UNIVERSIDAD
REY JUAN CARLOS

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN

GRADO EN INGENIERÍA EN TECNOLOGÍAS DE LA TELECOMUNICACIÓN

Curso Académico 2018/2019

Trabajo Fin de Grado

IMPLEMENTACIÓN DEL PROTOCOLO DE AUTORIZACIÓN OAUTH2

Autor : Pedro Tello Sánchez

Tutor : Pedro De Las Heras Quirós

Trabajo Fin de Grado

Implementación del Protocolo de Autorización oAUTH2

Autor : Pedro Tello Sánchez

Tutor : Pedro de las Heras Quirós

La defensa del presente Proyecto Fin de Carrera se realizó el día de de 2019, siendo calificada por el siguiente tribunal:

Presidente:

Secretario:

Vocal:

y habiendo obtenido la siguiente calificación:

Calificación:

Fuenlabrada, a de de 20XX

Dedicado a

mi familia / mi abuelo / mi abuela

II

Agradecimientos

Aquí vienen los agradecimientos... Aunque está bien acordarse de la pareja, no hay que olvidarse de dar las gracias a tu madre, que aunque a veces no lo parezca disfrutará tanto de tus logros como tú... Además, la pareja quizás no sea para siempre, pero tu madre sí.

Resumen

Aquí viene un resumen del proyecto. Ha de constar de tres o cuatro párrafos, donde se presente de manera clara y concisa de qué va el proyecto. Han de quedar respondidas las siguientes preguntas:

- ¿De qué va este proyecto? ¿Cuál es su objetivo principal?
- ¿Cómo se ha realizado? ¿Qué tecnologías están involucradas?
- ¿En qué contexto se ha realizado el proyecto? ¿Es un proyecto dentro de un marco general?

Lo mejor es escribir el resumen al final.

Summary

Here comes a translation of the “Resumen” into English. Please, double check it for correct grammar and spelling. As it is the translation of the “Resumen”, which is supposed to be written at the end, this as well should be filled out just before submitting.

Índice general

1. Introducción **1**

1.1. Historia 1

1.2. Diferencias entre Autorización y Autenticación 2

1.2.1. Autorización 2

1.2.2. Autenticación 3

1.3. Mecanismos de Autorización y Autenticación 4

1.3.1. SAML 4

1.3.2. OpenID Connect 5

1.3.3. oAuth 2.0 6

2. Objetivos **9**

2.1. Objetivo general 9

2.2. Objetivos específicos 9

2.3. Metodología 10

2.4. Planificación temporal 11

3. Aspectos del desarrollo **13**

3.1. Sección 1 13

4. Diseño e implementación **15**

4.1. Arquitectura general 15

5. Resultados **17**

6. Conclusiones **19**

6.1. Consecución de objetivos 19
6.2. Aplicación de lo aprendido 19
6.3. Lecciones aprendidas 19
6.4. Trabajos futuros 20

A. Manual de usuario **21**

Bibliografía **23**

Índice de figuras

1.1. Autorización VS Autenticación 2
1.2. OpenID Connect Protocol Suite 6

2.1. Esquema de las diferentes etapas 10
2.2. Diagrama de Gantt 11

4.1. Estructura del parser básico 16

Capítulo 1

Introducción

Este Trabajo Fin de Grado se desarrolla en el ámbito de la seguridad y la preservación de la confidencialidad en Internet. En este capítulo introduciremos los conceptos y tecnologías utilizadas a la vez que se presentará el esquema general de la custodia de la información.

1.1. Historia

Internet se ha convertido en el centro de nuestras vidas. Según un estudio realizado por la compañía Brandwatch [1] en Abril de 2018 la población mundial era de 7.8 mil millones de personas de los cuales 4.2 mil millones de personas eran usuarios de Internet y 3.03 mil millones eran usuarios de redes sociales y/o estaban registrados en páginas web.

Con tal magnitud de usuarios generando datos en poco tiempo, se llegó a la conclusión, de que la información de los usuarios contenida en diferentes sitios web, podía ser de valor para otras entidades y se comenzaron a crear servicios que permitían el envío de determinada información bajo autorización del propietario de la misma. Así fue como nacieron las API que ofrecían servicios y las Aplicaciones que podrían consumir los mismos.

