimo de dispositivo, nivel mínimo de localización.

RECURSOS						
Recurso	IP	Tipo	Protocolos	Medida de Usuario	Medida de Dispositivo	Medida de Localizacion
Recurso 1	10.100.32.2	Interno	[http]	1	1	2
Recurso 2	88.4.235.154	externo	[ftp]	3	2	2
Recurso 3	0.0.0.0	externo	[http;https]	0	0	1
Recurso 4	10.100.8.0/22	interno	[]	1	1	1
Recurso 5	10.100.4.179	interno	[svn;ssh]	2	1	2
•						
•						

Tabla 6 Tabla de Recursos

La Tabla 6 es un ejemplo de la tabla de recursos dados de alta en la aplicación de control. Necesariamente estos recursos se encuentran fuera de la red de acceso, pudiendo encontrarse tanto en la red interna corporativa como fuera, en Internet.

Como ya hemos introducido anteriormente, el firewall utilizado se trata de una solución software llamada IPTABLES de NETFILTER, disponible en los últimos kernels de LINUX. Se encuentra en una maquina aparte de la maquina en la que reside la aplicación de control y ambas se comunican por medio de SSH (ver Figura 27).

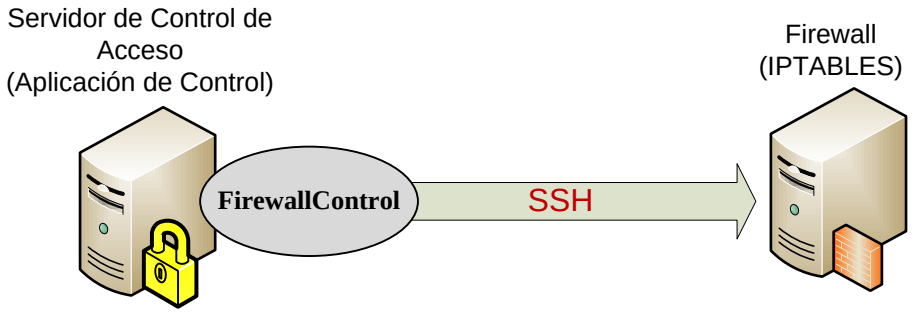


Figura 27 Comunicación entre Aplicación de Control y Firewall

De esta manera la aplicación de control podrá tanto configurar la máquina firewall como añadir nuevas reglas de filtrado.

En el arranque de la aplicación, en concreto en el arranque del hilo FirewallControl, se produce la configuración de la máquina firewall (ver Figura 28):

- Dirección IP del adaptador conectado a la red interna.
- GateWay utilizado para dar acceso a aquellos recursos fuera de la red interna.

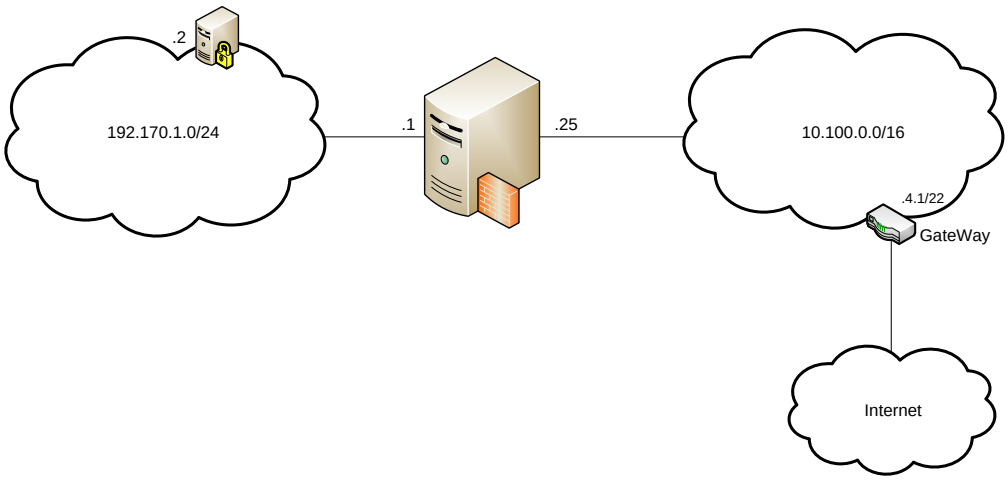


Figura 28 Red interna, red externa y Firewall

La aplicación de control contiene un archivo de configuración (ver ANEXO B) en el que se debe de especificar los siguientes parámetros:

- Dirección IP y máscara de red de la maquina firewall y el adaptador conectado a la red de acceso (en el ejemplo 192.170.1.1). Todo esto es necesario para que FirewallControl pueda establecer comunicación SSH con la maquina firewall.
- Usuario y contraseña de la maquina firewall necesario para iniciar la sesión SSH con la maquina firewall. Este usuario debe tener los permisos necesarios para ejecutar los comandos de configuración de red.
- Dirección IP que queramos que tenga la maquina firewall en su adaptador conectado a la red interna.
- Dirección IP del GateWay utilizado para enrutar los paquetes dirigidos a recursos fuera de la red interna
- Dirección o direcciones de los servidores DNS utilizados para la resolución de nombres, necesario siempre y cuando estos se encuentren fuera de la red de acceso.
- Dirección IP y mascara de la red interna (en el ejemplo 10.100.0.0/16).

En el arranque del hilo FirewallControl se producen las siguientes acciones:

1. Se inicia la sesión SSH con la maquina firewall según los parámetros introducidos en el fichero de configuración.
2. Se configura el adaptador de red conectado a la red interna mediante este comando Shell:

   #ifconfig eth0 (dirección IP) netmask (mascara)

   Se ha tenido en cuenta que el interfaz eth0 es el adaptador correspondiente.
3. Se añade la ruta por defecto, Gateway:

   #route add default gw (dirección IP GateWay)

4. Se inicializa el firewall IPTABLES estableciendo las políticas por defecto y añadiendo las reglas necesarias en el inicio:

   #iptables –P INPUT DROP
   #iptables –P OUTPUT FORWARD
   #iptables –P FORWARD DROP

   Mediante estas tres reglas se establecen la política por defecto de desechar cualquier paquete dirigido a la propia maquina firewall (INPUT), saliente de la propia maquina (OUTPUT), y cualquier paquete proveniente de la red de acceso dirigido hacia la red interna (FORWARD).

   Debido a la política por defecto de desechar cualquier paquete INPUT, podríamos perder la sesión SSH iniciada, con lo que se hacen necesarias las siguientes reglas:

   #iptables -A INPUT -s (direccion IP del servidor de acceso) -p tcp --dport 22 -j ACCEPT
   #iptables -A OUTPUT -d (direccion IP del servidor de acceso) -p tcp --sport 22 -j ACCEPT

El primer comando añade una regla (-A) para el tráfico de entrada (INPUT) proveniente de la dirección IP del servidor de acceso (-s), hacia el puerto (-dport) TCP (-p) número 22 (utilizado en el protocolo SSH) y la acción (-j) de aceptar (ACCEPT).
En el segundo comando ocurre lo mismo pero de manera opuesta, añade una regla que permite el tráfico de salida hacia el servidor de control de acceso proveniente del puerto TCP 22.

En el caso de que exista en el fichero de configuración algún servidor DNS, lo que quiere decir que se deben de utilizar estos servidores para la resolución de nombres en la red de acceso, debemos de permitir el tráfico de paquetes DNS entre la red acceso y cada uno de los servidores DNS indicados:

#iptables -A FORWARD -s (direccion de red acceso) -d (direccion IP DNS) -p udp --dport 53 -j ACCEPT
#iptables -A FORWARD -s (direccion IP DNS) -d (direccion de red acceso) -p udp --sport 53 -j ACCEPT
#iptables -A FORWARD -s (direccion de red acceso) -d (direccion IP DNS) -p tcp --dport 53 -j ACCEPT
#iptables -A FORWARD -s (direccion IP DNS) -d (direccion de red acceso) -p tcp --sport 53 -j ACCEPT

Mediante estos cuatro comandos añadimos las reglas cuatro reglas necesarias para permitir la redirección de tráfico (FORWARD) DNS entre la red de acceso y un solo servidor DNS.
El protocolo DNS usa el puerto 53, tanto TPC como UDP.

Una vez iniciado y configurado el firewall IPTABLES, el hilo FirewalControl procede a la lectura de los clientes conectados para la actualización de las reglas de filtrado, con el objetivo de dar acceso a los recursos permitidos para cada cliente.

Se considera un recurso permitido para un cliente a aquel cuyas medidas mínimas requeridas de usuario, dispositivo y localización son inferiores o igual a las medidas obtenidas respectivamente por el cliente.

Imaginemos el cliente de la Tabla 7, con las medidas de seguridad ya realizadas por AccessMeasurer:

Cliente	IP	MAC	Ultima Respuesta	Medida de Usuario	Medida de Dispositivo	Medida de Localizacion
Cliente 1	192.70.1.1	11:11:11:11:11:11	t_1	1	1	1

Tabla 7 Cliente identificado

Si la tabla siguiente corresponde con los recursos registrados en la aplicación de control:

RECURSOS						
Recurso	IP	Tipo	Protocolos	Medida de Usuario	Medida de Dispositivo	Medida de Localizacion
Recurso 1	10.100.32.2	Interno	[http]	1	1	2
Recurso 2	88.4.235.154	externo	[ftp]	3	2	2
Recurso 3	0.0.0.0	externo	[http;https]	0	0	1
Recurso 4	10.100.8.0/22	interno	[]	1	1	1
Recurso 5	10.100.4.179	interno	[svn;ssh]	2	1	2

Tabla 8 Recursos registrados

FirewallControl comprueba cuáles de los recursos están permitidos para el cliente, obteniendo el siguiente resultado:

RECURSOS PERMITIDOS						
Recurso	IP	Tipo	Protocolos	Medida de Usuario	Medida de Dispositivo	Medida de Localizacion
Recurso 3	0.0.0.0	externo	[http;https]	0	0	1
Recurso 4	10.100.8.0/22	interno	[]	1	1	1

Tabla 9 Recursos permitidos para el cliente

En este punto, deberá añadir las reglas necesarias al firewall para permitir ambos recursos (recursos 3 y 4 en el ejemplo):

- Recurso 3: se trata de un recurso externo que permite el acceso a cualquier maquina en internet (0.0.0.0). Las reglas necesarias para dar acceso a un recurso así tienen una estructura característica, diferente a las demás:
    - Se desecha todo el tráfico desde y hacia la red local en los puertos especificados por el recurso.
    - Se permite todo tráfico desde y hacia cualquier dirección (0.0.0.0).

  Necesariamente tiene que ser en el orden dado ya que las reglas se leen de arriba hacia abajo hasta que el paquete IP evaluado concuerde con una regla. Si fuese en orden inverso, se permitiría el trafico a cualquier maquina, tanto dentro como fuera de la red interna.

  Siguiendo con el ejemplo, las reglas añadidas para dar acceso al Recurso 3 serian las siguientes:

  #iptables -A FORWARD -d (IP del cliente) -s (IP de la red interna)-p tcp --sport 80 -j REJECT
  #iptables -A FORWARD -s (IP del cliente) -d (IP de la red interna) -p tcp --dport 80 -j REJECT
  #iptables -A FORWARD -d (IP del cliente) -p tcp --sport 80 -j ACCEPT
  #iptables -A FORWARD -s (IP del cliente) -p tcp --dport 80 -j ACCEPT

Con estas reglas daríamos acceso a internet en el protocolo HTTP, puerto TCP 80. Para HTTPS sería de igual manera con puerto TCP 443.

En este tipo de reglas nos encontramos con un inconveniente, si existe una regla posterior que dé acceso a un recurso en la red interna en el mismo puerto que el recurso externo, esa regla nunca será cumplida debido al orden en el que se evalúan las reglas, de arriba hacia abajo. Veamos el ejemplo:

- Para el Recurso 4 se añaden las siguientes reglas:

  #iptables -A FORWARD -d (IP del cliente) -s 10.100.8.0/22 -j ACCEPT
  #iptables -A FORWARD -s (IP del cliente) -d 10.100.8.0/22 -j ACCEPT

Si las reglas se han añadido en este mismo orden, el inconveniente es el siguiente: Si un pasa un paquete IP por el firewall con origen la IP del cliente y destino una máquina que se encuentra en la subred 10.100.8.0/22 y en el puerto TCP 80, este paquete será desechado a pesar de que el cliente tiene los permisos necesarios para acceder a él. Esto es así porque el paquete cumple la primera regla en el que se desecha todo paquete con origen la IP del cliente y destino toda la red interna en el puerto TCP 80.

Para solucionar el este problema, FirewallControl debe de llevar un registro de todos los recursos permitidos a un cliente, de manera que cada vez que se añadan las reglas necesarias para dar acceso a un recurso interno se realizan las siguientes acciones:

1. Se consulta en registro los recursos externos que el cliente tiene asociados.
2. Se eliminan las reglas asociadas con los recursos externos resultados de la anterior consulta.
3. Se añaden las reglas necesarias para dar acceso al recurso interno y se añade este al registro.
4. Se vuelven a añadir las reglas asociadas con los recursos externos previamente eliminados.

De esta manera conseguimos que las reglas correspondientes a recursos externos estén siempre en última posición, solucionando así el problema explicado.

El registro que asocia clientes con recursos es necesario también para conocer las reglas existentes en el firewall y así: no repetirlas, borrar aquellas reglas asociadas a un cliente que ya no está conectado y borras las reglas que dan acceso a un recurso a determinado cliente que ya no tiene los permisos necesarios para acceder a dicho recurso debido a una variación en una de sus medidas de seguridad.

4.2.2.1.4 Requisitos cubiertos por la Aplicación de Control

De la lista de requisitos de la sección 3.3, los siguientes son los que cubre la Aplicación de Control:

- Debe haber una política de seguridad definida por el usuario, la localización y el dispositivo, y debe estar guardada en un repositorio o en un servidor. Este repositorio debería estar apropiadamente asegurado para evitar que accedan usuarios no permitidos. De hecho solamente la Aplicación de Control de Seguridad debería tener acceso al repositorio de política de seguridad.
- La Aplicación de Control debe controlar el acceso a los recursos protegidos.
- En el caso de que se trate de información, ésta debe estar adecuadamente archivada para garantizar su confidencialidad y su integridad.
- El empleado debe ser previamente registrado en la Aplicación de Control de Seguridad, así como el dispositivo. Así aseguramos una correcta identificación y autenticación.
- Debe haber reglas definidas para la identidad de los empleados y su perfil. Estas reglas deben ser accesibles en la Aplicación de Control de Seguridad. EL perfil de usuario está relacionado con los usuarios en bas al nivel de seguridad que tienen cada uno de ellos, en nuestro caso definimos tres niveles: visitante, becario y trabajador.

4.2.2.2 SRF

Es la instancia principal del proyecto SERENITY donde se cargan las distintas soluciones S&D a petición de la aplicación. Es el encargado de comunicar todas las instancias del ACS (Servidor de Control de Acceso), aunque se puede observar que el flujo real de información es entre los distintos componentes periféricos. El SRF viene implementado como parte del proyecto europeo SERENITY.

Es un elemento fundamental para que el enfoque SERENITY sea real. Tiene muchas funciones, aunque su principal tarea es la de comunicar todos los módulos del ACS. Otra de las principales funciones es la de proporcionar las Soluciones a la Aplicación de Control.

Como hemos venido diciendo, las aplicaciones piden soluciones al SRF y éste las conecta con las aplicaciones., Sin embargo todavía no se ha mencionado cómo se agregan éstas soluciones a la base de datos de soluciones del SRF.

Una vez que tenemos programado el Componente Ejecutable hay que generar tres archivos XML (ver ANEXO C) para poder cargarlo en el SRF. También cabe la posibilidad de que hayamos implementado un capturador de eventos para el Componente

Ejecutable, en este caso el conjunto de Componente Ejecutable y el capturador de eventos se llama Esquema Integrado y se comporta de la misma manera que un EC. La diferencia es que el capturador de eventos envía eventos del EC al servicio de monitorización.

Por tanto tenemos tres archivos XML y un Componente Ejecutable o un Esquema Integrado para cada solución. Los tres archivos XML son la clase, el patrón y la implementación. Una vez que añadamos estos tres archivos al SRF, la implementación es la encargada de apuntar al directorio donde se encuentra el Componente Ejecutable. Por tanto el SRF ya sabe donde se encuentra cada uno de los Componentes Ejecutables que componen su abanico de Soluciones S&D.

Para hacer esto hay que ejecutar la consola del SRF. Una vez que estemos en la pantalla de la consola seleccionamos "add S&D artifact", como vemos en la Figura 29 y en la Figura 30.

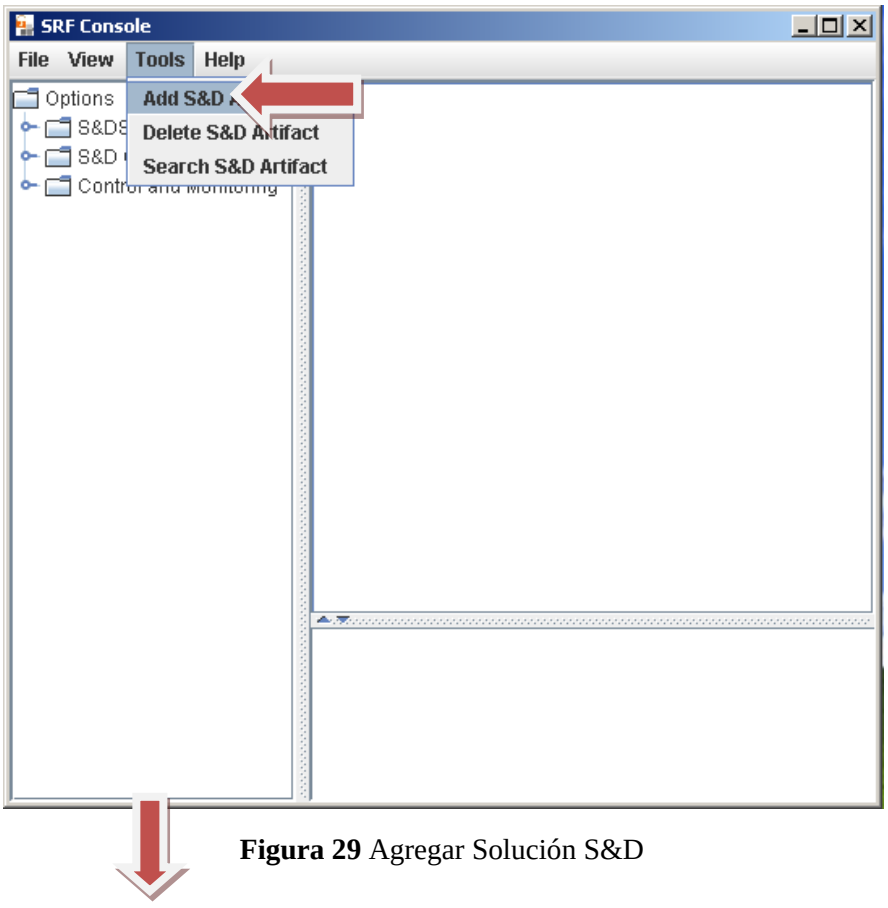


Figura 29 Agregar Solución S&D

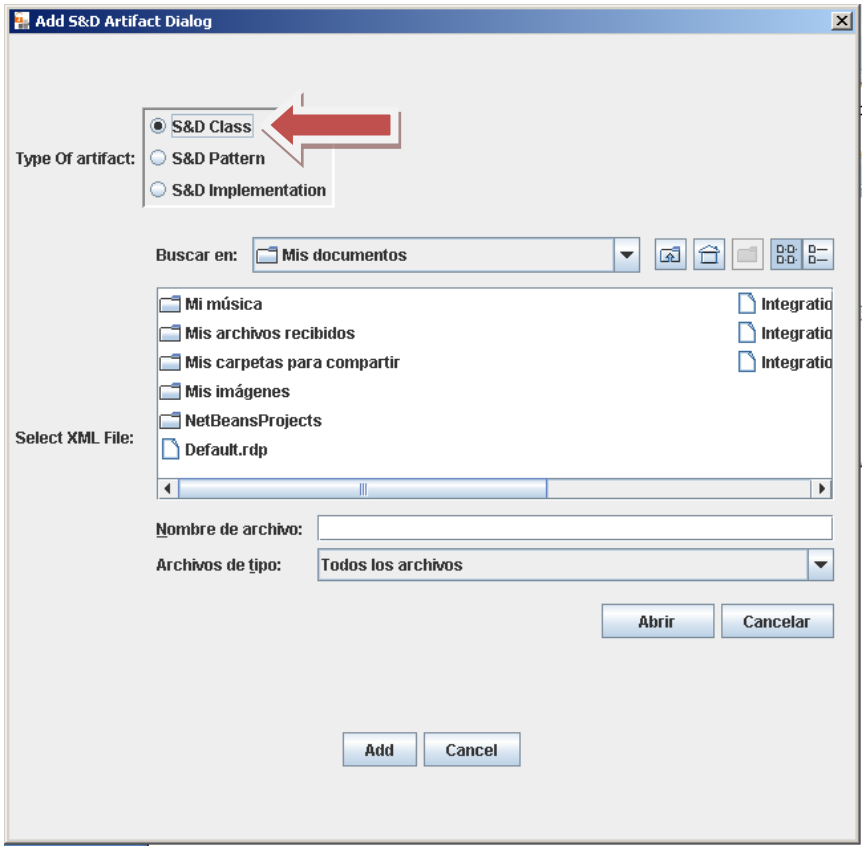


Figura 30 Agregar Solución S&D (Figura II)

Primero se añade la clase, luego el patrón y por último la implementación. Estos ficheros se encuentran en la carpeta XML de cada uno de los ECs.

Una vez hecho esto ya tenemos el Componente Ejecutable cargado en la base de datos del SRF y listo para ser solicitado por la Aplicación de Control.

4.2.2.3 Monitoring Service

Se encarga de verificar las reglas que le envían los Componentes Ejecutables, tanto precondiciones como en tiempo de ejecución. Las precondiciones se evalúan antes de cargar el Componente Ejecutable en el SRF, para verificar si se cumplen todos los requisitos para su funcionamiento y, una vez cargados, se evalúan las reglas de monitorización enviadas por los capturadores de eventos de cada Componente Ejecutable. Como podemos ver el flujo de información es desde las soluciones hasta la Aplicación de Control. Por otro lado es muy importante monitorizar en todo momento el estado general del ACS. Esta instancia ha sido implementada también dentro del marco del proyecto SERENITY y para su inclusión en este proyecto ha sido necesario el desarrollo de los Capturadores de Eventos asociados a cada Componente Ejecutable.

El servicio de Monitorización es un servicio que procesa unas reglas muy simples, por tanto las reglas que pretendamos monitorizar deben ajustarse a esta característica del servicio.

4.2.2.4 Soluciones S&D

Por último nos queda comentar las Soluciones S&D implementadas. En la siguiente figura observamos por qué elementos están compuestas las soluciones:

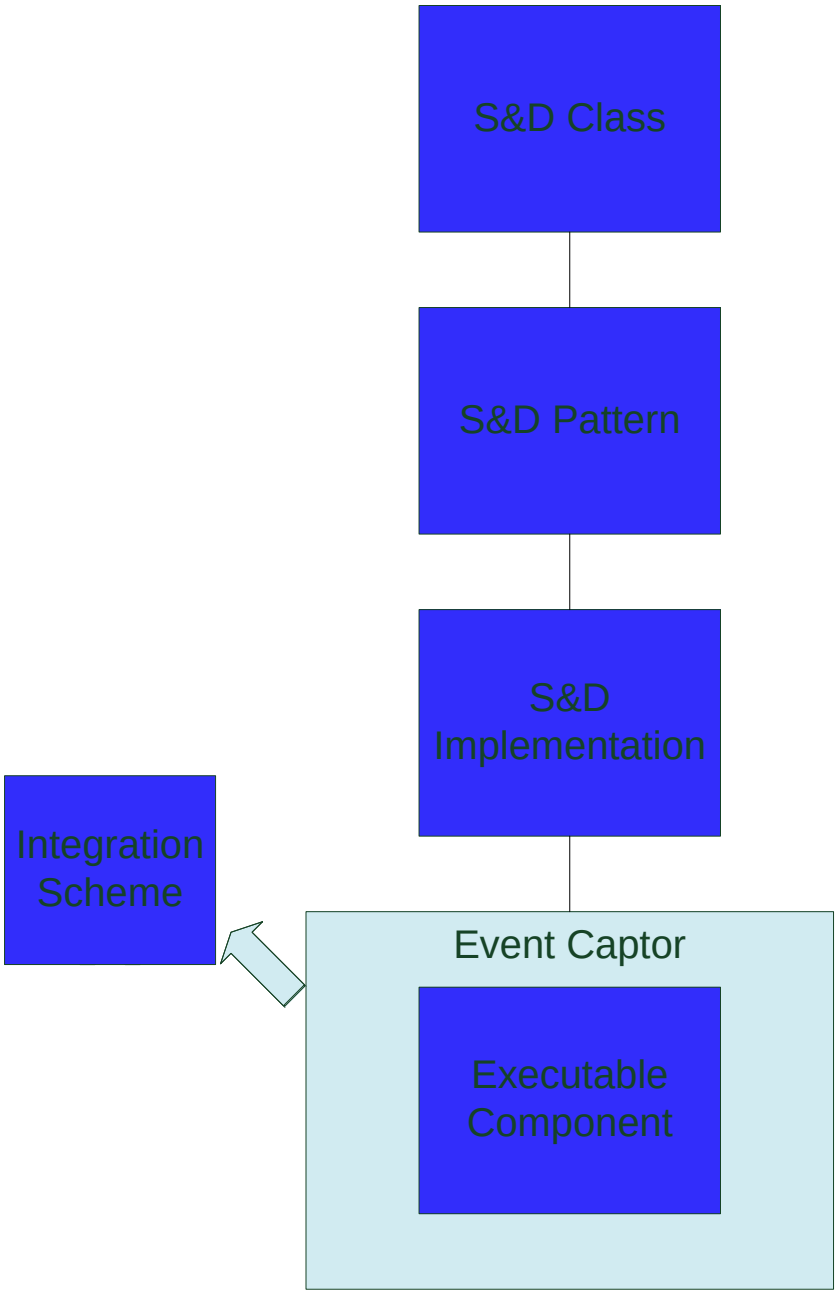


Figura 31 Solución S&D

En la Figura 31 se ve la clase, el patrón, la implementación, el componente ejecutable y también observamos que envolviendo al Componente Ejecutable se encuentra el capturador de eventos. El capturador de eventos no es imprescindible para que funcione la Solución S&D, es necesario para que funcione el servicio de monitorización del Componente Ejecutable.

Tanto la clase como el patrón como la implementación son archivos XML (ver ANEXO C) que sirven básicamente para incluir la solución en la base de datos del SRF (como hemos detallado en el apartado 4.2.2.2). El apartado 4.2.2.2, donde se describe cómo agregar una solución a la base de datos del SRF, está incluido en la sección del SRF ya que es una funcionalidad del mismo. Aquí nos limitamos a citar la referencia para clarificar esta sección.

El Componente Ejecutable y el Capturador de Eventos están programados en Java y el conjunto de ambos se denomina Esquema Integrado, como también hemos mencionado. Ahora pasamos a ver las tres soluciones explicadas por separado:

4.2.2.4.1 Solución de localización

La parte importante de la Solución es el Componente Ejecutable que ha sido desarrollado íntegramente en Java y con la ayuda de una API para comunicarnos con el Software propietario Ekahau, que es el encargado de localizar los dispositivos. Esta solución tiene también implementado su capturador de eventos, también en Java, que se encarga de comunicar el Componente Ejecutable (EC) con la Aplicación de Control (a través del SRF). Comprueba que la calidad de la localización obtenida está por encima de un determinado valor y también comprueba que la base de datos está funcionando correctamente.

El funcionamiento, una vez que está cargado el EC en el SRF, es el siguiente:

1. La Aplicación de Control solicita información relativa a la localización cada cierto tiempo (es configurable en el archivo de configuración). Junto a la petición envía la MAC del usuario al Componente Ejecutable.
2. El EC solicita la información relativa a esa MAC al servicio Ekahau, a través de la API en Java. El servicio Ekahau funciona con MACs para identificar a los usuarios.
3. El servicio devuelve diversas informaciones en su respuesta, incluida la zona localizada, la calidad de la localización, las coordenadas de localización y otras que no usamos en esta implementación.
4. El EC procesa la respuesta de Ekahau y selecciona la información relevante. Procede a consultar con su base de datos el nivel de seguridad de la zona.
5. Envía la zona y el nivel de seguridad de la zona a la Aplicación de Control.

La Aplicación de Control también tiene otra utilidad que necesita información de este EC, la localización precisa. Una vez que un usuario está localizado se habilita un botón en la parte de debajo de la ventana de usuarios de la Aplicación de Control, *Show Map* (Se puede ver en la Figura 32, en esa figura no está habilitado el botón al no tener seleccionado ningún usuario). Cuando se presiona la Aplicación envía otro tipo de petición al EC, la de localización precisa.

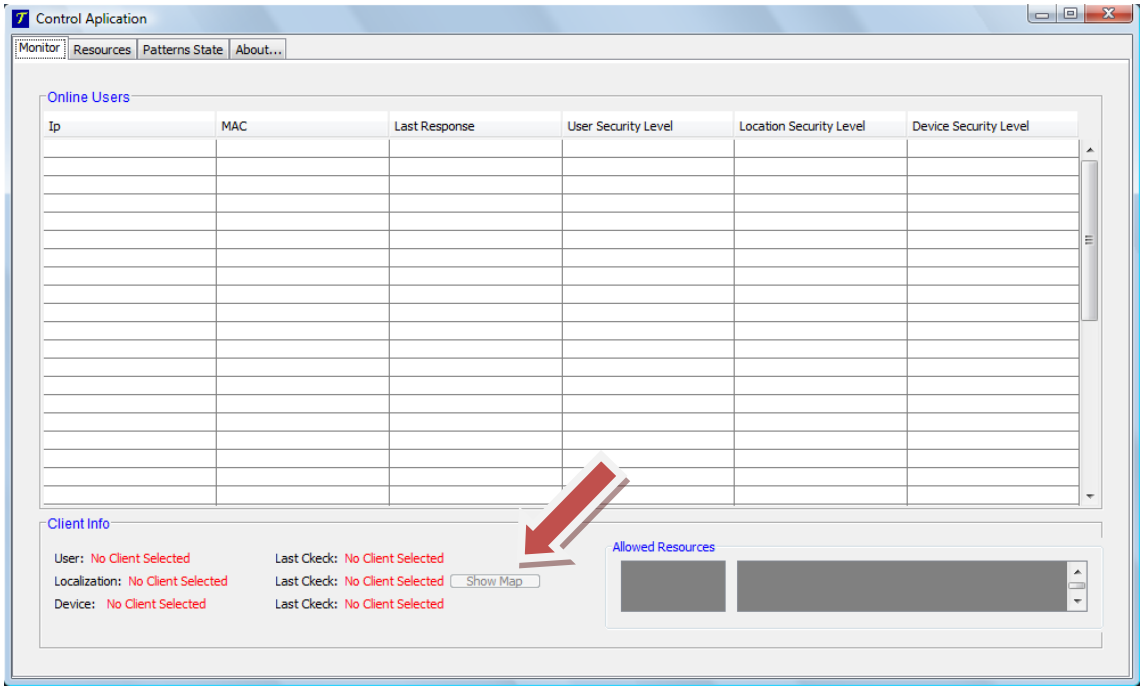


Figura 32 Aplicación de Control: Show Map

Cuando al Componente Ejecutable de localización le piden la localización precisa, responde un objeto *LocPrecisa* [29] con las coordenadas X e Y del dispositivo a localizar y el mapa de la zona. Esto es para poder representar en la Aplicación de Control al usuario en el mapa de la zona en la que se encuentra.

Por último la Aplicación de Control recibe este objeto *LocPrecisa* y pinta la posición del dispositivo en el mapa de la zona recibido también en el objeto *LocPrecisa*.

En definitiva este Componente Ejecutable tiene dos funciones:

- Devolver a la Aplicación de Control un valor de seguridad y la zona en la que se encuentra el dispositivo.
- Devolver a la Aplicación de Control la localización precisa, el objeto *LocPrecisa,* del dispositivo para que ésta sea capaz de situar al usuario en el mapa.

En cuanto al capturador de eventos, se basa en la siguiente funcionalidad:

- Precondición: que la base de datos esté operativa.

- Regla de monitorización: comprobar que la calidad de la localización es mayor que un umbral, si no se cumple la condición el servicio de monitorización envía un mensaje a la Aplicación de Control con el fin de que el administrador conozca el estado de los Componentes Ejecutables en cada momento.

4.2.2.4.2 Solución de identificación de dispositivos

Como ya hemos mencionado, la parte importante de una solución es el Componente Ejecutable que está desarrollado en Java y usa consultas a una base de datos MySQL.

En este caso hay dos Componentes Ejecutables, uno en el servidor y otro en el cliente. Los dos ECs tienen que ejecutarse en un orden establecido y esperarse el uno al otro. Como ya hemos mencionado en la sección 4.1.2.2 el SRF cuenta con una característica de negociación que sirve para estos fines.

Vamos a ver cada uno de los Componentes por separado.

4.2.2.4.2.1 EC del servidor TPMServerEC

Se ocupa de contactar con el EC en el cliente y proceder con el protocolo de Remote Attestation (descrito en el apartado 4.2.2.4.2.3).

Una vez realizado este protocolo el servidor conoce la identidad del dispositivo TPM del cliente, por tanto consulta con la base de datos y obtiene el nivel de seguridad del mismo.

4.2.2.4.2.2 EC del cliente TPMClientEC

En este caso el cliente espera la petición de iniciar el protocolo, una vez recibida se procede con el Remote Attestation entre cliente y servidor (descrito en el apartado 4.2.2.4.2.3).

Para la comunicación con el chip TPM del equipo hemos empleado la API *tpm4java* [30], que es en código abierto y proporciona una alta operabilidad con el chip. Los detalles de esta API se muestran en el apartado 2.3.4.1.

4.2.2.4.2.3 Remote Attestation

Anteriormente hemos introducido el término y ahora procedemos a un análisis más exhaustivo del protocolo. Este protocolo ha sido desarrollado para este proyecto y más específicamente para asegurar la correcta identificación del dispositivo cliente por parte del servidor.

Aunque el protocolo también permite realizar una verificación de la integridad, en este proyecto sólo lo usamos para identificar de forma fiable los dispositivos.

Remote Attestation ese encarga de crear un hash resumen partiendo del soporte físico y de los programas. Este hash es diferente en función del chip TPM, del hardware y del software, por lo que el hash generado es diferente incluso en maquinas idénticas, tanto en hardware como en software.

Vemos un ejemplo cuando A quiere validar la integridad de la plataforma B

A : A crea un número no predecible de 160 bits llamado nonce.

A → B : A envía una petición de integridad con el nonce.

B : B hace un quote del (los) PCR junto con el nonce.

A ← B : B responde con el resultado del quote a A.

A : Descifra el quote, elimina la parte de nonce y comprueba que los resultados de los PCRs se corresponden con lo que tenía almacenado.

A : Queda validada o no la integridad de B.

El quote se implementa en el servidor para poder verificar que el TPM instalado en el cliente es fiable y también es quien dice ser. Ahora vamos a ir conociendo cada una de las fases por las que pasa el servidor para componer el quote.

Definimos el funcionamiento del comando TPM_QUOTE, programado en el servidor del Remote Attestation

En primer lugar construimos un elemento al que llamamos COMPOSITE, que consiste en lo siguiente:

Logitud	**PCRs**	**Logitud de los PCRs**	**Valores de los PCRs**
2 bytes	2 bytes	4 bytes	20 bytes * num. De PCRs

Explicamos el contenido de cada elemento:

- Longitud: define la longitud del siguiente elemento [PCRs], esto siempre es 2 en bytes.
- PCRs: es la parte en la que se definen cuáles de los PCRs son los que se han tomado para el attestation. La forma de fijarlos es un poco extraña y se explica en bits, cada bit a 1 representa que el PCR forma parte de la trama total. Al ser 2 bytes son 16 bits, así que vamos a detallar que bit corresponde con cada PCR:

Byte	**0**								**0**							
Bit	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PCR	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8

Por tanto si queremos activar los PCRs 0, 1, 2 y 10, nos quedaría lo siguiente:

Byte	**07**								**04**							
Bit	0	0	0	0	0	1	1	1	0	0	0	0	0	1	0	0
PCR	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8

- Longitud de los PCRs: esto es multiplicar el tamaño de cada PCR (20 bytes) por el número de PCRs. En el ejemplo que hemos hecho antes sería 20 bytes * 4 PCRs = 80, quedaría de la manera siguiente:

Byte	**00**	**50**

- Valores de los PCRs: en este punto se concatenan los valores de los PCRs de 20 bytes cada uno.

Una vez construido el COMPOSITE entero, pasamos a la construcción del quote, en primer lugar construimos el quote de 48 bytes para luego hacer el Hash que lo dejará en 20 bytes.

El quote tiene la siguiente estructura:

Versión	**QUOT**	**COMPOSITE**	**Nonce**
4 bytes	4 bytes	20 bytes	20 bytes

- Versión: siempre es 1100 que en bytes queda: 01 01 00 00 (2 bytes)
- QUOT: en estos 4 bytes se introduce QUOT en bytes.
- COMPOSITE: se concatena el COMPOSITE que ya hemos generado.
- Nonce: por último se concatena el nonce.

Por último se hace el Hash, con la función de java, de estos 48 bytes y se comprueba que el resultado obtenido es de 35 bytes. Se eliminan los primeros 15 bytes y los últimos 20 bytes deben corresponderse con el quote que ha enviado el cliente al servidor, si no se corresponden es que ha habido alguna modificación en alguno de los registros PCR seleccionados.

Nosotros usamos los PCRs del 0 al 7 ya que almacenan datos de hardware y se rellenan automáticamente al inicio del sistema. Del 8 al 15 no los usamos ya que se borran y se ponen a 0 cada vez que se reinicia el sistema.

4.2.2.4.3 Solución de autenticación de usuarios

Esta solución se basa en un Componente Ejecutable que está desarrollado en Java y usa consultas a su base de datos PostgreSQL. En este caso está implementado el Capturador de Eventos, el cual básicamente comprueba la disponibilidad de la base de datos PostgreSQL.

La Aplicación sabe envía la MAC al Componente Ejecutable, (también utilizada para la localización). Una vez que un usuario se conecta con un dispositivo, queda ligada su MAC con su nombre de usuario en la base de datos PostgreSQL. El EC consulta la base de datos y extrae el nombre del usuario y su perfil, con el cual obtiene el nivel de seguridad asignado al usuario.

Por tanto el Componente Ejecutable responde a la Aplicación con un nombre de usuario y un nivel de seguridad.

En cuanto al capturador de eventos, se basa en:

- Precondición: que la base de datos esté operativa.
- Regla de monitorización: comprueba cada período de tiempo la disponibilidad de la base de datos.

4.2.3 Clientes

El cliente tiene también una disposición semejante a la del ACS, pero menos compleja. En la siguiente figura podemos observar las partes del software desarrollado para los clientes.

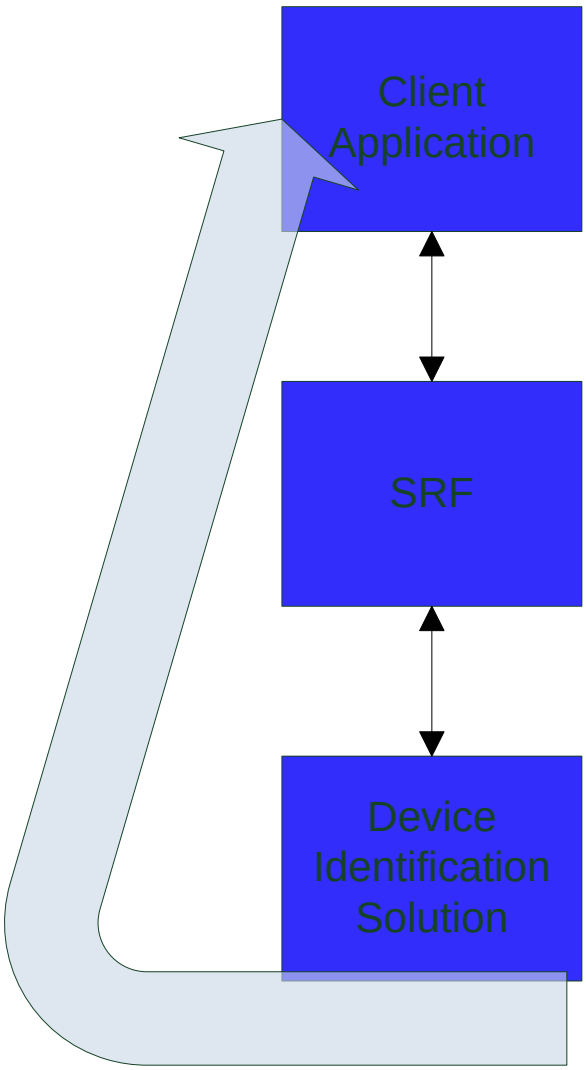


Figura 33 Software cliente

Cómo vemos en la Figura 33 el software del cliente está compuesto por tres elementos fundamentales: La Aplicación Cliente, la instancia del cliente del SRF, y la solución de identificación de dispositivos.

4.2.3.1 Client Application

Es una aplicación que identifica los interfaces de red presentes en el equipo y da al usuario la posibilidad de elegir qué interfaz está utilizando. Esta aplicación cliente envía por un socket la información de su MAC a la Aplicación de Control para que ésta busque al dispositivo en la base de datos. También tiene la función de solicitar a su instancia del SRF la solución de identificación de dispositivos. Una vez que el Componente Ejecutable *TPMClientEC* está cargado, espera a que llegue la petición de la Aplicación de Control para iniciar el Remote Attestation.

Por tanto las funciones de la Aplicación Cliente son:

- El usuario elige el interfaz por el que está accediendo a la red, y la aplicación envía la MAC de esta interfaz.
- Carga el *TPMClientEC* y una vez que reciba la petición, comienza con el protocolo de Remote Attestation.

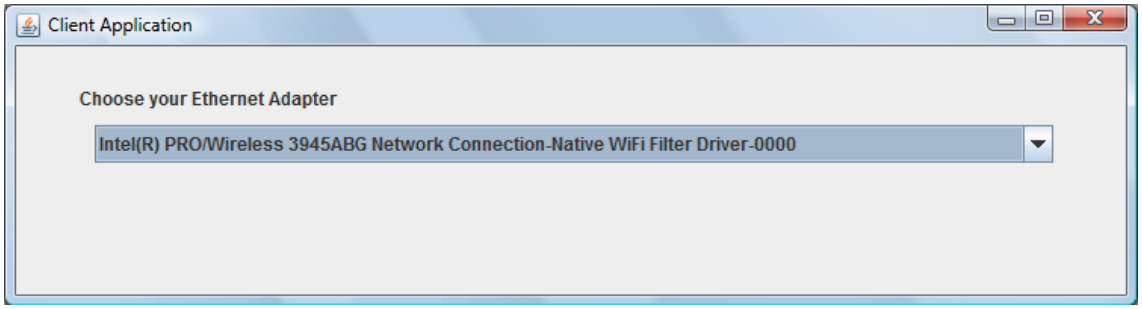


Figura 34 Client Application

En la Figura 34 podemos ver que la aplicación cliente es mucho más sencilla que la Aplicación de Control.

4.2.3.2 SRF para la aplicación cliente

Como se ha mencionado anteriormente en la memoria es el elemento que sirve de enlace con los Componentes Ejecutables, en este caso enlaza la Aplicación Cliente con el *TPMClientEC*. En el caso de la aplicación cliente no hay servicio de monitorización.

4.2.3.3 Solución de identificación de dispositivos

Descrita en el apartado 4.2.2.4.2. Se comunica con el TPM del equipo gracias a la API *tpm4java* e implementa el protocolo Remote Attestation (descrito en el apartado 4.2.2.4.2.3).

4.2.4 DHCP

El servidor DHCP es el encargado de proporcionar las direcciones ip a los equipos que se conectan a la red corporativa. Para este proyecto ha sido implementado en una máquina virtual con un Debian.

4.2.5 Firewall

También está alojado en una máquina virtual con un Debian. La configuración del firewall es una implementación de la Aplicación de Control a tiempo real, ejecutando comandos *iptables* que permiten definir reglas sobre los paquetes de la red. La Aplicación

de Control se comunica con esta máquina por SSH y ejecuta los comandos para los distintos usuarios, esto es, para las direcciones ip de los usuarios (ver apartado 4.2.2.1.3).

4.3 Sinopsis

En este capítulo hemos descrito los elementos desarrollados y los elementos integrados en el escenario del proyecto. Cabe destacar como elementos desarrollados la Aplicación de Control y los distintos Componentes Ejecutables, estos elementos son los que le dan sentido a todo el escenario. Resumimos:

- Aplicación de Control: se encarga de otorgar permisos para acceder a determinados recursos mediante la recopilación de información en base a tres factores de seguridad: localización, identificación de dispositivo y autenticación de usuario. Una vez establecidos estos valores de seguridad se comunica con el Firewall para proporcionar los recursos correspondientes.
- Componentes Ejecutables: podemos observar la siguiente tabla que los resume.

	Localización	**Identificación**	**Autenticación**
Función	**Valor de seguridad en base a la localización del dispositivo.**	**Valor de seguridad en base a la identificación del dispositivo.**	**Valor de seguridad en base a la autenticación del usuario.**
Precondición	**Comprobar BBDD.**		**Comprobar BBDD.**
Regla	**Calidad mayor que un umbral.**		**Comprobar BBDD.**

Tabla 10 Resumen de Soluciones S&D

En cuanto al resto de elementos utilizados destacamos la integración del SRF y del servicio de monitorización, ambos elementos ya desarrollados en el marco del proyecto europeo SERENITY.