El problema surgió cuando para acceder a la información de un determinado usuario era necesario suministrar las credenciales de acceso de dicho usuario. Esa información era crítica y bajo ningún concepto se podría suministrar a alguien que no fuera la API o el usuario. Por tanto se necesitaba conseguir que los usuarios, propietarios de los datos, pudieran disponer de un protocolo seguro mediante el cuál facilitar las credenciales únicamente a la API. Así se llegó

[1] `www.brandwatch.com`

a la conclusión de que era necesario encontrar métodos de autenticación que no pusieran en peligro la información personal del usuario a la vez que concedieran los permisos necesarios de autorización para acceder a la información específica que quisiera compartir el usuario.

A lo largo de los años han ido surgiendo diferentes tecnologías que iban proporcionando soluciones tanto al concepto de autorización como al concepto de autenticación.

1.2. Diferencias entre Autorización y Autenticación

Son dos conceptos se suelen confundir con bastante frecuencia. Mientras que la autenticación se enfoca en determinar que el usuario es quien dice ser, la autorización se encarga de controlar qué acciones ese usuario puede realizar.

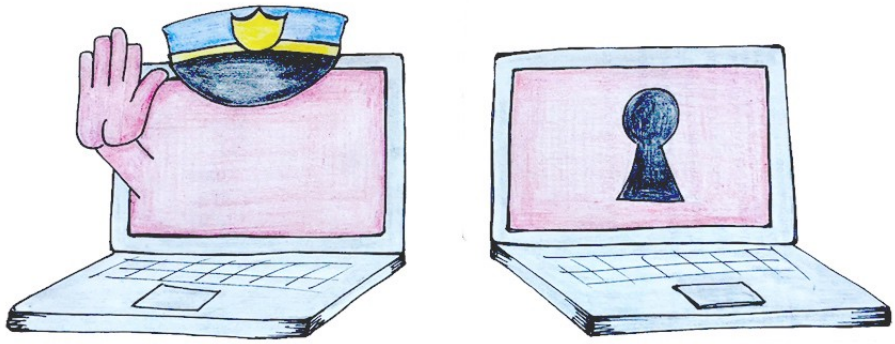

Figura 1.1: Autorización VS Autenticación

1.2.1. Autorización

La autorización se encarga de determinar si tenemos autoridad para hacer algo. Pensemos por ejemplo en las aplicaciones móviles en las que existe una versión Lite (gratuita) y una

versión Pro (de pago). Un usuario gratuito está autorizado a acceder a un conjunto limitado de funcionalidades y no está autorizado a otras funcionalidades, mientras no pague la suscripción y se convierta en usuario Premium.

Cada vez que un usuario intenta acceder a una funcionalidad, la aplicación realiza un control para determinar si puede hacerlo o no. En segundo plano, lo que está ocurriendo es un proceso de autorización. Siempre que el usuario acceda a recursos para los que tiene permisos, no habrá problema. Sin embargo, en cuanto el usuario intente acceder a un recurso que solo está disponible para la versión Pro, se le denegará el permiso y comunmente se le invita a unirse al paquete Premium.

1.2.2. Autenticación

La autenticación se define como el proceso mediante el cual se verifica que un individuo, es quien dice ser. Un ejemplo sería, por ejemplo, el proceso de autenticación que realiza Google. Cuando se inicia sesión introduciendo un correo electrónico y contraseña estamos autenticándonos. Google está comprobando que la contraseña introducida coincide con la contraseña que se asoció a mi correo electrónico en el momento del registro. Si la contraseña coincide con el correo podremos acceder a la aplicación, sin embargo, si la contraseña no coincide con el correo, Google no permitirá el inicio de sesión porque no se puede asegurar que la persona sea quien dice ser.