5 Integración, pruebas y resultados

5.1 Integración

La integración de todos los elementos que hemos ido describiendo anteriormente ha sido muy dispar, dependiendo de las características concretas de cada uno de ellos.

Cabe destacar el trabajo y tiempo dedicados a implementar todos los elementos del ACS (ver sección 4.2.2) e integrarlos. Conforme se desarrollaba el proyecto cambiamos algunas especificaciones con vistas a la integración, para adaptarlas a las necesidades de ejecución del resto de elementos. En la siguiente lista podemos ver una serie de modificaciones en cuanto a integración que hemos tenido que adoptar para que el escenario no se vea perjudicado en su conjunto:

En cuanto a la negociación entre los Componentes Ejecutables de TPM (ver sección 4.2.2.4.2) presentes en el ACS y el cliente, tuvimos que adaptarla para utilizar una negociación implementada en los propios ECs, ya que el SRF (ver sección 4.2.2.2) no realizaba correctamente esta parte. Para solucionarlo hubo que implementar un paso previo en el código de los ECs, por el cual el EC cliente esperaba a recibir la petición del servidor, y una vez que la recibía, se podía proceder con el Remote Attestation (ver sección 4.2.2.4.2.3).

En cuanto al servicio de Monitorización (ver sección 4.2.2.3) fue desarrollado en el contexto del proyecto europeo SERENITY y la integración correspondía al SRF y a los Componentes Ejecutables. Después de algunos problemas debido a falta de comunicación con los creadores de este módulo (un departamento de una universidad de Londres). Después de estos pequeños problemas la integración con el proyecto se consiguió sin problemas.

Una vez que tuvimos desarrollados por separado la Aplicación de Control (ver sección 4.2.2.1) y los Componentes Ejecutables (ver sección 4.2.2.4), la integración corrió a cargo del SRF, procediendo a agregar las Soluciones S&D implementadas de la manera descrita en la sección 4.2.2.2.

Con respecto a la Aplicación del Cliente (ver sección 4.2.3.1), no tuvimos problemas con la integración debido su simplicidad comparada con el ACS. Sólo tenía un Componente Ejecutable que se carga de la manera mencionada en la sección 4.2.2.2. y no tiene servicio de monitorización. Por tanto esta parte del proyecto fue sencilla de integrar.

La integración del Firewall con la red cableada, la red Wi-Fi y con la Aplicación de Control no fue especialmente compleja. Tuvimos que tener en cuenta todas las direcciones de red, sin dejar ninguna comunicación abierta entre la red Wi-Fi y la red corporativa. Aun

así la integración fue satisfactoria, consiguiendo que los usuarios de la red Wi-Fi estuvieran perfectamente aislados de la red cableada.

Una vez integrados todos los elementos que componen el proyecto procedimos a hacer pruebas para integrarlo en diferentes escenarios, como detallamos en la siguiente sección de pruebas y resultados (sección 5.2). El escenario de Telefónica I+D es un escenario genérico de una oficina o empresa pequeña donde la evolución de la seguridad en tiempo real es muy importante. En el segundo caso es el escenario de la Escuela Politécnica Superior, el Laboratorio de Ambientes Inteligentes, que representa un entorno domótico donde los elementos del hogar se controlan a través de un servidor.

En este segundo caso cabe destacar la versatilidad y seguridad aportada al escenario existente, ya que el Laboratorio AmI no contaba con control de seguridad a tiempo real dependiente de los tres factores expuestos a lo largo de la memoria:

- Localización del dispositivo.
- Autenticación de usuario.
- Identificación de dispositivo.

Por tanto con estos dos escenarios estamos dando al proyecto una proyección muy importante dentro de las tecnologías de los escenarios AmI.

5.2 Pruebas y resultados

En cuanto a las pruebas realizadas en los diferentes entornos tenemos que decir que han sido muy satisfactorias, pudiéndose representar todos los casos propuestos con una funcionalidad muy elevada.

Hemos probado el sistema dentro de las oficinas de Telefónica I+D, uno de los integrantes del proyecto SERENITY. A su vez, hemos realizado también la integración del mismo con el laboratorio AmI presente en la Escuela Politécnica Superior de la Universidad Autónoma de Madrid, para dar más versatilidad al proyecto y demostrar que se puede migrar a distintos escenario de una manera sencilla.

En primer lugar representamos de forma genérica el escenario que hemos elaborado. Como se puede observar la Figura 35 representa un escenario que puede ser aplicable a distintos ámbitos, con esto queremos hacer hincapié en la portabilidad y adaptabilidad del entorno que hemos generado.

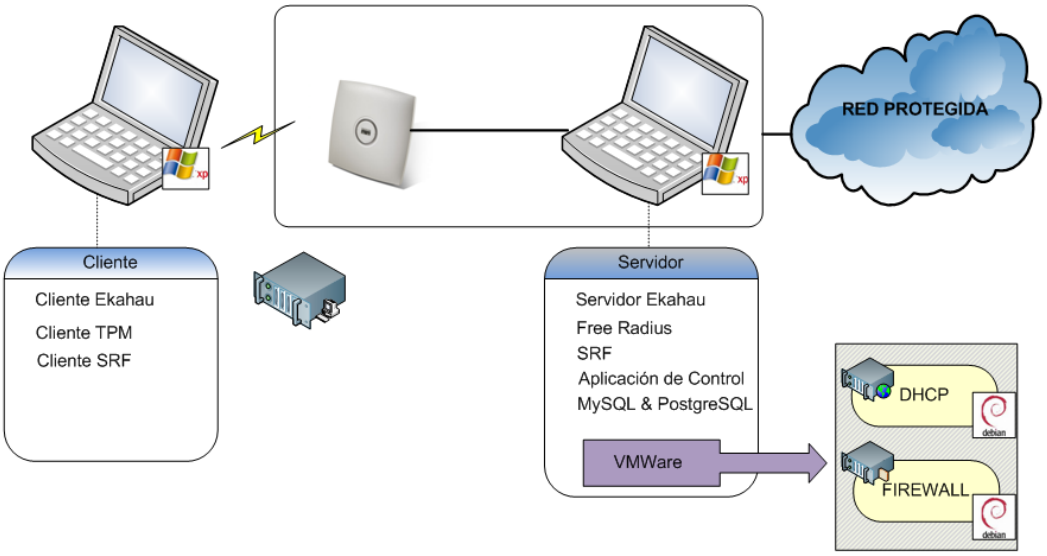


Figura 35 Modelo del escenario genérico

Ahora pasamos a detallar los aspectos en cada uno de los entornos que hemos probado este proyecto, en Telefónica I+D y en el laboratorio AmI de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid.

5.2.1 Escenario en Telefónica I+D

Sistema de Control de Acceso para la red Wi-Fi de una oficina corporativa. El Sistema se encarga de determinar la autorización de todas las peticiones de recursos hechas por los usuarios conectados a la red inalámbrica, siendo un recurso cualquier elemento accesible con una dirección IP y un puerto, en nuestro caso hemos definido dos recursos: Internet y la impresora del despacho.

El Sistema utiliza SERENITY como gestor de las soluciones de seguridad. El control de acceso se basa en tres factores:

- Identidad del usuario y su perfil asociado (empleado, becario, visitante, etc.)
- Identidad del dispositivo que utiliza
- Localización actual (del dispositivo)

Se han modelado tres Patrones SERENITY que evalúan estos tres factores y proporcionan un nivel de seguridad. La política de acceso a cada recurso se define estableciendo el valor “de corte” para poder acceder a él, es decir, el nivel mínimo que tiene que tener cada uno de estos tres elementos para que un usuario pueda acceder a él.

A continuación describimos todo lo necesario para montar esta demostración, poniendo así de manifiesto la versatilidad del sistema. Partimos de la base que tenemos un servidor y un cliente tal y como se menciona en el ANEXO A.

- El primer paso es calibrar el plano de la zona donde va a actuar la Aplicación de Control. Esta tarea consiste en recorrer las zonas definidas en el servicio de localización. Como podemos ver en la Figura 36 hay tres zonas en el escenario. Una vez que tengamos calibradas las zonas, tenemos que cargar este modelo en el Ekahau Positioning Engine, que está presente en el Servidor de Control de Acceso.
- Una vez cargado el modelo hay que actualizar la base de datos del Componente Ejecutable de localización con los nombres de las nuevas áreas y aplicarles su nivel de seguridad, como podemos ver en la Tabla 11.

	Medida de Localización
Own Office Room	**Alta**
Coffe Room	**Baja**
Meeting Room	**Media**

Tabla 11 Medidas de localización en el escenario TID

- Llegado a este punto configuramos el Firewall con los datos de la red cableada donde están los recursos, esto se configura en el archivo de control de la Aplicación (ver el ANEXO B).
- En la base de datos de Autenticación de Usuarios y en la de Identificación de Dispositivos hay que dar de alta a un usuario y un nuevo dispositivo, con sus respectivos niveles de seguridad. Se puede ver en la Tabla 12.

	Medida de Usuario	**Medida de Dispositivo**
Empleado A	**Alta**	**Alta**

Tabla 12 Medidas de seguridad del empleado A

En estos momentos ejecutamos la Aplicación de Control en el servidor para proceder a agregar los recursos. El proceso de añadir recursos se hace con la Aplicación de Control corriendo y se actualizan de forma dinámica. Los Recursos que añadimos se pueden ver en la Tabla 13.

	Medida de Usuario	Medida de Dispositivo	Medida de Localización
Internet	**Baja**	**Media**	**Media**
Impresora	**Alta**	**Alta**	**Alta**

Tabla 13 Recursos en el escenario TID

Con esto se tiene al servidor y al cliente listos para probar todo el proyecto con la modificación dinámica de los permisos del firewall. Ahora el servidor ya tiene corriendo la Aplicación de Control y el cliente debe conectarse a la red inalámbrica, para poder proceder con la verificación de resultados.

Como hemos visto resulta sencillo instalar el sistema en un nuevo entorno. Queremos mencionar el trabajo llevado a cabo para crear un sistema altamente portable y compatible con todo múltiples entornos. Una vez llegado a este punto los clientes de la red tendrán una seguridad dinámica en función de los tres factores de seguridad del sistema:

- Autenticación del usuario.
- Localización del dispositivo.
- Identificación del dispositivo.

5.2.1.1 Resultados en el escenario de Telefónica I+D

Ahora pasamos a detallar los resultados de las pruebas en el escenario de Telefónica I+D.

El escenario en el que hemos montado las pruebas consiste en diversos despachos y una sala de reuniones. Como podemos ver en la Figura 36.

Figura 36 Escenario Telefónica I+D

Los distintos niveles de seguridad de las zonas se pueden ver en la Tabla 14.

	Medida de Localización
Own Office Room	**Alta**
Coffe Room	**Baja**
Meeting Room	**Media**

Tabla 14 Medidas de localización en el escenario TID

La Tabla 15 representa los requisitos de seguridad para los recursos.

	Medida de Usuario	Medida de Dispositivo	Medida de Localización
Internet	**Baja**	**Media**	**Media**
Impresora	**Alta**	**Alta**	**Alta**

Tabla 15 Recursos en el escenario TID

Vemos cómo la impresora sólo debería poderse acceder por el propietario del despacho, por tanto los requisitos de seguridad son altos. En cambio el recurso Internet no tiene grandes requisitos ya que debería poderse acceder desde diversas salas y por los visitantes que participen en reuniones.

5.2.1.1.1 Situación I

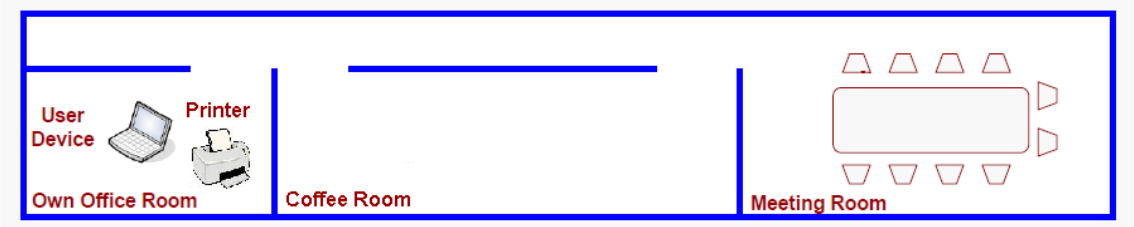


Figura 37 Escenario TID: Situación I

En el caso de la Figura 37 vemos cómo el empleado A está situado en su despacho. Suponemos que el empleado A tiene las siguientes características, ver Tabla 16.

	Medida de Usuario	Medida de Dispositivo
Empleado A	**Alta**	**Alta**

Tabla 16 Medidas de seguridad del empleado A

El empleado A se conecta a la red Wi-Fi corporativa e inmediatamente se autentica con sus credenciales corporativas (User&Password). Ahora intenta acceder a internet y el Servidor de Control de Acceso evalúa la situación:

- Chequea la identidad del dispositivo
- Chequea la localización del dispositivo

Tomado los dos resultados anteriores junto con el dato previo de la identidad, consulta las políticas de acceso y de acuerdo con ellas le permite el acceso a Internet. Ahora también se dispone a imprimir unos documentos que ha visto en Internet y el Servidor de Control de Acceso también le permite acceder al recurso Impresora.

Vemos como la tabla de medidas de seguridad del empleado A queda de la siguiente manera cuando está situado en su despacho:

	Medida de Usuario	Medida de Dispositivo	Medida de Localización
Empleado A	**Alta**	**Alta**	**Alta**

Tabla 17 Medidas de seguridad del empleado A. Situación I

Por tanto cubre las necesidades de seguridad para los dos recursos representados en la Tabla 15.

	Recursos Asignados
Empleado A	**Internet**
	Impresora

Tabla 18 Situación I: Recursos asignados

5.2.1.1.2 Situación II

En este caso podemos ver cómo el empleado ahora se ha movido a otra localización.

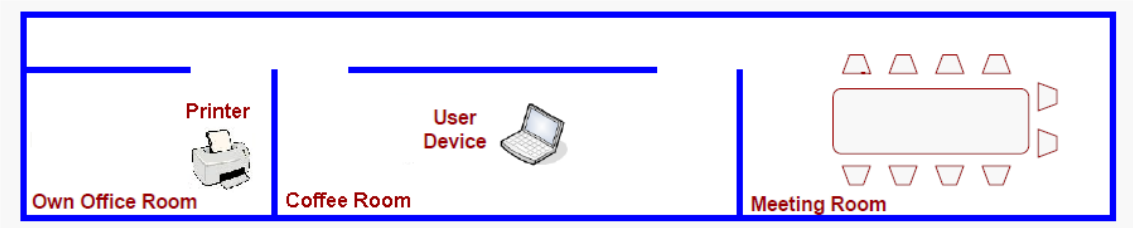


Figura 38 Escenario TID: Situación II

Vemos cómo el empleado se ha movido a la sala de cafés sin desconectar su equipo. En este caso su tabla de seguridad queda de la siguiente manera:

	Medida de Usuario	Medida de Dispositivo	Medida de Localización
Empleado A	**Alta**	**Alta**	**Baja**

Tabla 19 Medidas de seguridad del empleado A. Situación II

En este caso la Aplicación de Control vuelve a comprobar la capacidad del empleado de acceder a los recursos que tenía asignados. Ahora la seguridad en la localización está en nivel bajo y por tanto no cumple los requisitos mínimos para acceder a ningún recurso, como se puede observar en la Tabla 15.

Por tanto en esta situación se puede ver cómo la Aplicación de Control deniega el acceso a los recursos.

En la Tabla 20 observamos que la Aplicación de Control no asigna ningún recurso al empleado en esta situación.

	Recursos Asignados
Empleado A	**Ninguno**

Tabla 20 Situación II: Recursos asignados

5.2.1.1.3 Situación III

En este caso vemos cómo el empleado tiene una reunión y se ha desplazado a la sala de reuniones.

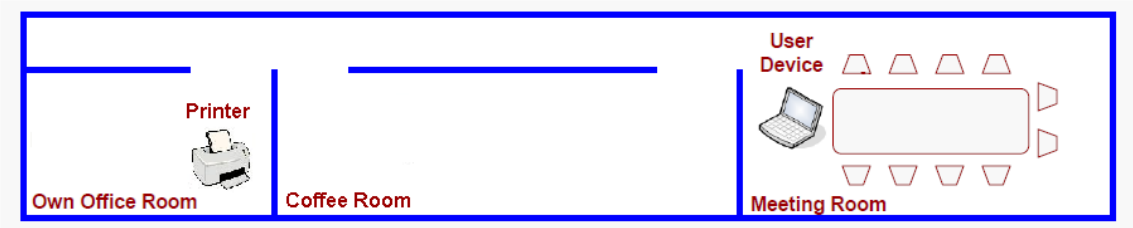


Figura 39 Escenario TID: Situación III

En este último caso se produce un proceso similar al anterior. El sistema evoluciona a tiempo real y reconsidera los recursos que le tiene asignados al usuario, en la situación anterior no tenía ninguno, y valora si tiene los requisitos de seguridad suficientes como para asignarle algún recurso que no tuviera.

La tabla de seguridad del empleado A en este caso queda de la siguiente manera:

	Medida de Usuario	Medida de Dispositivo	Medida de Localización
Empleado A	**Alta**	**Alta**	**Media**

Tabla 21 Medidas de seguridad del empleado A. Situación III

Entonces visualizando la Tabla 21 y la Tabla 15 la Aplicación de Control asigna el recurso de Internet al empleado A.

En la Tabla 22 observamos que la Aplicación de Control asigna el recurso de Internet.

	Recursos Asignados
Empleado A	**Internet**

Tabla 22 Situación III: Recursos asignados

5.2.2 Laboratorio AmI en la Escuela Politécnica Superior

La Figura 40 muestra el laboratorio AmI de la Escuela Politécnica Superior. Observamos que la habitación de la derecha es el Salón y la de la izquierda es el Office. Al integrar este proyecto con el laboratorio AmI le estamos aportando al laboratorio una serie de funcionalidades que no poseía, como es la localización de los usuarios en el interior del mismo, pudiendo discriminar entre dos zonas de seguridad. También estamos aportando seguridad en cuanto al dispositivo y al usuario.

Uno de los aspectos importantes es el de proporcionar la posibilidad de acceso a los elementos del entorno a diferentes usuarios según los criterios de seguridad establecidos. Considerando la localización y el resto de factores de seguridad, los usuarios sólo tendrán acceso a los elementos del entorno si cumplen los requisitos de seguridad para acceder a los mismos.

Figura 40 Laboratorio de Ambientes Inteligentes EPS

Para poder llevar a cabo este proceso ha sido necesario configurar una serie de elementos. Empezamos calibrando el laboratorio para que el servidor de localización pudiera localizar a los usuarios. El Laboratorio AmI posee dos estancias, Salón y Office,

que son las que calibramos. De esta manera podemos diferenciar ambas salas en sus niveles de seguridad para que tengan distintos permisos de acceso los usuarios que estén situados en salas distintas.

Como se puede ver en la Figura 41 las líneas verdes muestran las zonas por las que hemos caminado y guardado para que luego el servidor sea capaz de establecer una relación entre estas medidas guardadas y la posición real del dispositivo. A su vez, también hay que instalar un par de routers para simular la red corporativa a la que se conectan los usuarios.

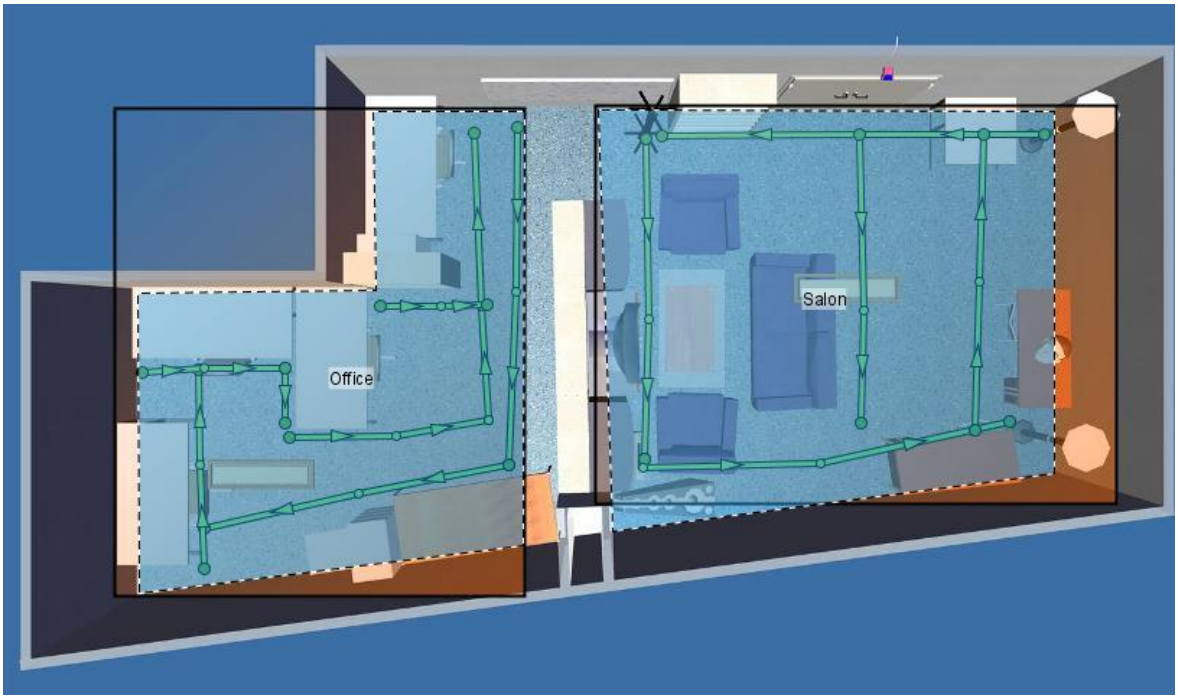


Figura 41 Calibración de la localización en el AmILab en la EPS

A continuación describimos todo lo necesario para montar esta demostración, poniendo así de manifiesto la versatilidad del sistema. Partimos de la base que tenemos un servidor y un cliente tal y como se menciona en el ANEXO A. En este caso queremos destacar también que se trata de un escenario domótico y por tanto un nuevo tipo de escenario que pone de manifiesto la portabilidad del sistema.

- El primer paso es calibrar el plano de la zona donde va a actuar la Aplicación de Control. Esta tarea consiste en recorrer las zonas definidas en el servicio de localización, Figura 41. Como podemos ver en la Figura 41, hay dos zonas en el escenario, Salón y Office. Una vez que tengamos calibradas las zonas, tenemos que cargar este modelo en el Ekahau Positioning Engine, que está presente en el Servidor de Control de Acceso.

- Una vez cargado el modelo hay que actualizar la base de datos del Componente Ejecutable de localización con los nombres de las nuevas áreas y aplicarles su nivel de seguridad, como podemos ver en la Tabla 23.

	Medida de Localización
Salon	**Alta**
Office	**Media**

Tabla 23 Medidas de localización en el AmILab

- Llegado a este punto configuramos el Firewall con los datos de la red cableada donde están los recursos, esto se configura en el archivo de control de la Aplicación (ver el ANEXO B).
- En la base de datos de Autenticación de Usuarios y en la de Identificación de Dispositivos hay que dar de alta a dos usuarios que acceden desde el mismo dispositivo, con sus respectivos niveles de seguridad. Se puede ver en la Tabla 24.

	Medida de Usuario
Daniel	**Alta**
Ariel	**Media**

Tabla 24 Medidas de Usuario

En estos momentos ejecutamos la Aplicación de Control en el servidor para proceder a agregar los recursos. El proceso de añadir recursos se hace con la Aplicación de Control corriendo y se actualizan de forma dinámica. Los Recursos que añadimos se pueden ver en la Tabla 25.

	Medida de Usuario	**Medida de Dispositivo**	**Medida de Localización**
Iluminación	**Alta**	**Media**	**Alta**

Tabla 25 Recursos en el AmILab

Con esto se tiene al servidor y al cliente listos para probar todo el proyecto con la modificación dinámica de los permisos del firewall. Ahora el servidor ya tiene corriendo la Aplicación de Control y el cliente debe conectarse a la red inalámbrica, para poder proceder con la verificación de resultados.

Como hemos visto lo sencillo de correr una demostración, queremos mencionar el gran trabajo llevado a cabo para crear un sistema altamente portable y totalmente compatible con todo tipo de entornos. Una vez llegado a este punto los clientes de la red tendrán una seguridad dinámica en función de los tres factores de seguridad del sistema:

- Autenticación del usuario.
- Localización del dispositivo.
- Identificación del dispositivo.

En esta segunda demostración en el escenario del Laboratorio AmI de la EPS podremos ver la versatilidad que le aporta al laboratorio, proporcionándole nuevas herramientas de control de acceso a los recursos basadas en elementos que pueden modificarse en tiempo real. En nuestro caso el recurso que hemos planteado es el de iluminación, pero una vez comprobado que funciona con este recurso es muy sencillo extender esta demostración al manejo de varios recursos por varios usuarios de manera dinámica.

5.2.2.1 Resultados en el escenario de AmILab en la EPS

Ahora pasamos a detallar los resultados de las pruebas en el AmILab de la EPS, Figura 42.

Figura 42 AmILab en la EPS

En la Tabla 26 se pueden observar los distintos niveles de seguridad de las zonas del laboratorio.

	Medida de Localización
Salon	**Alta**
Office	**Media**

Tabla 26 Medidas de localización en el AmILab

La Tabla 27 representa los requisitos de seguridad para el recurso.

	Medida de Usuario	**Medida de Dispositivo**	**Medida de Localización**
Iluminación	**Alta**	**Media**	**Alta**

Tabla 27 Recursos en el AmILab

Vemos cómo el recurso Iluminación solo puede accederse a él si estamos situados en el Salón, nuestro usuario tiene permisos altos y el dispositivo desde el que está accediendo tiene una seguridad media.

La Tabla 28 muestra la seguridad de los dos usuarios que hemos utilizado para las pruebas en el AmILab.

	Medida de Usuario
Daniel	**Alta**
Ariel	**Media**

Tabla 28 Medidas de Usuario

Vemos como el usuario Daniel tiene más seguridad que Ariel, por tanto Daniel es el único que cumple con el requisito de seguridad de usuario del recurso de Iluminación, como podemos ver en la Tabla 27.

Ahora pasamos a ver cada una de las situaciones que hemos probado en el AmILab.

5.2.2.1.1 Situación I

Esta primera situación es cuando accede el usuario Daniel y se encuentra en el salón. Como se puede ver en la Figura 43.

Figura 43 Escenario AmILab: Situación I

Daniel se conecta a la red Wi-Fi e inmediatamente se autentica con sus credenciales(User&Password). Ahora intenta acceder al recurso de iluminación y la Aplicación de Control evalúa la situación:

- Chequea la identidad del dispositivo
- Chequea la localización del dispositivo

Tomado los dos resultados anteriores junto con el dato previo de la identidad, consulta las políticas de acceso y de acuerdo con ellas le permite el acceso al recurso y enciende la luz.

	Medida de Usuario	**Medida de Dispositivo**	**Medida de Localización**
Daniel	**Alta**	**Media**	**Alta**
Iluminación	**Alta**	**Media**	**Alta**

Tabla 29 Medidas de seguridad de Daniel. Situación I

Por tanto cubre las necesidades de seguridad para tener acceso al recurso y por tanto puede encender y apagar la luz mientras se encuentre en el Salón, como podemos ver en la Tabla 29.

5.2.2.1.2 Situación II

En esta situación Daniel cambia de localización y ahora se encuentra en el Office. Como se puede ver en la Figura 44.

Figura 44 Escenario AmILab: Situación II

Vemos como la tabla de medidas de seguridad de Daniel queda de la siguiente manera cuando está situado en el Office:

	Medida de Usuario	Medida de Dispositivo	Medida de Localización
Daniel	**Alta**	**Media**	**Media**
Iluminación	**Alta**	**Media**	**Alta**

Tabla 30 Medidas de seguridad de Daniel. Situación II

En este caso Daniel pierde el acceso al recurso, no cumple el requisito de localización como podemos ver en la Tabla 30 y no puede encender y apagar la luz desde su portátil cuando está situado en el Office.

5.2.2.1.3 Situación III

En esta situación viene Ariel, un amigo de Daniel, y accede a la red desde el Office, como podemos ver en la Figura 45.

Figura 45 Escenario AmILab: Situación III

Ariel se conecta a la red Wi-Fi corporativa e inmediatamente se autentica con sus credenciales corporativas (User&Password). Ahora intenta acceder al recurso de iluminación y la Aplicación de Control evalúa la situación:

- Chequea la identidad del dispositivo
- Chequea la localización del dispositivo

Tomado los dos resultados anteriores junto con el dato previo de la identidad, consulta las políticas de acceso y de acuerdo con ellas no le permite el acceso al recurso debido a que no se cumplen los requisitos de Localización y de Usuario.

Vemos como la tabla de medidas de seguridad de Ariel queda de la siguiente manera cuando está situado en el Office:

	Medida de Usuario	Medida de Dispositivo	Medida de Localización
Ariel	**Media**	**Media**	**Media**
Iluminación	**Alta**	**Media**	**Alta**

Tabla 31 Medidas de seguridad de Ariel. Situación III

En este caso Ariel no tiene acceso al recurso ya que sólo cumple el requisito de Dispositivo, el de Usuario y el de Localización no los cumple comparando los valores de Ariel con los necesarios para acceder al recurso, como podemos ver en la Tabla 31.

5.2.2.1.4 Situación IV

En esta situación Ariel se traslada al Salón, como podemos ver en la Figura 46.

Figura 46 Escenario AmILab: Situación IV

En este caso Ariel tampoco tiene acceso al recurso, de hecho Ariel nunca será capaz de acceder al recurso debido al nivel de seguridad que tiene asignado, Medio, que no es mayor o igual que el necesario para acceder al recurso, Alto. Como podemos ver en la Tabla 32.

	Medida de Usuario	**Medida de Dispositivo**	**Medida de Localización**
Daniel	**Media**	**Media**	**Alta**
Iluminación	**Alta**	**Media**	**Alta**

Tabla 32 Medidas de seguridad de Ariel. Situación IV

6 Conclusiones y trabajo futuro

6.1 Conclusiones y trabajo futuro

El concepto de AmI representa un nuevo y prometedor paradigma de computación que tiene el potencial de representar avances muy importantes tanto en el trabajo como en la vida cotidiana de las personas. Sin embargo, la combinación de heterogeneidad, movilidad, dinamismo y número ingente de dispositivos, junto con la relevancia que los aspectos de seguridad y confiabilidad tienen en estos escenarios, ponen de manifiesto que la seguridad debe ser una prioridad y un requisito indispensable si queremos que los escenarios AmI puedan salir de los laboratorios.

En este trabajo hemos puesto de manifiesto la necesidad de llevar a cabo la supervisión y adaptación dinámica de los sistemas de seguridad ante los cambios que se producen en los ecosistemas AmI.

Este proyecto propone una estrategia realista, basada en el proyecto SERENITY, para la consecución de la seguridad en este tipo de entornos de computación. Esta estrategia se sustenta en un concepto global de seguridad que incluye tanto aspectos estáticos como dinámicos. Asimismo se ha descrito el marco de trabajo que posibilita su aplicación en escenarios reales de comunicación inalámbrica.

Por tanto queremos dejar patente con este proyecto que el desarrollo de nuevos elementos para dotar de seguridad los entornos AmI es fundamental. Podemos partir de este proyecto como base para desarrollar un sistema de seguridad necesario en un entorno AmI.

El desarrollo del proyecto ha perseguido en todo momento dotar de seguridad las comunicaciones inalámbricas en un escenario AmI con una red Wi-Fi y con sus recursos en una red cableada. Con este propósito se ha establecido que la seguridad dependa de tres factores importantes como son la localización, la autenticación del usuario y la identificación del dispositivo a través del cual está accediendo a la red Wi-Fi. Debido a estos tres factores la seguridad en el acceso a los recursos de la red cableada es muy elevada.

Una vez que se ponen en conjunto estos tres factores y se actualizan en tiempo real, la seguridad de los elementos que componen la red cableada donde se encuentran los recursos aumenta de una manera muy considerable. Por tanto este proyecto proporciona seguridad dinámica en Ambientes Inteligentes en base a tres factores de seguridad (localización, la autenticación del usuario y la identificación del dispositivo).

En la sección de pruebas y resultados (apartado 5.2) podemos ver como es un proyecto versátil y transportable para ser instalado en entornos en los que se quiera

permitir acceso de manera fiable a los recursos en una red Wi Fi. Más concretamente cuando describimos la integración efectuada en el Laboratorio AmI de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid, ver sección 0, podemos denotar características de portabilidad y adaptabilidad, así como de funcionalidad.

Hemos integrado este proyecto en el Laboratorio de la EPS proporcionándole ciertas funciones que no poseía antes, tales como acceso controlado a los recursos del laboratorio en función de la localización, identificación del dispositivo y la autenticación del usuario. La funcionalidad de localización y la posibilidad de que sólo ciertos recursos sean accesibles en determinadas áreas tiene una amplia progresión futura, ya que abre un amplio abanico de nuevas posibilidades a los Ambientes Inteligentes dentro de la EPS.

Las oportunidades que representa esta integración son numerosas y mucho más amplias que si analizamos cada uno de estos elementos por separado. La combinación de la Aplicación de Control de Seguridad de SERENITY con el AmILab de la EPS ofrece posibilidades domóticas basadas en el desarrollo de componentes de localización permanente de los miembros del hogar para proporcionarles servicios instantáneos de control de los elementos donde están presentes. Así si se encuentra en el salón, y tiene permiso para ello, tendrá la posibilidad de encender el televisor o de subir y bajar las persianas, pero si cambia de localización o no posee permisos suficientes perderá estos recursos.

En un futuro este tipo de escenarios, con seguridad en diferentes factores que se actualizan en tiempo real, responderá a las mayores necesidades de la población. Un funcionamiento global de diversos factores de seguridad en tiempo real dotará a todas las comunicaciones de una mayor seguridad así como permitirá a los usuarios estar permanentemente conectados a los recursos que por cercanía, identidad o autenticación estén disponibles.

Otro aspecto en el que podemos establecer las líneas de investigación futuras consiste en ofrecer soluciones de seguridad modulares e intercambiables que puedan ir evolucionando en tiempo real. Estas soluciones, si se combinan, dan lugar a escenarios donde varios elementos de seguridad son tenidos en cuenta a la vez y a tiempo real, y por tanto desemboca en un incremento notable de la versatilidad del escenario, así como de su seguridad.

Concretando proponemos como parte del trabajo futuro un desarrollo mencionado en la sección 4.1.1.2, que nos indica la posibilidad de desarrollar un sistema de localización basado en otros parámetros, que complemente al ya implementado basado en Wi-Fi. Con esta mejoría el sistema podría elegir que opción de localización que es más interesante en cada momento pudiendo, cambiar de una a otra en tiempo real.

7 Referencias

[1] SERENITY project. Funded by European Commission. Directorate General Information Society & Media.Unit D4 - ICT for Trust and Security, under grant IST-027587. 2006.

[2] Spanoudakis, G., Gomez, A. M., Kokolakis, S. Security and Dependability for Ambient Intelligence. Springer US 2009. ISSN: 1568-2633.

[3] Gallego-Nicasio Crespo, B., Piñuela, A. The SERENITY Runtime Framework chapter 11 of Security and Dependability for Ambient Intelligence. Springer US 2009. ISSN: 1568-2633.

[4] Hellenschmidt, M. Kirste1, T. A Generic Topology for Ambient Intelligence. pp 112-123 from Ambient Intelligence. Springer Berlin / Heidelberg 2004.

[5] http://www.Intel.com/technology/security

[6] Brooks, P., Kalaher, P., Ni Riain, T. Measuring the Value of Intel® Core™2 Processor with vPro™ Technology in the Enterprise. Intel Corporation 2006.

[7] Trusted Computing Group http://www.trustedcomputinggroup.org

[8] Chris J. Mitchell. Research Workshop on Future TPM Functionality. 2006.

[9] William Stallings. Cryptography and network security. Prectice Hall 2006. Chapter 9.2

[10] William Stallings. Cryptography and network security. Prectice Hall 2006. Chapter 12.1

[11] William Stallings. Cryptography and network security. Prectice Hall 2006. Chapter 12.3

[12] SafeXcel-2141. High-Performance Security-System-On-A-Chip. Safenet 2005.

[13] Schneier, B. Applied cryptography: protocols, algorithms, and source code in C. Wiley-India 2007. Chapter 24.8.

[14] Muñoz, A., Maña, A. A Hardware Based Infrastructure for Agent Protection 3rd Symposium of Ubiquitous Computing and Ambient Intelligence 2008. pp 39-47.

[15] TPM/J Java-based API for the Trusted Platform Module, 2007.

[16] jTSS Java-based API for the Trusted Platform Module, 2008

[17] Ekahau Positioning Engine 2.1, EPE

[18] Porcino, D.; Hirt, W. Ultra-wideband radio technology: potential and challenges ahead. Philips Research 2003. pp 66-74.

[19] http://www.newburynetworks.com/

[20] http://www.skyhookwireless.com/

[21] http://www.celtic-initiative.org/Projects/BUGYO/default.asp

[22] https://www.cenitsegura.com/cenit/

[23] http://www.itea-econfidential.org/?q=node/6

[24] http://www.mvia.es/

[25] http://www.cvisproject.org/

[26] Gier, J. Deploying WPA and WPA2 in the Enterprise. Wi-Fi Alliance 2005.

[27] http://www.wi-fi.org

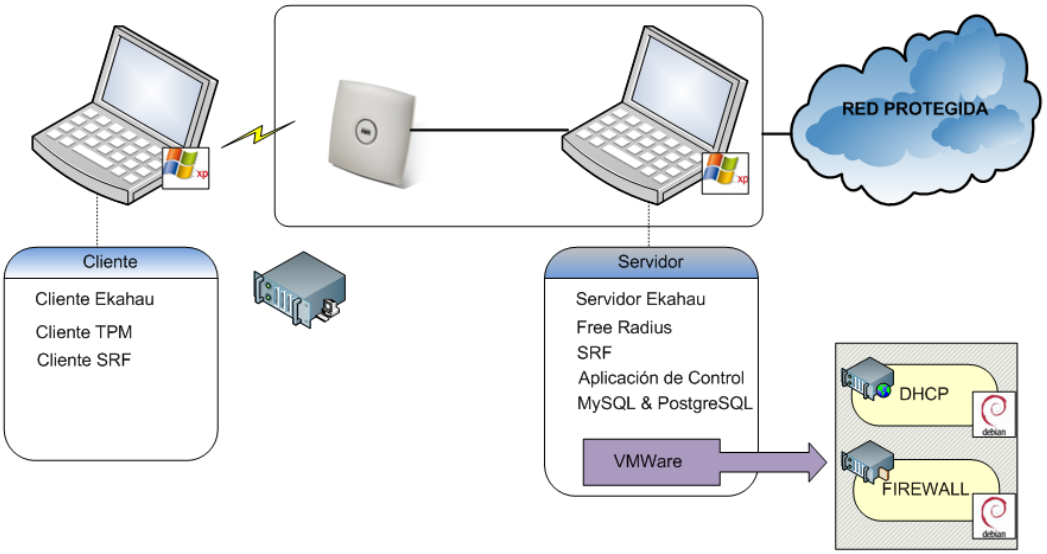
[28] Rigney C., Rubens A. C., Simpson W. A. and Willems S. Remote Authentication Dial In User Service (RADIUS). Internet Engineering Task Force (IETF) 2000. RFC 2865

[29] El objeto *LocPrecisa* lo hemos implementado expresamente para que el envío de información entre el Componente Ejecutable y la Aplicación de Control sea más sencillo. El objeto contiene las coordenadas X e Y del dispositivo localizado y el mapa de la zona. La imagen del mapa está guardada como cadena de bytes para que el objeto sea serializable.

[30] Muñoz, A., Maña, A. A Hardware Based Infrastructure for Agent Protection 3rd Symposium of Ubiquitous Computing and Ambient Intelligence 2008. pp 52-65.

ANEXO A Instalación del escenario

1 Descripción del prototipo



2 Instalación de la Red

La red de acceso estará en el rango ip 192.170.0.0/24 y consta de cuatro puntos de acceso CISCO Aironet 1130AG, de ellos actúa de punto de acceso principal y el resto de repetidores.

Consta también de un servidor DHCP para la asignación de ips a los clientes y un Firewall que se situará entre nuestra red de acceso y la red que deseamos proteger, tendrá que tener entonces dos interfaces de red.

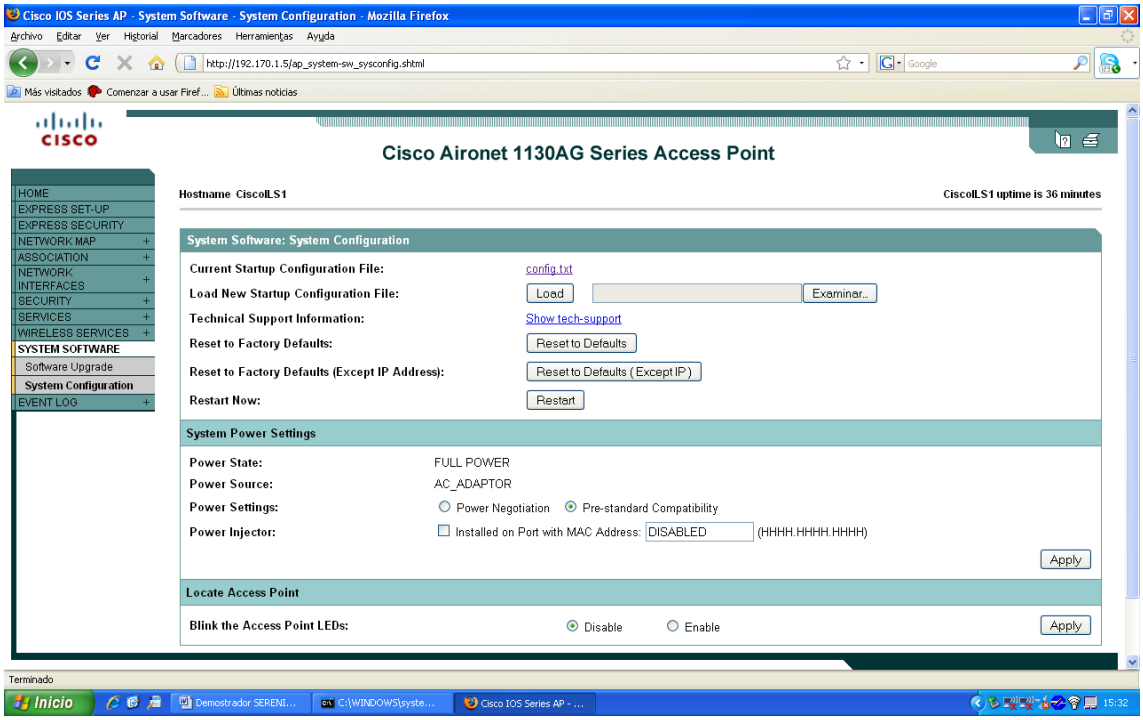
2.1 Configuración de los puntos de acceso

Asignar las direcciones IP de cada punto de acceso.

Punto de Acceso	**IP**
CiscoILS1 (Maestro)	**192.170.1.5**
CiscoILS2	**192.170.1.6**
CiscoILS3	**192.170.1.7**
CiscoILS4	**192.170.1.8**

Los puntos de acceso se configuran cargando los ficheros de configuración preparados para tal efecto.

Para cargar un fichero de configuración, ir a SYSTEM SOFTWARE → System Configuration. El fichero de configuración es un fichero de texto plano con todos los datos que se generan al utilizar el interfaz gráfico de administración.



2.2 Configuración del servidor DHCP

Para esta labor hemos elegido una maquina liux debían que la tenemos virtualizada. (Valdría cualquier servidor DHCP conectado a la red de acceso)

Para instalar el servidor DHCP en nuestra maquina basta con ejecutar el comando: # apt-get install dhcpd

Para configurarlo editamos el fichero dhcpd.conf, primero paramos el servicio con el comando # /etc/init.d/dhcp stop y lo editamos con el comando: # nano /etc/dhcpd.conf

El contenido será el siguiente:

```
option domain-name "serenity"; #nombre de dominio
option domain-name-servers X.X.X.X X.X.X.X; #servidores DNS de la red
option subnet-mask 255.255.255.0;
default-lease-time 600;
max-lease-time 7200;
subnet 192.170.1.0 netmask 255.255.255.0{
        range 192.170.1.10 192.170.1.200;
        option broadcast-address 192.170.1.255;
        option routers 192.170.1.3;
}
```

Volvemos a arrancar el servicio # /etc/init.d/dhcp start

Nuestro servidor DHCP tiene la dirección 192.170.1.4

2.3 Configuración del Firewall

Para esta tarea hemos elegido una maquina Linux debian también virtualizada, valdría con cualquier distribución Linux con iptables.

La maquina tiene necesariamente dos interfaces de red, una conectada a la red de acceso, con dirección 192.170.1.3 y la otra conectada a la red que queremos proteger, con ip X.X.X.X.

3 Servidor

3.1 Instalación del SRF

1. Ejecutar el instalador del SRF y seguir las indicaciones de la pantalla.
2. Ejecutar el instalador del servicio de monitorización (IMPORTANTE: no modificar la carpeta de instalación del Apache Tomcat!!). Una vez instalado el mismo hay que hacer algunas cosas:
    1. Iniciar el Apache Tomcat, hay un menú en Inicio.
    2. Una vez iniciado el Apache Tomcat ejecutar el archivo "deploy.bat" que se encuentra en la carpeta "C:\GuardConditionService".

3. Cerrar el Apache Tomcat. Una vez cerrado asegurarse de borrar todos los archivos ".jar" que estén dentro del directorio "C:\Archivos de programa\Apache Software Foundation\Tomcat 5.0\common\endorsed".