Debido a que la posibilidad de que nos roben la contraseña existe, en los últimos años ha surgido la autenticación de dos factores. Dicha autenticación consiste en agregar un segundo paso para verificar que la persona es quien dice ser. Entonces, no solo se necesitaría la contraseña sino que se requeriría un paso adicional. La manera más común de realizar dicha autenticación consiste en enviar un mensaje al correo electrónico desde el cual se está intentando acceder o mediante el envío de un código al teléfono móvil de contacto.

Autenticación mediante Certificados Electrónicos

A partir de los avances tecnológicos actuales podemos predecir que las contraseñas ya no son un método confiable de autenticación de usuario. Este problema, combinado con la creciente amenaza de máquinas malintencionadas, hace que muchos expertos en TI se pregunten como

pueden asegurarse de que solo los usuarios y dispositivos aprobados tengan acceso. La solución a este problema lo podemos encontrar en los certificados electrónicos.

La autenticación basada en certificados es el nombre que se le da a utilizar un Certificado Digital para identificar a un usuario, máquina o dispositivo antes de otorgar acceso a un recurso, red, aplicación, etc. En el caso de la autenticación de usuario, a menudo se implementa en conjunto con métodos tradicionales como nombre de usuario y contraseña. A diferencia de algunas soluciones que solo funcionan para usuarios, como la biométrica y las contraseñas de un solo uso, la autenticación mediante certificado se puede utilizar para cualquier gestión final: usuarios, máquinas, dispositivos e incluso para gestionar la tecnología creciente de Internet of Things.

La autenticación basada en certificados hace uso de la infraestructura de clave pública (de sus siglas en inglés PKI) para generar un par de claves (una pública y una privada). La privada es personal e intransferible y la pública de libre acceso. De esta forma cualquier intento de acceso por parte del usuario se realizará utilizando la clave privada (firmando el mensaje) para que posteriormente la plataforma destino mediante un mecanismo de validación, haciendo uso de la clave pública de esta entidad, pueda descifrar dicho mensaje. En caso de poder descifrarlo correctamente se podrá afirmar con total certeza que el usuario es quien dice ser y se realizará correctamente la autenticación.

1.3. Mecanismos de Autorización y Autenticación

A continuación se van a exponer dos mecanismos de autorización y autenticación conjunta para finalizar con el mecanismo de autorización en el que vamos a basar este TFG.

1.3.1. SAML

Security Assertion Markup Language (SAML) es un estándar abierto que funciona transfiriendo la identidad del usuario desde un proveedor de identidad al proveedor del servicio. Esto se produce mediante el intercambio de documentos XML firmados digitalmente.

Un ejemplo de uso sería en el que un usuario inicia sesión en un sistema que actúa como proveedor de identidad. El usuario desea iniciar sesión en una aplicación (el proveedor de servicios). Los pasos serían los siguientes:

1. El usuario accede a la aplicación remota mediante un enlace en una intranet, un marcador o similar y se carga la aplicación.

2. La aplicación identifica el origen del usuario (por subdominio de la aplicación, dirección IP del usuario o similar) y redirige al usuario al proveedor de identidad, solicitando la autenticación. Esta es la solicitud de autenticación.

3. El usuario tiene una sesión de navegador activa existente con el proveedor de identidad o establece una, iniciando sesión en el proveedor de identidad.

4. El proveedor de identidad construye la respuesta de autenticación en forma de un documento XML que contiene el nombre de usuario o la dirección de correo electrónico del usuario, lo firma con un certificado X.509 y publica esta información al proveedor de servicios.

5. El proveedor de servicios, que ya conoce al proveedor de identidad y tiene una huella digital de certificado, recupera la respuesta de autenticación y la valida utilizando la huella digital del certificado.

6. Se establece la identidad del usuario y se le proporciona acceso a la aplicación.

SAML 2.0

Es una versión del estándar SAML para tramitar el intercambio de datos de autenticación y autorización entre dominios de seguridad. Es un protocolo basado en XML que utiliza tokens de seguridad que contienen aserciones para pasar información sobre un principal (generalmente un usuario final) entre un Proveedor de Identidad, y un Proveedor de Servicios.

SAML 2.0 fue ratificado como un estándar OASIS en marzo de 2005, reemplazando a SAML 1.1.