3.2 Instalación de la aplicación Access server

Ejecutar el instalador del Acces Server. A continuación modificar el fichero "parámetros.properties", situado en la carpeta config del directorio de instalación donde se cambian los parámetros de la red protegida y todas las características configurables de la aplicación. Lo fundamental de este fichero es cambiar los siguientes valores:

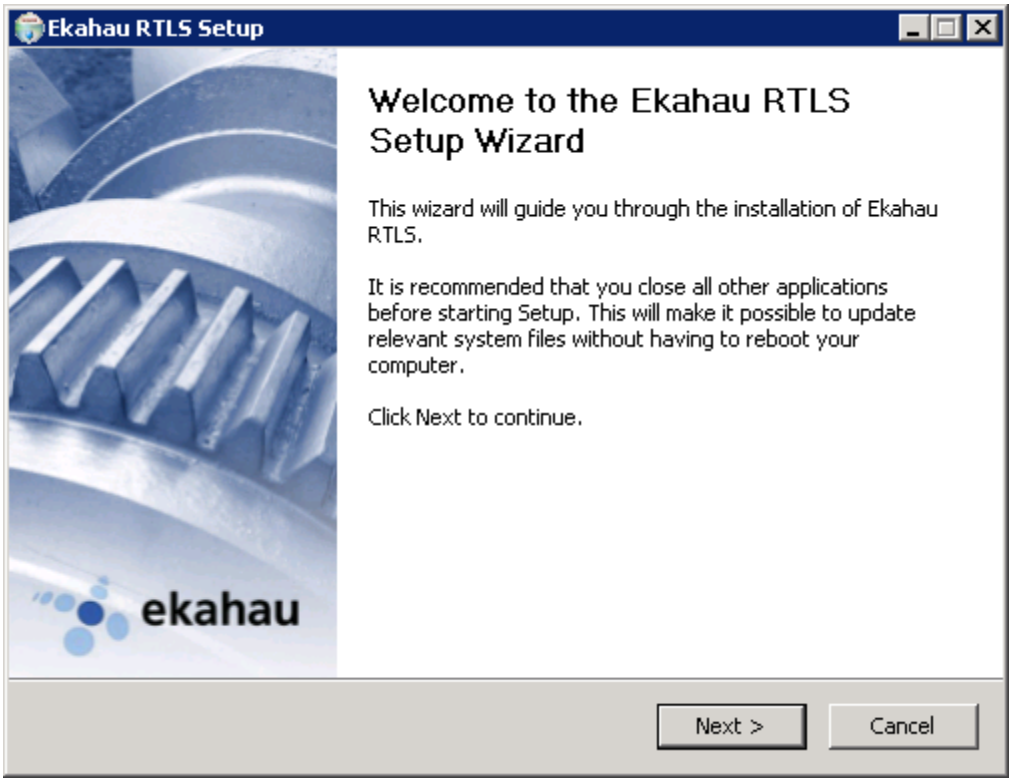
1. **localNet**, Dirección de la red protegida, sirve para ejecutar las sentencias de iptables teniendo en cuenta si es acceso local o externo
2. **gw**, Gateway de la red protegida
3. **dns**, Servidores DNS, deben ir separados por un espacio.
4. **localNetIP**, dirección ip del firewall de la interfaz conectada a la red protegida
5. **localNetMask**, mascara del firewall de la interfaz conectada a la red protegida.

3.3 Instalación de los ECs

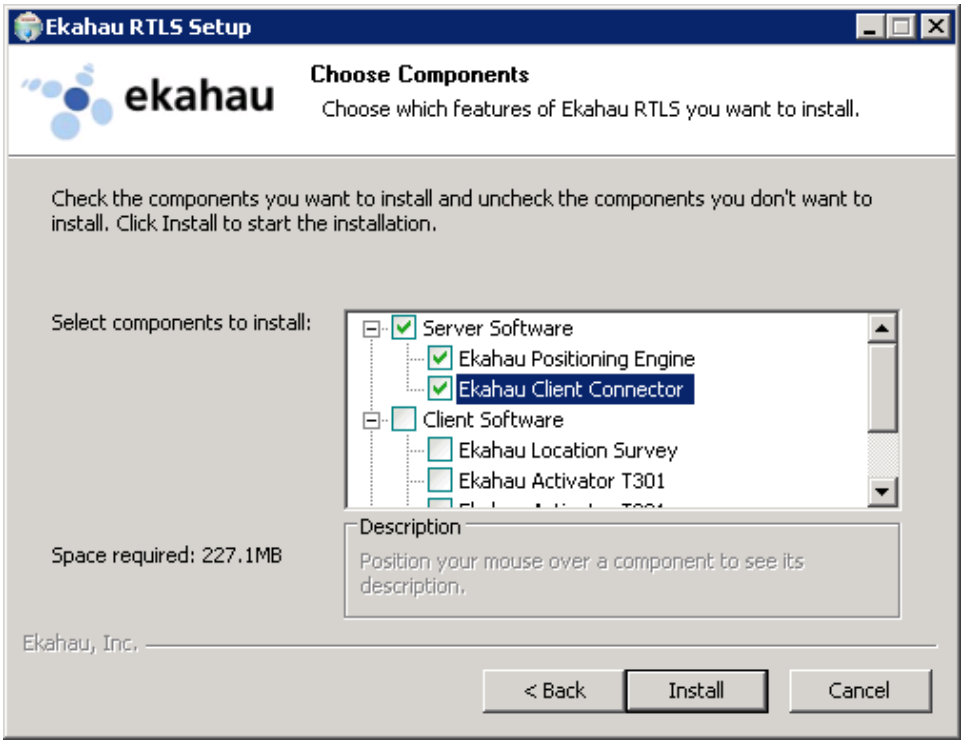
3.3.1 EC de localización

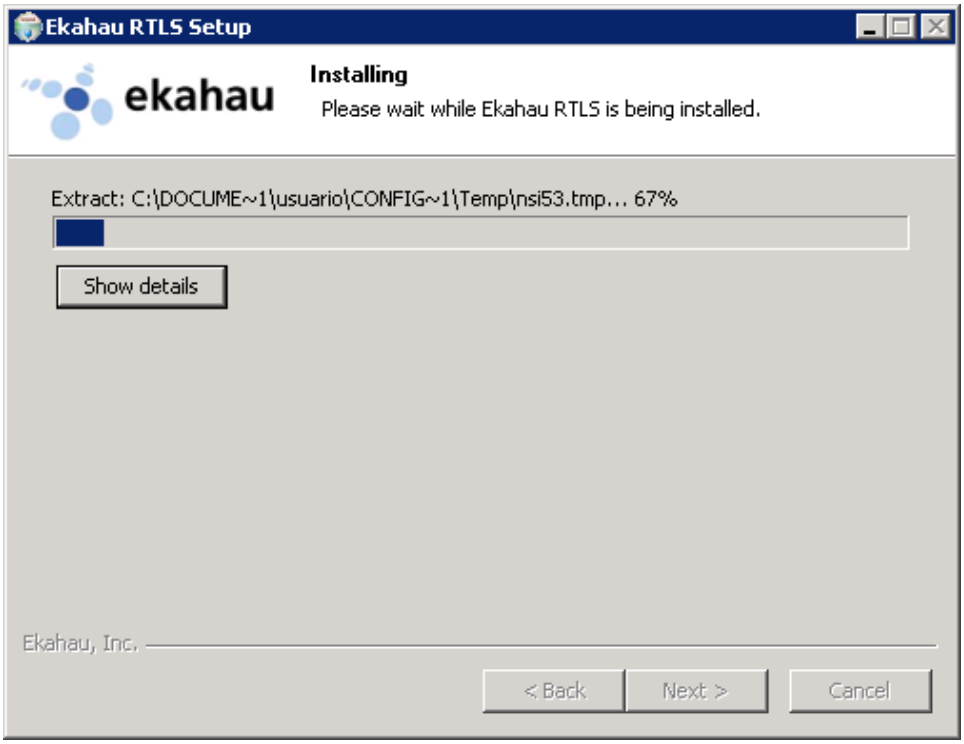
3.3.1.1 Ekahau RTLS.

1. Ejecutar el instalador del producto (*Ekahau RTLS-Setup.exe*). El instalador se encuentra en la carpeta *EC Localización.*

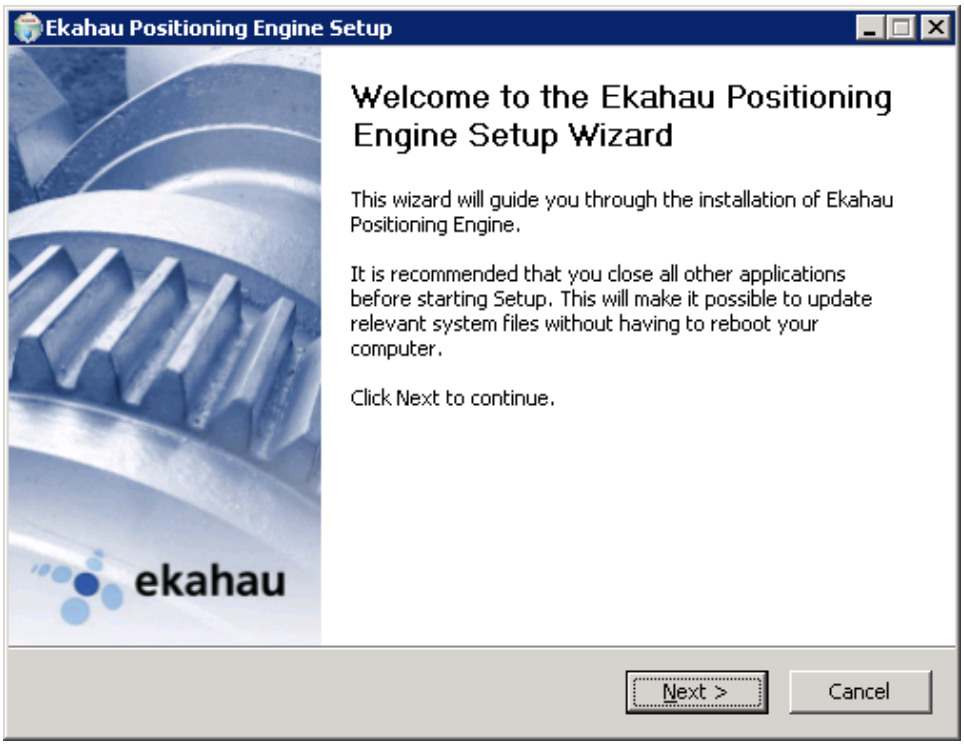


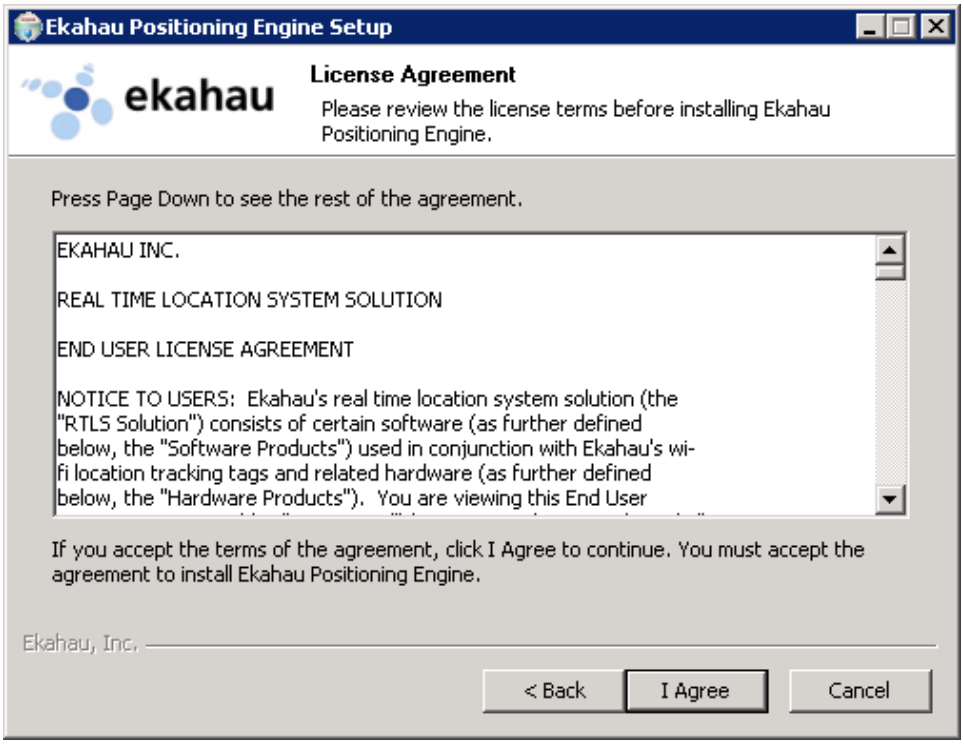
2. Activar únicamente las opciones de la rama *Server Software*: ***Positioning Engine*** y ***Client Connector***



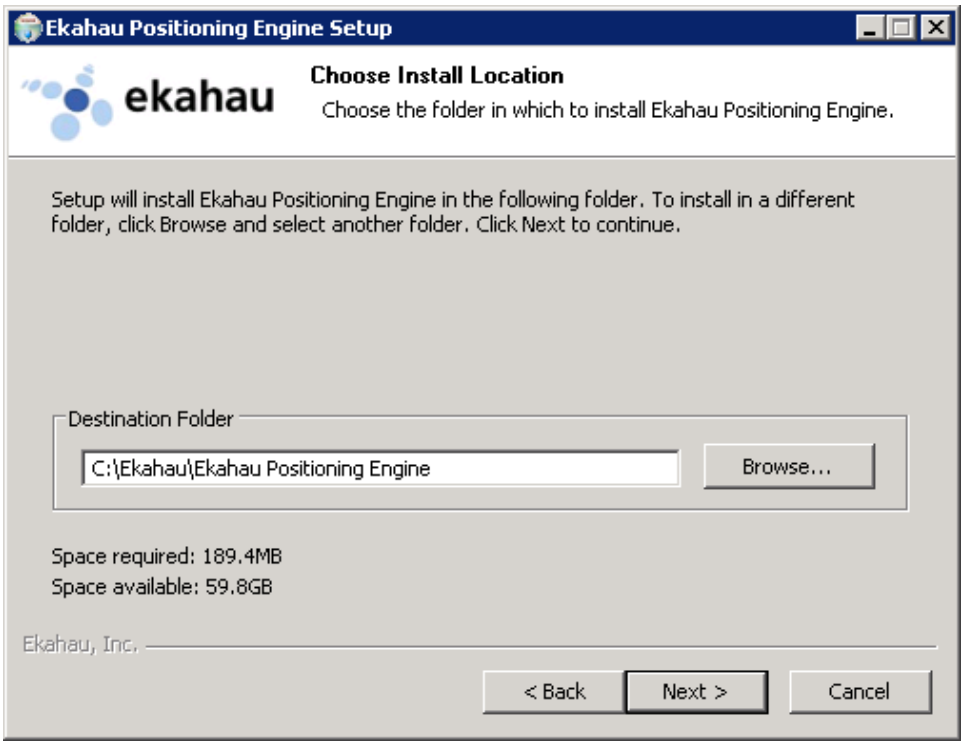


3. Instalación del ***Positioning Engine***.

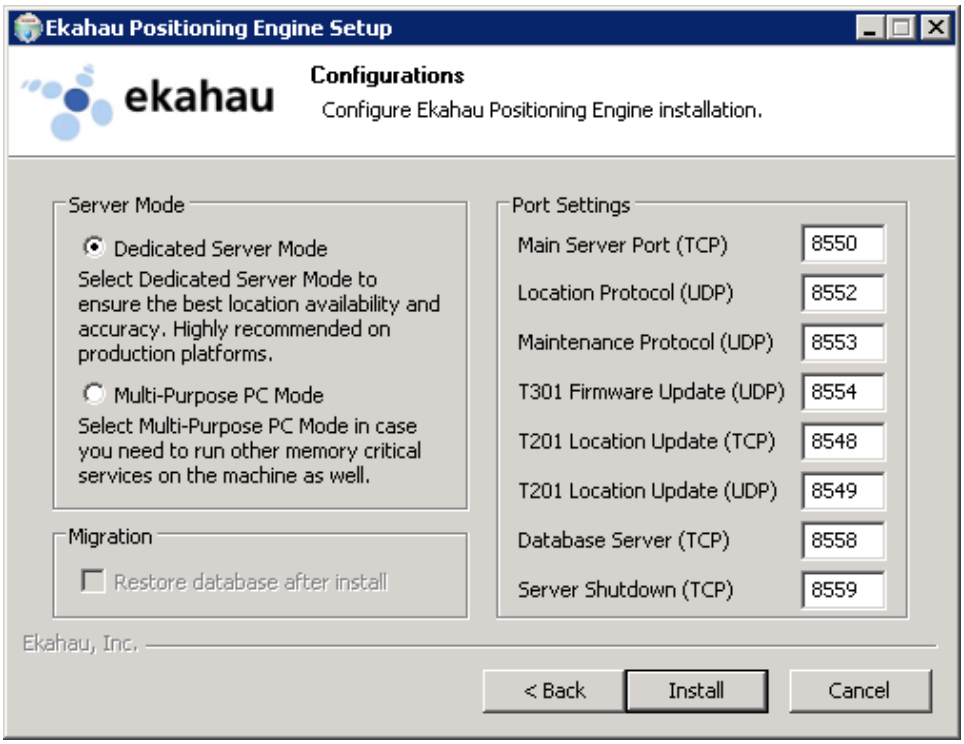


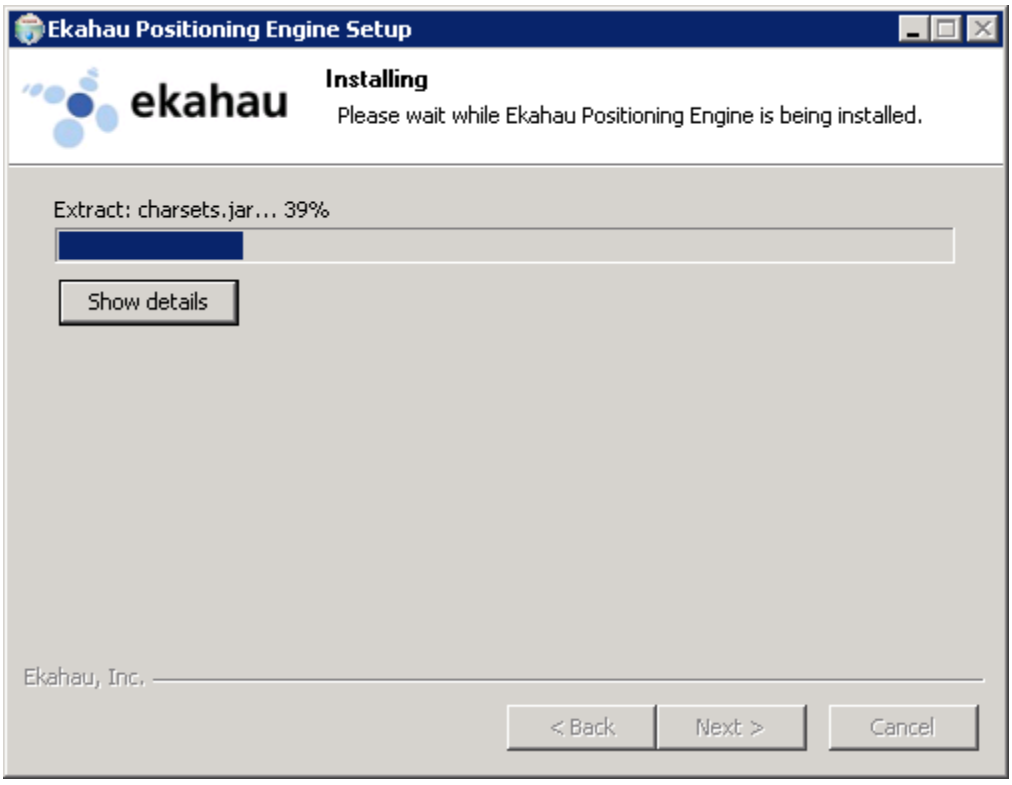


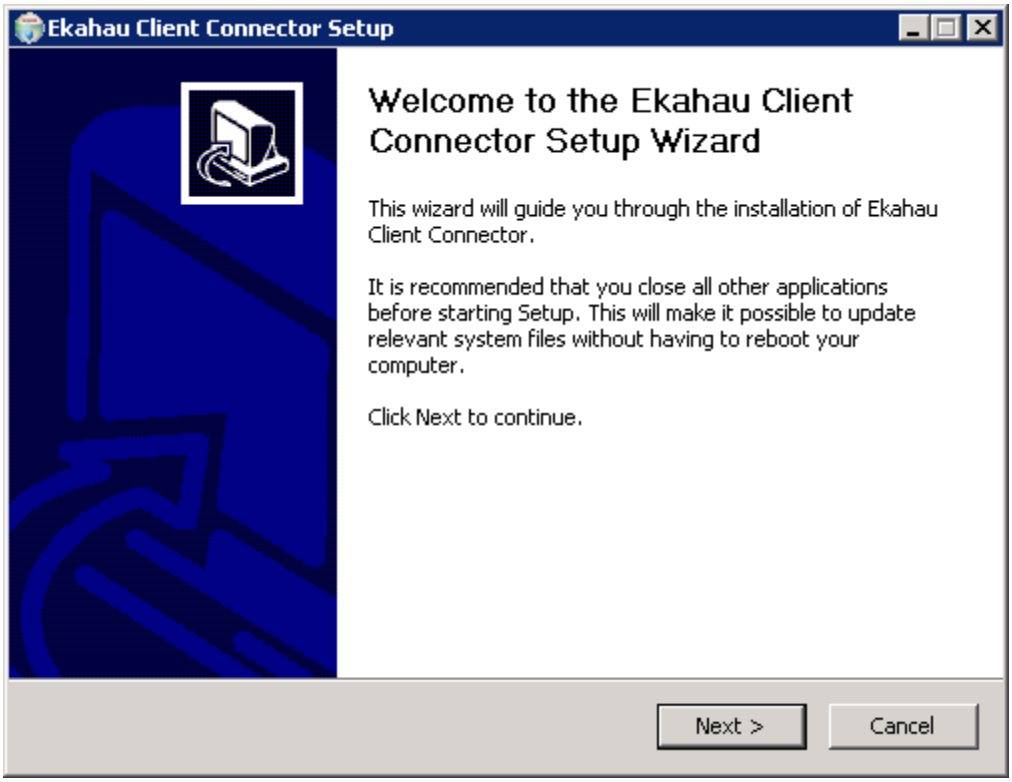
4. Elegir el directorio de instalación.



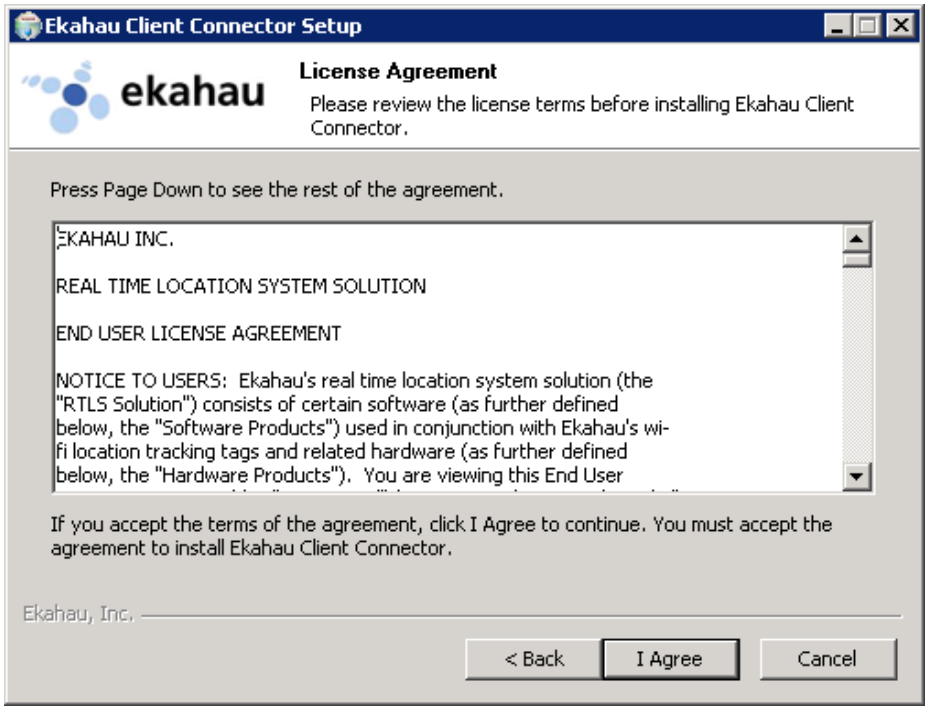
5. Dejar las opciones por defecto.



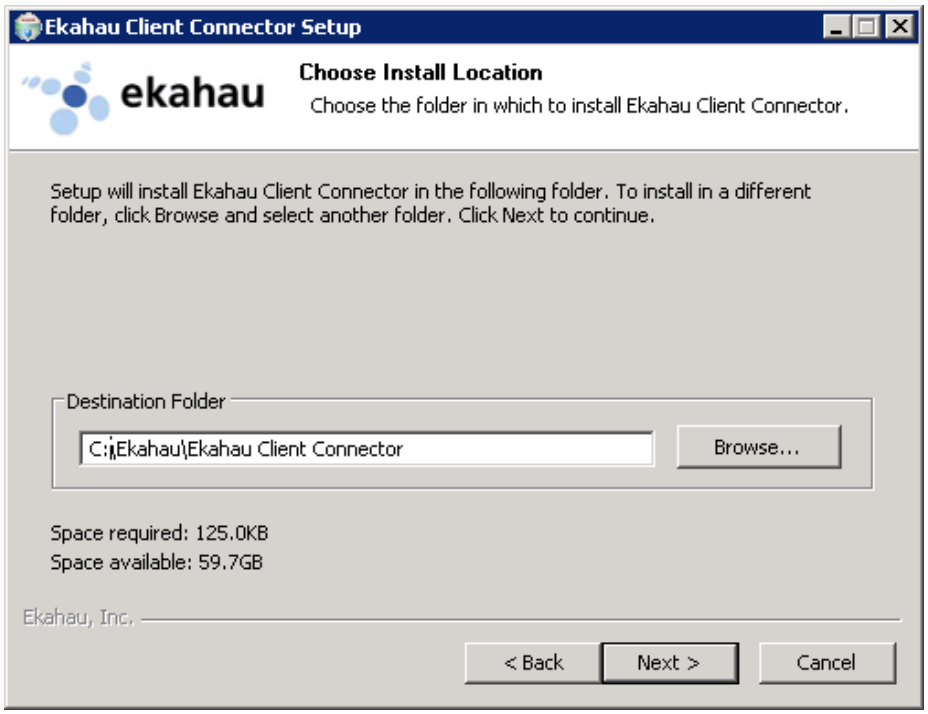




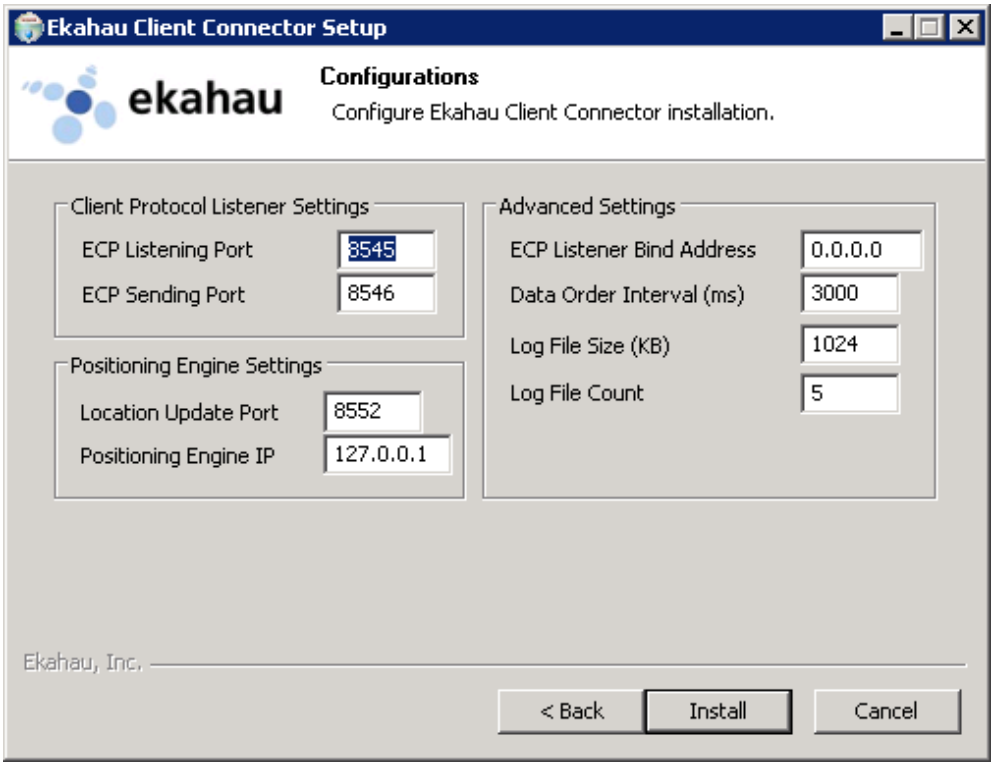
6. Instalación del ***Client Connector***.

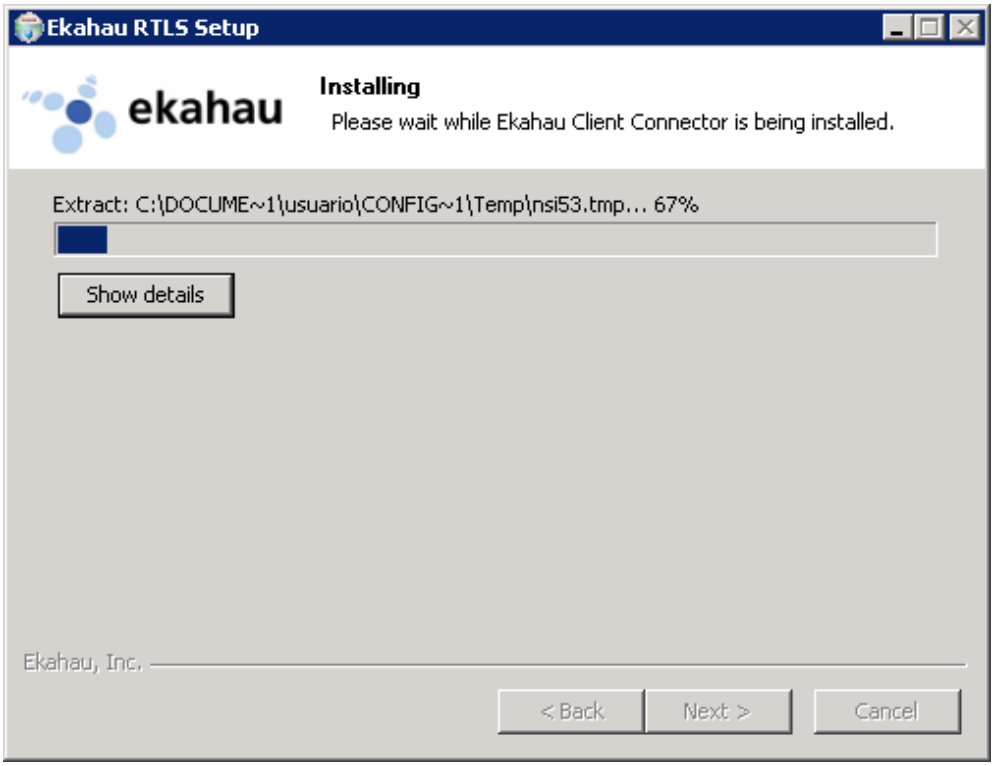


7. Elegir directorio de instalación.

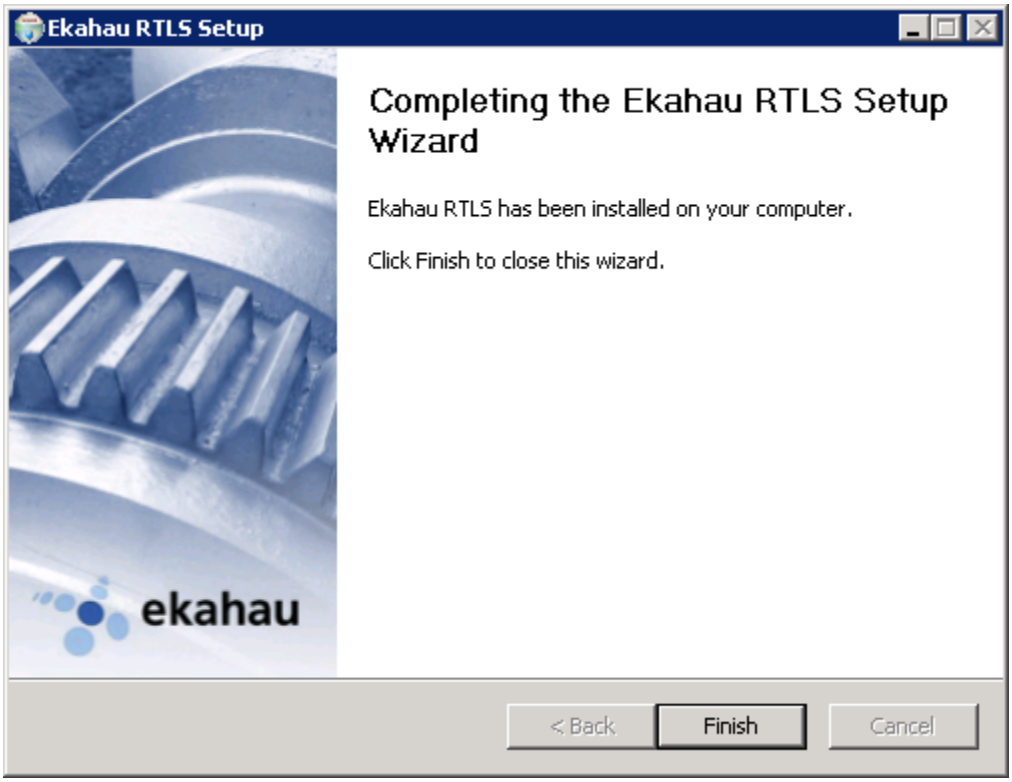


8. Dejar opciones por defecto.





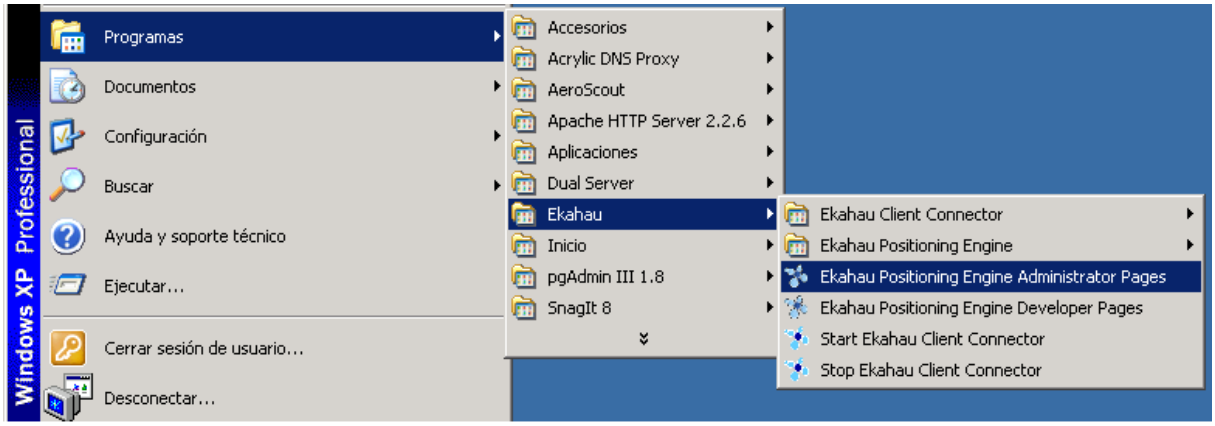
9. Fin de la instalación.

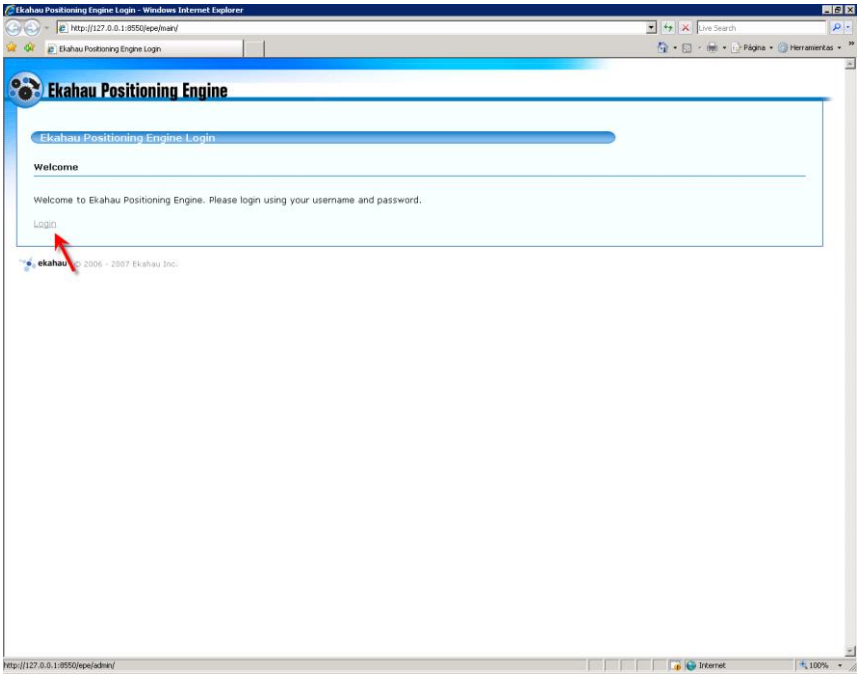


10. Introducir licencia

Para que Ekahau RTLS pueda funcionar es necesario cargar el archivo de licencia en el sistema. Para ello, Ekahau RTLS debe estar ejecutándose.

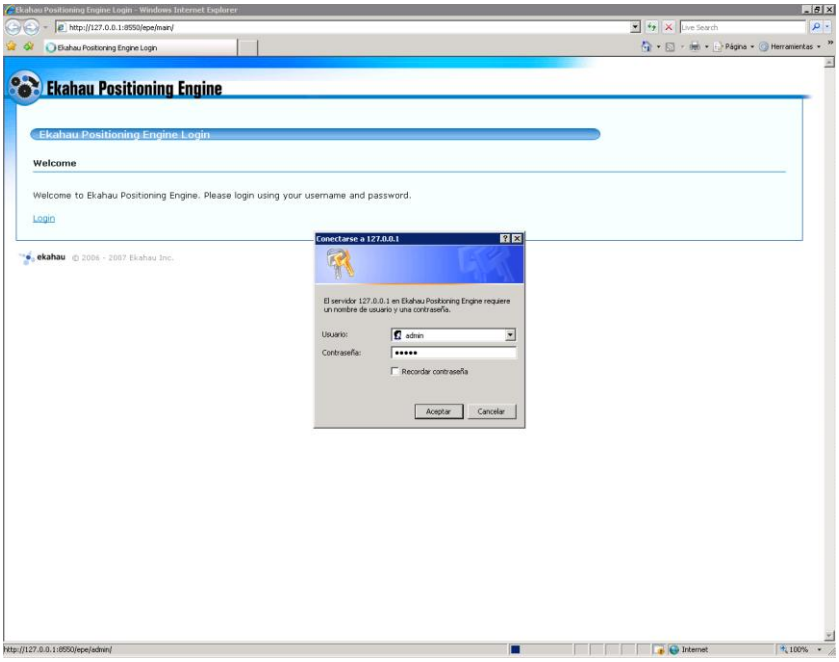
Arrancar la interfaz Web de administración de Ekahau. Ir a: *Inicio* → *Programas* → *Ekahau* → *Ekahau Positioning Engine Administrator Pages*.



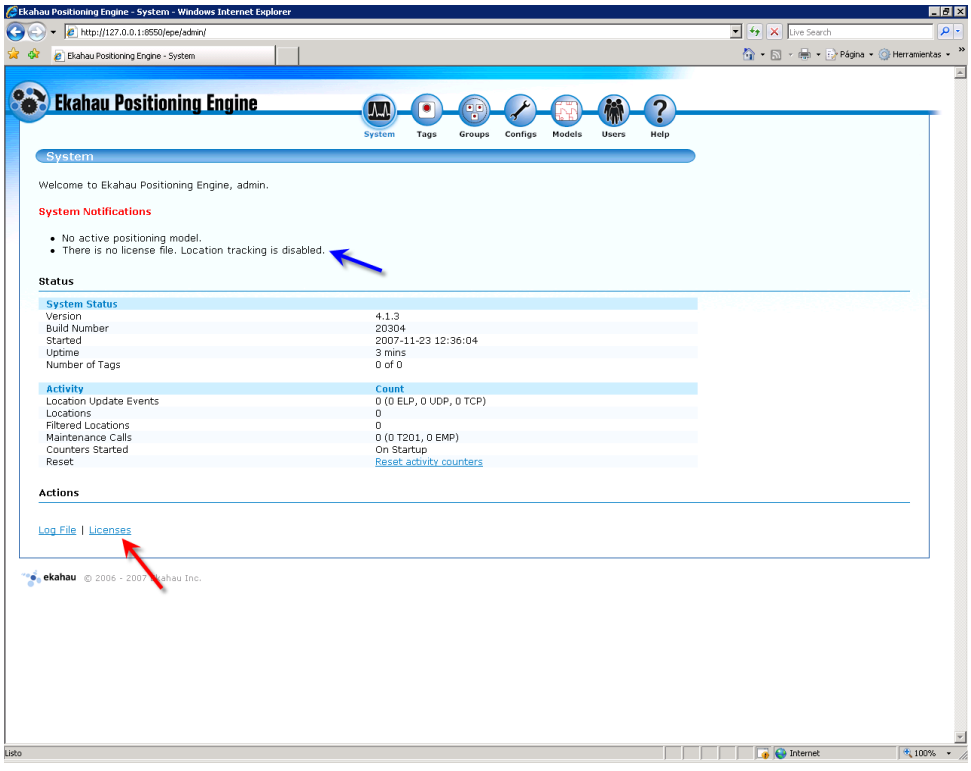


Introducir nombre de usuario y contraseña. Por defecto: admin en los dos campos.

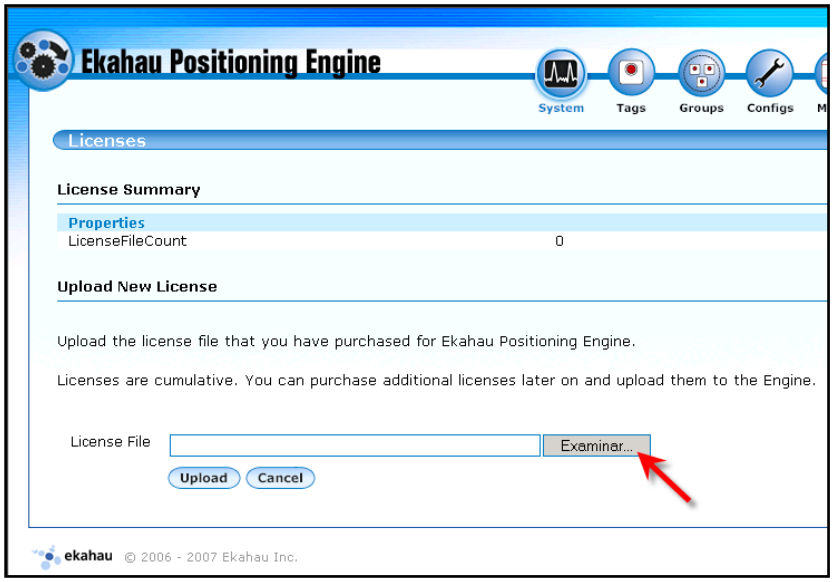
Nota: No cambiar el nombre de usuario y la contraseña, ya que la aplicación de ILSec está configurada con estos valores.

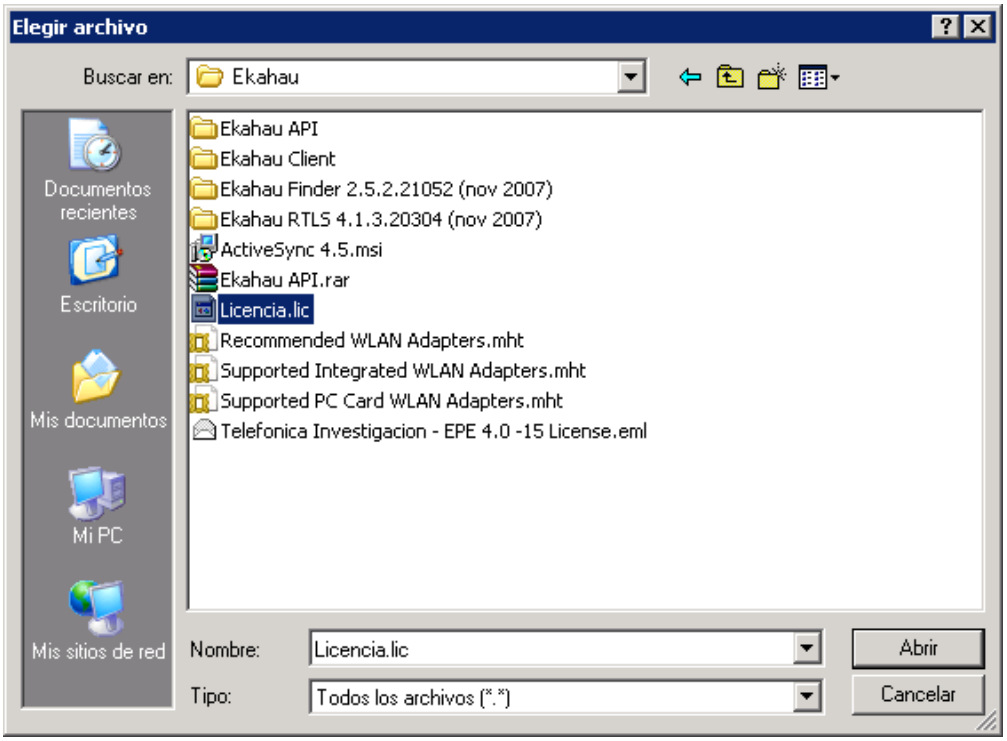


El sistema avisa de que no hay ficheros de licencia cargados (flecha azul). Ir a la sección de licencias (flecha roja).

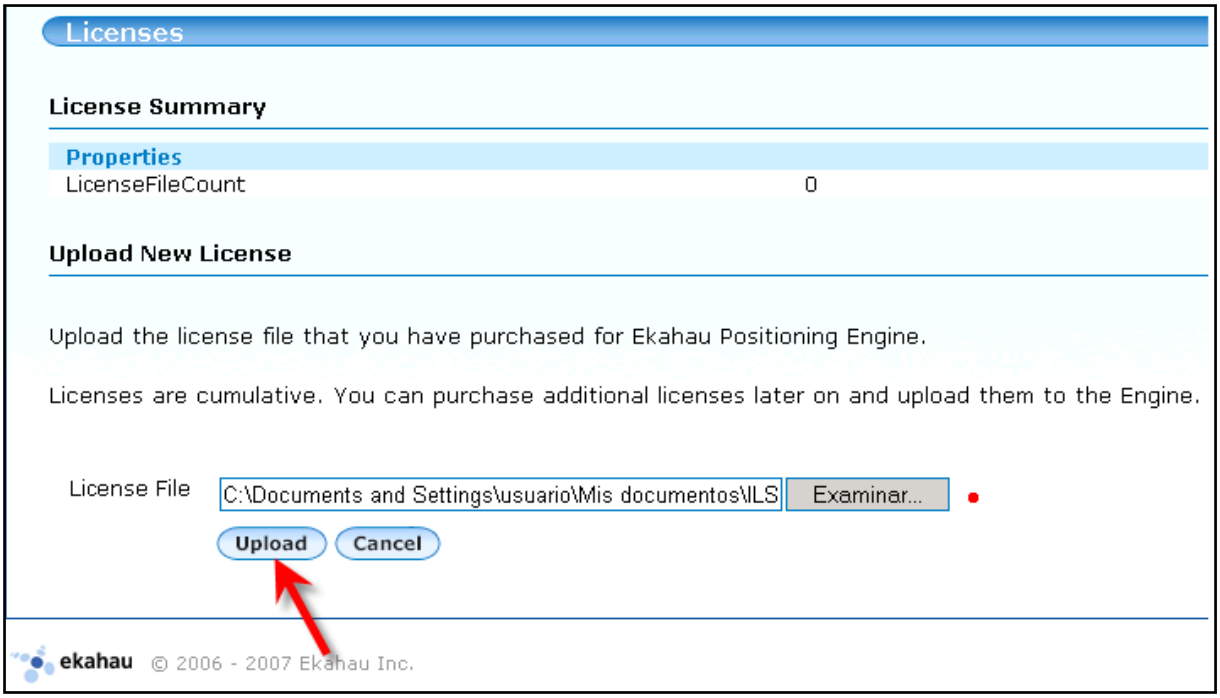


Cargar el fichero de licencia *Licencia.lic* que se encuentra en la carpeta *EC Localizacion\Ekahau RTLS*.

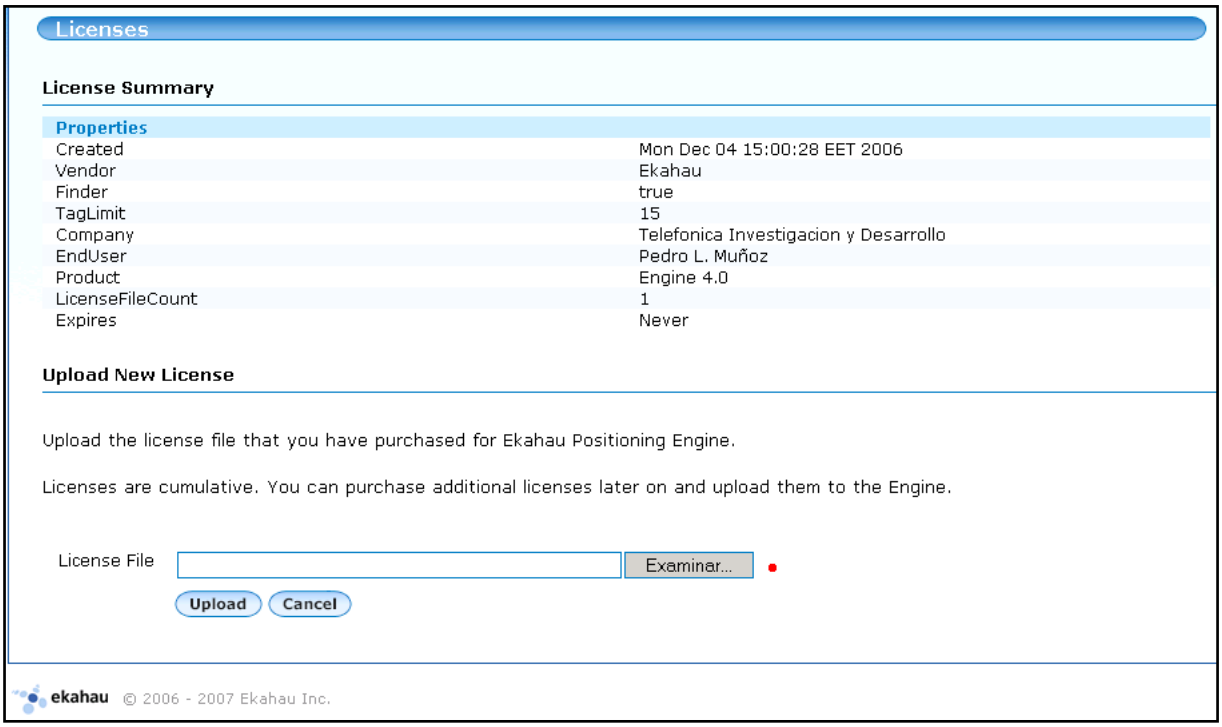




Pulsar el botón *Upload*.



En la interfaz aparece la información de la licencia.

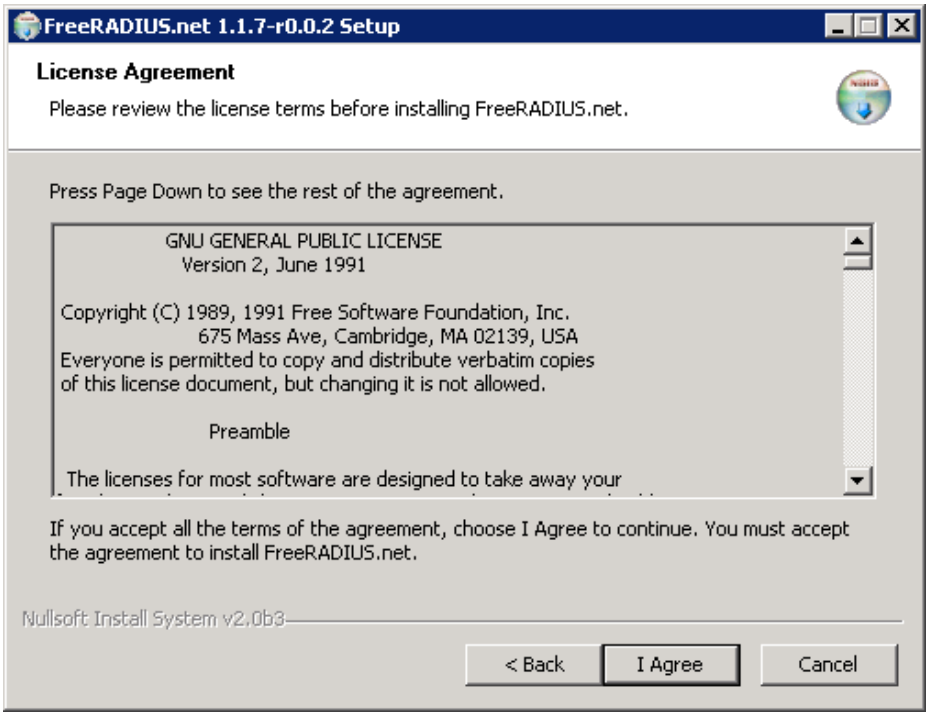


3.3.1.2 Instalación de los binarios del EC

1. Copiar la carpeta del EC de localización *LocZoneSM* en el directorio *execComps* de la aplicación SRF.
2. Ejecutar el BDSetUpLocalization.jar del directorio BBDD para crear la base de datos necesaria para su correcto funcionamiento
3. Agregar el patrón a la librearía del SRF según lo mencionado en el apartado de la memoria 4.2.2.2.

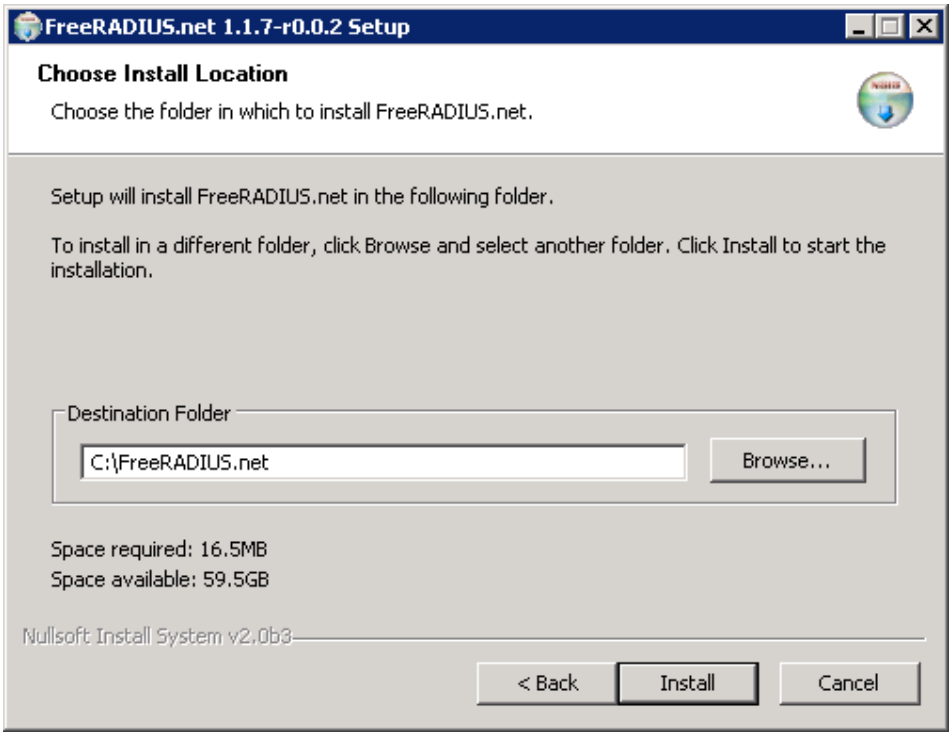
3.3.2 EC de identificaciónFreeRADIUS

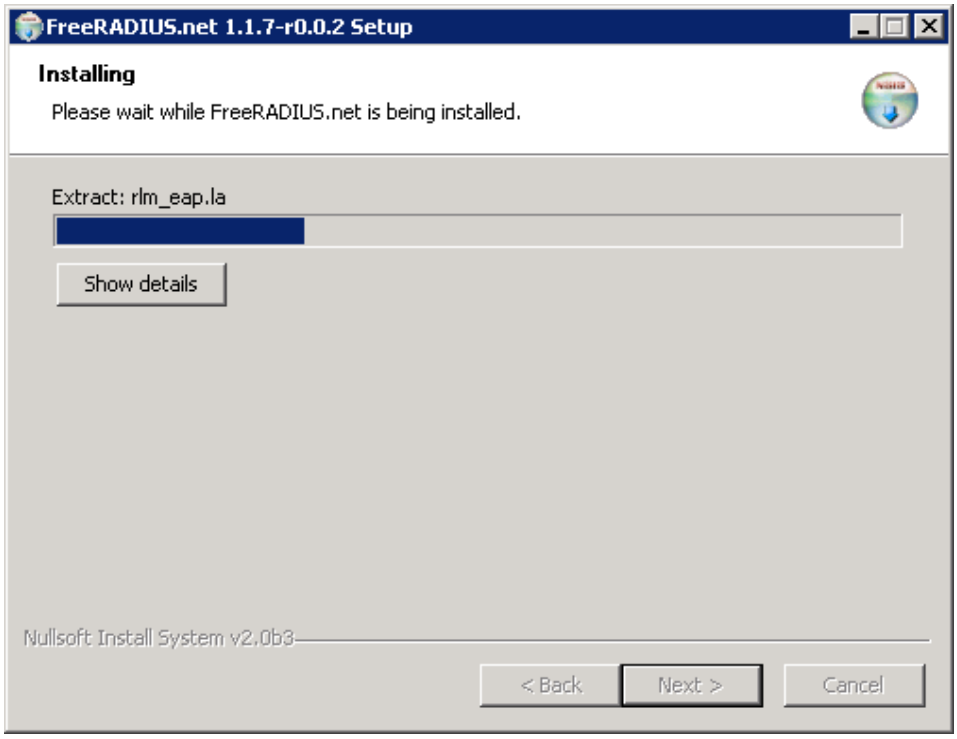
1. Ejecutar el instalador del producto (FreeRADIUS.net-1.x.exe). El instalador se encuentra en la carpeta EC Identificacion\FreeRADIUS.



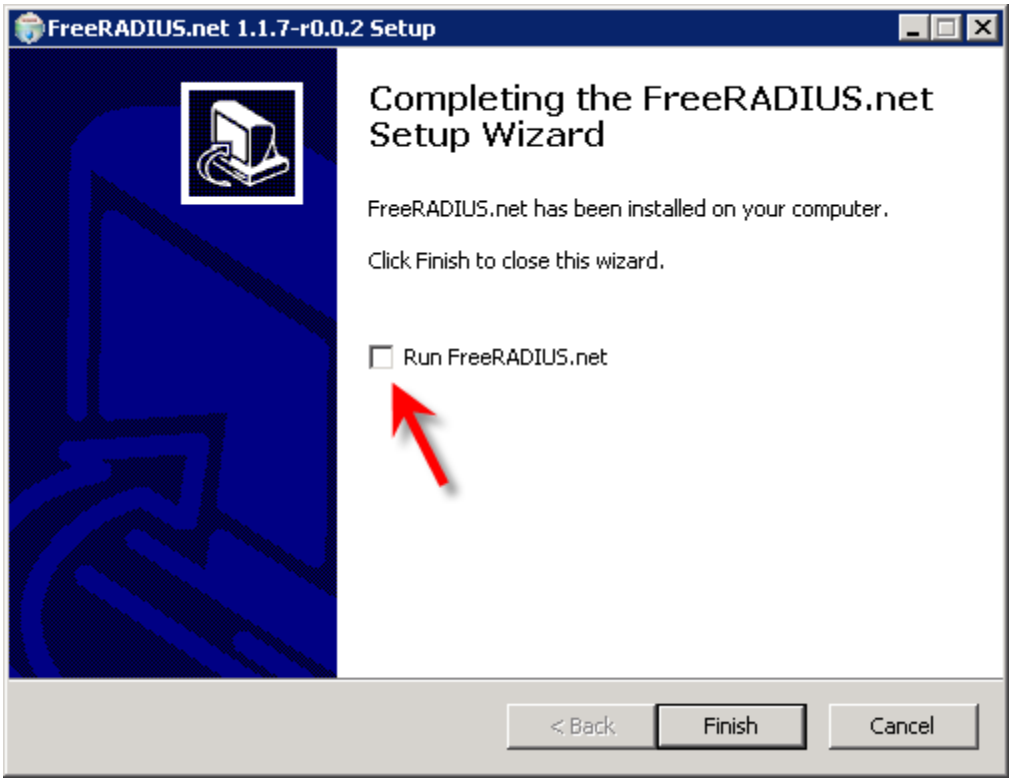
2. Elegir el directorio de instalación.

Nota: El directorio de instalación no debe incluir espacios en blanco.





3. Fin de la instalación. Desactivar la casilla Run FreeRADIUS.net.



3.3.2.1.1 Instalación de los ficheros de configuración de FreeRADIUS

Una vez acabada la instalación coger la carpeta comprimida que se encuentra en el directorio *EC Identificacion\FreeRADIUS* y copiarla en *C:\FreeRADIUS.net\etc, suponiendo que C:\FreeRADIUS.net sea el directorio de instalación de FreeRADIUS.*

3.3.2.2 Instalación de los binarios del EC

1. Copiar la carpeta del EC de identificación IdentitySM en el directorio execComps de la aplicación SRF.
2. Ejecutar el BDSetUpIdentity.jar del directorio BBDD para crear la base de datos necesaria para su correcto funcionamiento
3. Agregar el patrón a la librearía del SRF según el apartado de la memoria 4.2.2.2.

3.3.3 EC de identificación de dispositivos vía TPM

3.3.3.1 Instalación de los binarios del EC

1. Copiar la carpeta del EC de TPM TPMECServer en el directorio execComps de la aplicación SRF.
2. Ejecutar el BDSetUpTPM.jar del directorio BBDD para crear la base de datos necesaria para su correcto funcionamiento
3. Agregar el patrón a la librearía del SRF según el apartado de la memoria 4.2.2.2.

4 Cliente

4.1 Instalación en un ordenador portátil.

4.1.1 Instalación de la aplicación Access Client.

1. Ejecutar el instalador del Access Client.
2. Si el ordenador portátil no dispone de dispositivo TPM, pasar directamente al punto 4.1.4.

4.1.2 Instalación del SRF.

1. Ejecutar el instalador del SRF y seguir las indicaciones de la pantalla.
2. En este caso para el cliente no es necesario instalar el servicio de monitorización.

4.1.3 EC de identificación de dispositivos vía TPM.

4.1.3.1 Instalación de los binarios del EC

1. Copiar la carpeta del EC de TPM TPMECClient (situada en cliente\EC TPM) en el directorio execComps de la aplicación SRF.
2. Agregar el patrón a la librería del SRF según el apartado de la memoria 4.2.2.2.

4.1.4 Instalación de software para el cliente.

1. Instalar el software de ekahau para el cliente. Este software se encuentra el la carpeta ekahau dentro de la carpeta cliente.

4.1.5 Configuración de la conexión en el cliente.

Ahora pasamos a describir los parámetros de conexión del equipo portátil a localizar. Se configurarán de la siguiente forma:

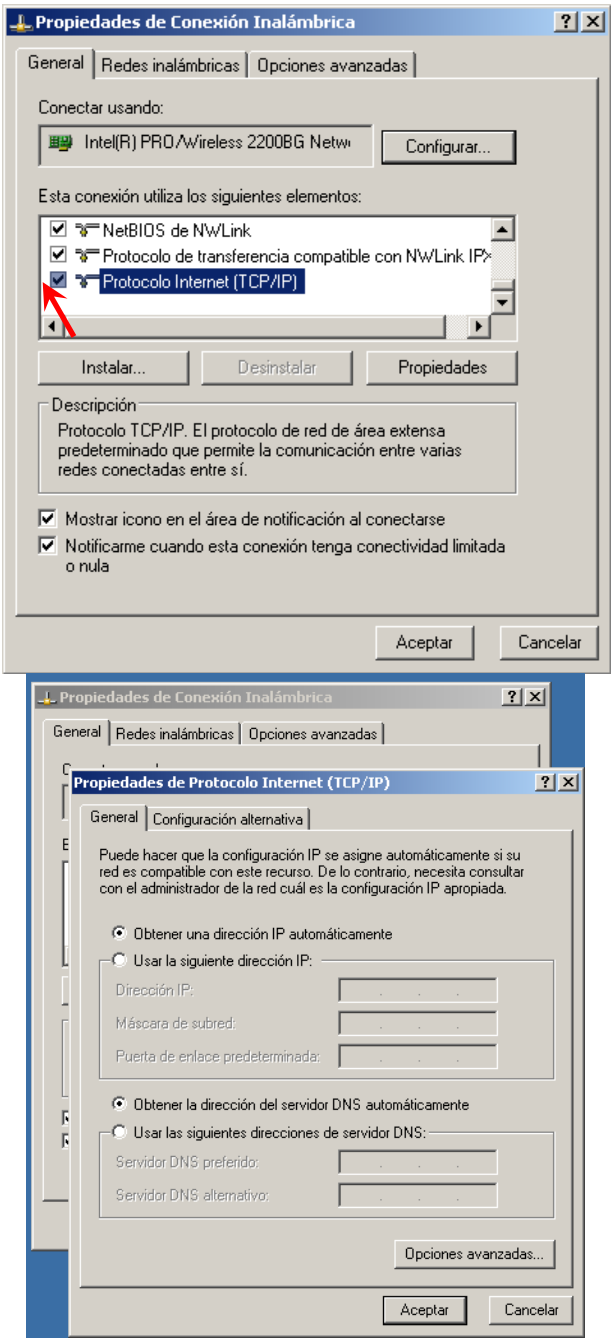
- Configuración IP: Asignada por DHCP.
- Configuración de seguridad inalámbrica:
    - Nombre de red (SSID): RedILS
    - Autenticación de red: abierta o WPA Empresarial
    - Cifrado de datos: WEP (para autenticación abierta) o AES (para WPA).
    - Configuración 802.1x:
        - Tipo de autenticación (EAP): PEAP

- Método de autenticación: MS-CHAPv2
- Certificado de servidor: NO

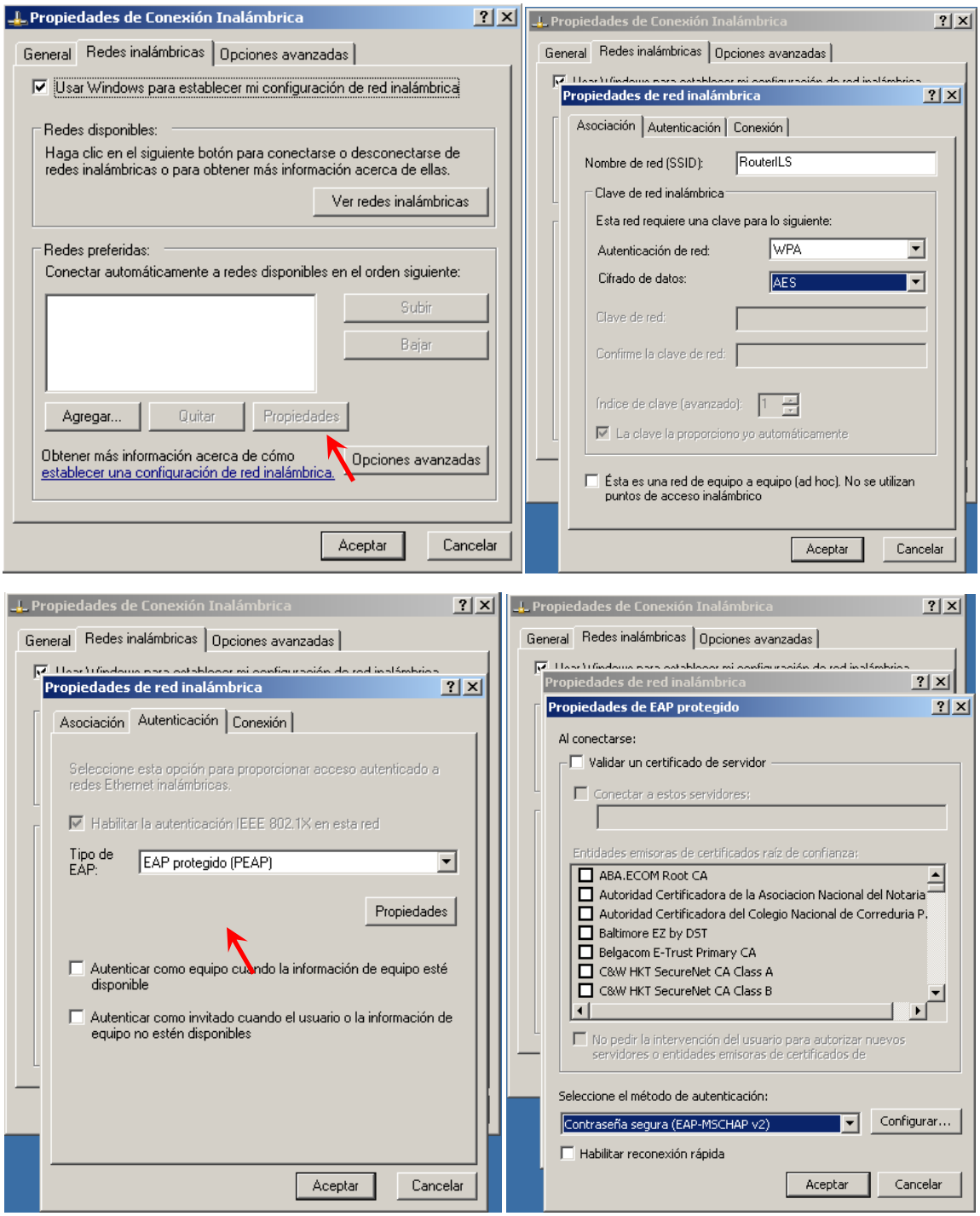
Nota: La configuración de seguridad (WEP o WPA) debe coincidir con la configurada en el router.

Para ello, se pueden seguir las siguientes pantallas:

Configurar la IP: Panel de Control → Conexiones de Red → Conexión Inalámbrica (botón derecho) → Propiedades → Pestaña General. Seleccionar Protocolo Internet (TCP/IP) y pulsar en Propiedades



Configuración de seguridad inalámbrica: *Propiedades de Conexión Inalámbrica* → *Pestaña Redes Inalámbricas* → *Agregar*



Nota: Cuando se introducen las credenciales de usuario al hacer la demostración, Windows las almacena y no las vuelve a pedir. Para borrar las credenciales y que Windows las pida cada vez que nos conectemos al router, hay que ejecutar el archivo *borrar.bat* en el equipo cliente. Dicho archivo se puede copiar desde *Cliente*.

4.2 Instalación en un dispositivo PDA.

4.2.1 Instalación de la máquina virtual Mysaifu.

4.2.1.1 Sistemas soportados por JVM Mysaifu

- Windows Mobile 6.0
- Windows Mobile 5.0
- Windows Mobile 2003 Second Edition software for Pocket PC (Pocket PC 2003 SE)
- Windows Mobile 2003 software for Pocket PC (Pocket PC 2003)

4.2.1.2 Instalación de Mysaifu

1. Copiar el archivo jvm.Release.CAB (se encuentra en cliente\Client PDA) en \Mis Documentos.
2. Doble click en el archivo jvm.Release.CAB.

4.2.2 Configuración de la aplicación AccessClientPDA

Hay que modificar el código fuente de la aplicación *AccessClientPDA.java* de la siguiente manera:

1. Mirar la dirección MAC del dispositivo PDA que vamos a utilizar.
2. Asignar estos valores al array de bytes MAC en las líneas de código 20 a la 26.
3. Generar el .jar del código.

4.2.3 Ejecución de la aplicación AccessClientPDA

1. Copiar el archivo AccessClientPDA.jar a la PDA.
2. Ejecutar Mysaifu JVM.
3. Una vez en la pantalla de Mysaifu JVM, seleccionar en type => JAR file. Y en browse buscar nuestro AccessClientPDA.jar.
4. Una vez hecho todo esto hay que darle a execute y ya tenemos la aplicación funcionando.

4.2.4 Configuración de la conexión a la RedILS

Para configurar la conexión a la RedILS se remite a la configuración para el cliente detallada en la sección 4.1.5 del ANEXO A.

4.3 Preparación de una demo

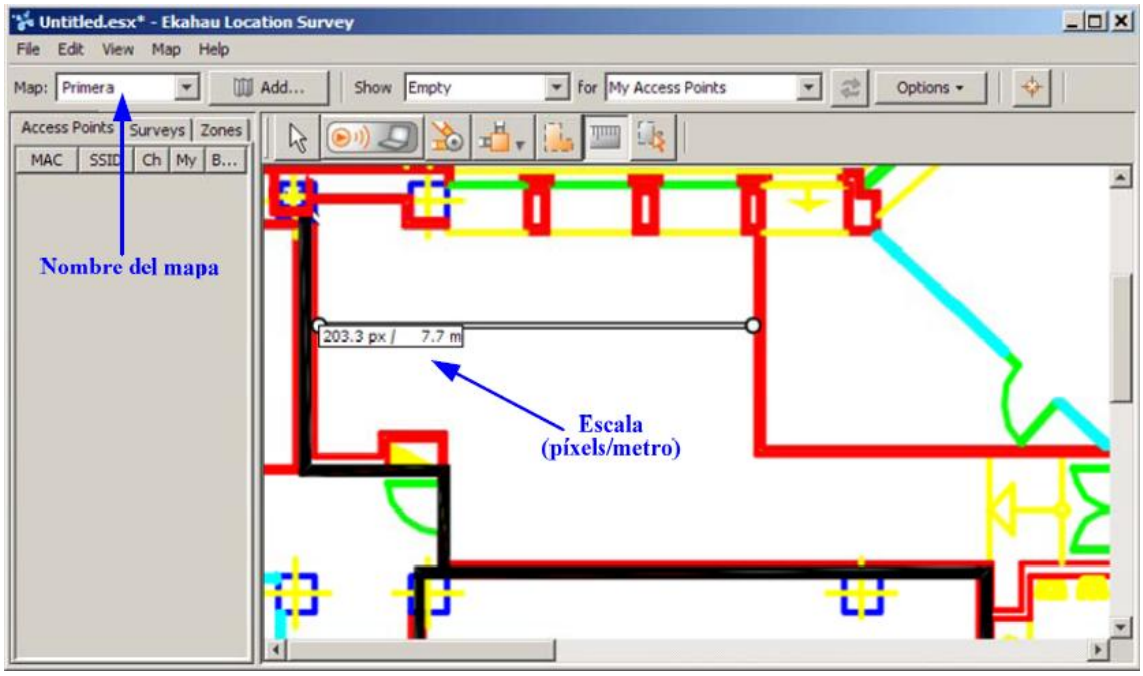
4.3.1 Creación del modelo de posicionamiento de Ekahau

Para que Ekahau pueda llevar a cabo las funciones de localización es necesario realizar un proceso previo de calibración del sistema durante el cual se crea un modelo del entorno. En dicho modelo se relaciona la información de las señales radioeléctricas con el lugar en el que estas señales han sido recogidas.

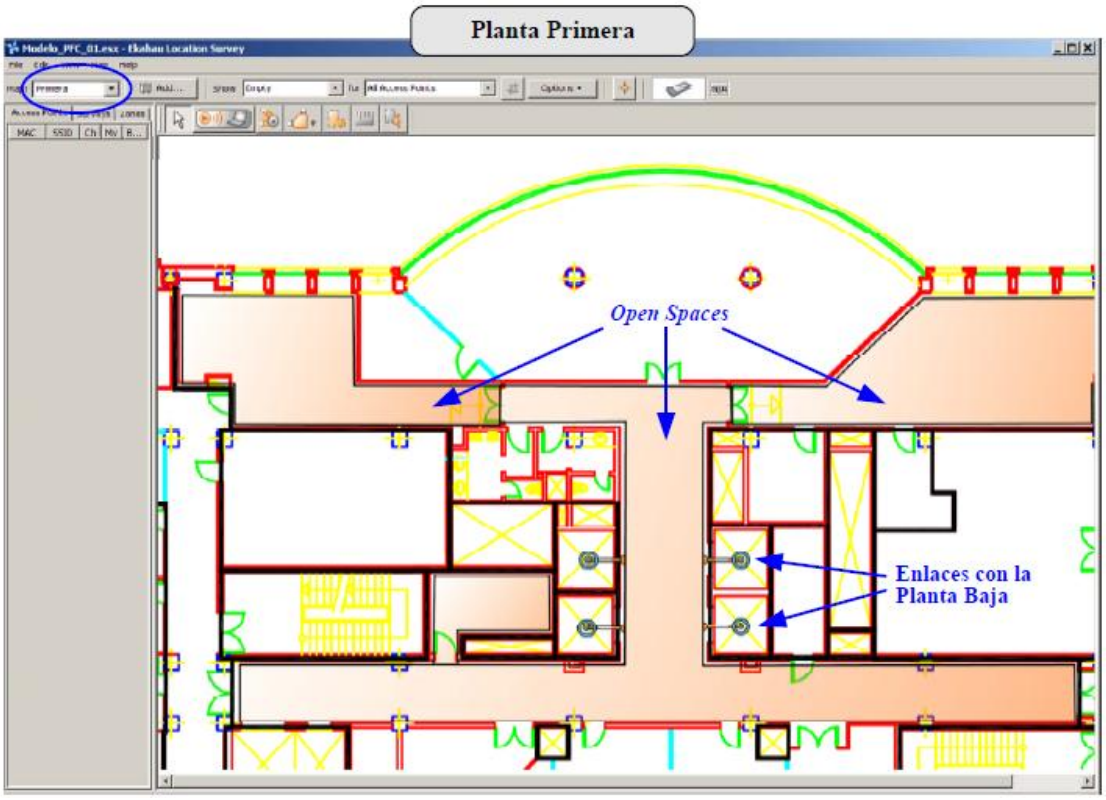
Para crear los modelos de posicionamiento, Ekahau dispone de una herramienta llamada *Ekahau Location Survey*.

A continuación se expone el proceso seguido:

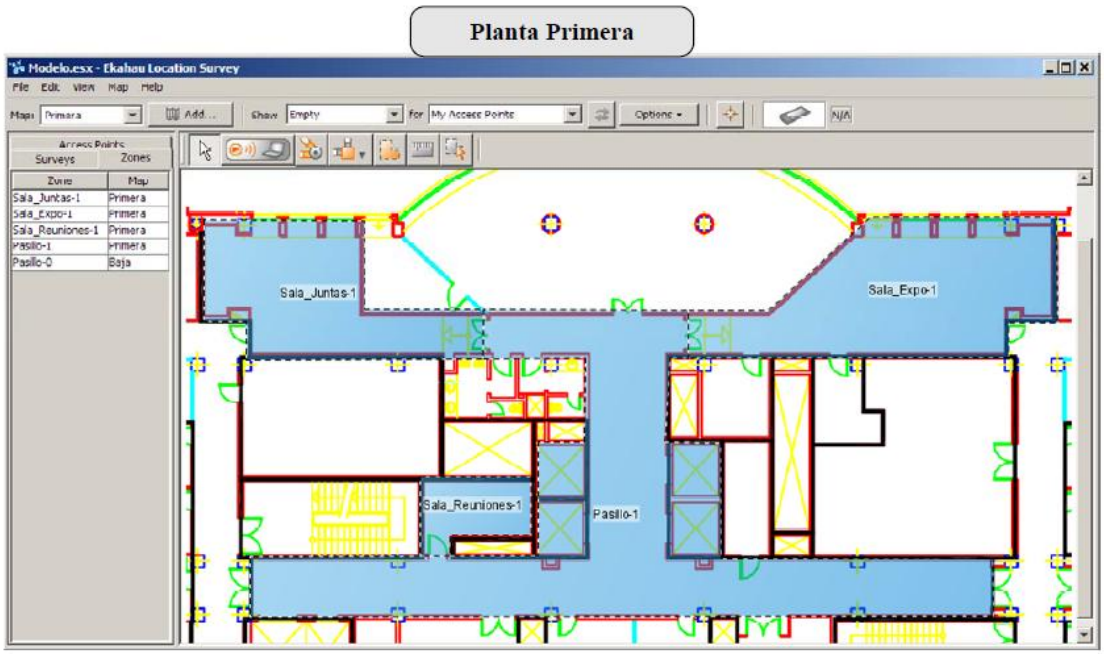
1. Añadir los mapas y fijar la escala: En este caso se han añadido dos mapas, uno por cada planta definida en la tabla plantas de la Base de Datos. A los mapas del modelo hay que asignarles el mismo nombre que consta en la Base de Datos, para que la Aplicación de Control pueda relacionar los mapas definidos en Ekahau con las plantas definidas en la Base de Datos.



2. Definir el entorno: Todo el entorno se definió, únicamente, empleando Open Spaces.



3. Definir zonas: Al igual que ocurre con las plantas, los nombres de las zonas definidas en Ekahau deben coincidir con los de la tabla zonas de la Base de Datos.



4. Calibrar el modelo: para calibrar el modelo hay recorrer todos los espacios en los que se han definido Open Spaces, indicando los lugares por los que se va pasando.
5. Guardar el modelo de posicionamiento y cargarlo en el EPE: finalmente se guarda el modelo de posicionamiento, se carga, y se activa en el Ekahau Positioning Engine.

4.3.2 Pasos a seguir

4.3.2.1 Servidor

1. Asegurarse de que las máquinas están arrancadas, sean virtuales o no.
2. Cerciorarse de que el servidor está conectado al router principal de la RedILS.
3. Deshabilitar el Wi-Fi del servidor, ya que en el caso de que esté activo el servicio Ekahau no localiza correctamente a los clientes.
4. Arrancar el FreeRadius, para que se puedan conectar los clientes.
5. Arrancar el Tomcat. Aparecerá un icono en la barra de tareas.
6. Iniciar el SRF, primero el rmi_Registry y después el SRF.
7. Arrancar la aplicación servidor "AccessServer".
8. Si aparece un mensaje de error que se llama "cannot connect the firewall" hay que cerciorarse de que la conexión al router es correcta, que la máquina esta arrancada. Si aún así sigue dando el error hay que hacer un reboot de la máquina Firewall.
9. En la pestaña de estado de patrones ver que se están ejecutando todos los ECs.

4.3.2.2 Cliente

1. Conectarse por Wi-Fi a la red "RedILS".
2. Iniciar el SRF, primero el rmi_Registry y después el SRF. En el caso de no tener instalado el SRF en el cliente no ejecutar este punto.
3. Ejecutar el AccesClient.jar, o el AccessClientPDA.jar dependiendo del caso.
    - En el caso del cliente PDA, si hay una reconexión del Wi-Fi hay que volver a iniciar el "AccessClientPDA" (presionar close y volver a ejecutarlo).

ANEXO B Archivo de configuración de la Aplicación de Control

```
# FICHERO DE CONFIGURACION DE LA APLICACIÓN DE CONTROL

##Configuracion del Firewall

#Datos de la maquina virtual
ipMWare=192.170.1.3
usrMWare=root
pwdMWare=serenity

#Direccion de la red local, sirve para ejecutar las sentencias de iptables teniendo en cuenta
#si es acceso local o externo
localNet=192.168.0.1/255

#Gateway de la red
gw=192.168.0.239


#Servidores DNS (separados por espacio)
dns=192.168.0.239

#Direccion ip y mascara del firewall de la interfaz conectada a la red local
localNetIP=192.168.0.100
localNetMask=255.255.255.0

#Base de datos
urlDB=jdbc:mysql://localhost/accesscontrol
loginDB=accesscontrol
pwdDB=serenity

##TIEMPOS DE REFRESCO
#Tiempo de refresco
T_CHECK=1000

#Localizacion
T_LOCATION=2000
```

```
#User
T_USER=20000

#Device
T_DEVICE=30000
```

ANEXO C Archivos XML de configuración de las Soluciones

En este anexo visualizamos el modelo de archivo de configuración XML. Ponemos los archivos de la solución de Identificación de Usuarios como ejemplo.

1 Clase

- En primer lugar tenemos la clase.

```
<?xml version="1.0" encoding="utf-8" ?>
-  <!--    Created  with  Liquid  XML  Studio  1.0.8.0  (http://www.liquid-
  technologies.com)
  -->
-     <SandDClass     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="D:\SRFv2.2\serenity\schemas\S&DClass_v1.x
  sd" name="SecurityMeasurement" domain="tid.es" version="1.0">
- <informationalPart>
- <creator>
  <name>tid.es</name>
  <date>-6135</date>
  </creator>
  <label>This class provides simple way to measure security of a situation
  based on a determined criteria</label>
  <comments>Criteria  is  not  definided  at  this  level.  Each  pattern
  belonging to this class define a specific criteria</comments>
- <providedProperties>
- <property>
  <name>quantitative security</name>
  <domain>tid.es</domain>
  <version>1.0</version>
  <timestamp>-9963</timestamp>
  </property>
  </providedProperties>
- <solutionFeatures>
  <feature>Quantitative Security</feature>
  </solutionFeatures>
- <roles>
- <role>
  <roleName>clienteIProfile</roleName>
  <description>tid.es</description>
- <interface>
- <calls>
- <call>
  <callName>measure</callName>
  <signature>int measure(String);</signature>
  <description>This  operation  measure  an  user  acording  to  his/her
  profile</description>
  </call>
  </calls>
- <sequence>
- <step>
  <order>1</order>
  <callName>measure</callName>
  </step>
  </sequence>
  </interface>
  </role>
  </roles>
  </informationalPart>
```

```
- <operationalPart>
  <trustMechanisms>TO BE DEFINED</trustMechanisms>
- <validity>
  <validFrom>-7339</validFrom>
  <validUntil>-4054</validUntil>
  </validity>
  </operationalPart>
  </SandDClass>
```

2 Patrón

- Ahora vemos el patron.

```
<?xml version="1.0" encoding="utf-8" ?>
-  <!--    Created  with  Liquid  XML  Studio  1.0.8.0  (http://www.liquid-
  technologies.com)
  -->
-          <SandDPattern          xmlns:tns="http://tempuri.org/ec/formula"
  xmlns:tnsa="http://www.omg.org/XMI"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="D:/SRFv2.2/serenity/schemas/S&DPattern_v1
  .xsd"      name="IdentityProfileSecurityMeasurement"      domain="tid.es"
  version="1.0">
- <informationalPart>
- <creator>
  <name>tid.es</name>
  <date>1214753928</date>
  </creator>
  <label>This pattern describes a mechanism to measure the security of an
  user based on his or her profile</label>
- <providedProperties>
- <property>
  <name>Profile-based security</name>
  <domain>tid.es</domain>
  <version>1.0</version>
  <timestamp>1214752496</timestamp>
  </property>
  </providedProperties>
- <roles>
- <role>
  <roleName>clienteIProfile</roleName>
- <interface>
- <calls>
- <call>
  <callName />
  <signature />
  </call>
  </calls>
- <sequence>
- <step>
  <order />
  <callName />
  </step>
  </sequence>
  </interface>
  </role>
  </roles>
  </informationalPart>
- <operationalPart>
  <trustMechanisms />
- <validity>
  <validFrom>1214750275</validFrom>
  <validUntil>1449550800</validUntil>
  </validity>
- <monitors>
- <monitor>
```

```
  <id>1</id>
  <localization>localhost:5050</localization>
  <type>synchronous</type>
  <initialization>--</initialization>
  </monitor>
  </monitors>
- <roles>
- <role>
  <roleName>clienteIProfile</roleName>
- <classAdaptors>
- <class>
  <classReference>SecurityMeasurement</classReference>
  <classRole>clienteIProfile</classRole>
  <adaptor />
  </class>
  </classAdaptors>
  </role>
  </roles>
  </operationalPart>
  </SandDPattern>
```

3 Implementación

- Y por ultimo tenemos la implementación.

```
  <?xml version="1.0" encoding="utf-8" ?>
-  <!--    Created  with  Liquid  XML  Studio  1.0.8.0  (http://www.liquid-
  technologies.com)
  -->
-     <SandDImplementation     xmlns:xsi="http://www.w3.org/2001/XMLSchema-
  instance"
  xsi:noNamespaceSchemaLocation="D:/SRFv2.2/serenity/schemas/S&DImplementa
  tion_v1.xsd"    name="IdentityProfileSMImplementation"    domain="tid.es"
  version="1.0">
- <informationalPart>
- <creator>
  <name>tid.es</name>
  <date>-2919</date>
  </creator>
  </informationalPart>
- <operationalPart>
- <SandDPatternReference>
  <name>IdentityProfileSecurityMeasurement</name>
  <domain>tid.es</domain>
  <version>1.0</version>
  <role>clienteIProfile</role>
  </SandDPatternReference>
- <implementationReference>
  <URL>file:IdentitySM\\IdentitySM.jar</URL>
  <type>jar</type>
  <signature>signedbyme</signature>
  </implementationReference>
  </operationalPart>
  </SandDImplementation>
```

PRESUPUESTO

1) Ejecución Material

- Compra de dos ordenadores portátiles ..... 2.500 €
- Material de oficina ..... 150 €
- Licencia de Software de Ekahau ..... 15.000 €
- Total de ejecución material ..... 5.650 €

2) Gastos generales

- 16 % sobre Ejecución Material ..... 904 €

3) Beneficio Industrial

- 6 % sobre Ejecución Material ..... 339 €

4) Honorarios Proyecto

- 960 horas a 30 € / hora ..... 28.800 €

5) Material fungible

- Encuadernación ..... 200 €

6) Subtotal del presupuesto

- Subtotal Presupuesto ..... 53.543 €

7) I.V.A. aplicable

- 16% Subtotal Presupuesto ..... 8.566,88 €

8) Total presupuesto

- Total Presupuesto ..... 62.109,88 €

Madrid, Junio de 2009

El Ingeniero Jefe de Proyecto

Fdo.: Jesús Marcos Morell
Ingeniero Superior de Telecomunicación

PLIEGO DE CONDICIONES

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un de una Aplicación de Control de Seguridad basada en el modelo SERENITY en un escenario de comunicaciones inalámbricas. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

Condiciones generales

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partida alzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción

provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrata" y anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

Condiciones particulares

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.

trial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.