1.3.2. OpenID Connect

Es un protocolo de autenticación implementada utilizando oAuth 2.0, un framework de autorización. El estándar está controlado por el OpenID Foundation. Permite a los Clientes verificar la identidad del Usuario Final basándose en la autenticación realizada por un Servidor

de Autorización, así como obtener información de perfil básica sobre el Usuario final de manera interoperable y similar a REST.

OpenID Connect permite a los clientes de todo tipo, incluidos los clientes basados en web, móviles y de JavaScript, solicitar y recibir información sobre sesiones autenticadas y usuarios finales. El conjunto de especificaciones es extensible, lo que permite a los participantes utilizar funciones opcionales, como el cifrado dedatos de identidad, el descubrimiento de proveedores de OpenID y la administración de sesiones, cuando sea conveniente para ellos.

Las especificaciones de OpenID Connect y las especificaciones en las que se basan se muestran en el diagrama a continuación:

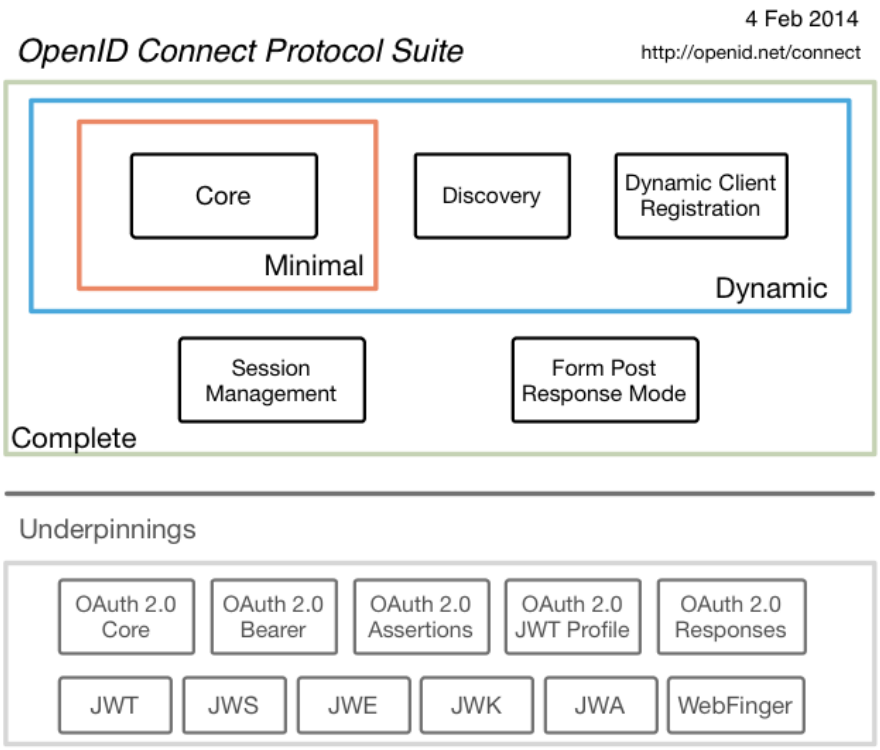

Figura 1.2: OpenID Connect Protocol Suite

1.3.3. oAuth 2.0

El estandar OAuth 2.0 (RFC 6749) es un framework de autorización que permite controlar el acceso por parte de las aplicaciones a los datos de los usuarios sin tener que proporcionar las credenciales. Según el estandar, el marco de autorización de OAuth 2.0 permite que una aplicación de terceros obtenga acceso limitado a un servicio HTTP, ya sea en nombre de un

propietario de recursos mediante la organización de una interacción de aprobación entre el propietario del recurso y el servicio HTTP, o permitiendo a la aplicación de terceros obtener acceso en su propio nombre.

Capítulo 2

Objetivos

2.1. Objetivo general

Este trabajo fin de grado ha consistido en crear los mecanismos necesarios para la concesión de Autorización entre una Aplicación y una API a través de una implementación del estandar oAuth 2.0.

2.2. Objetivos específicos

Los objetivos específicos del proyecto son los siguientes:

1. Elaborar una propuesta de tecnología, arquitectura y límites del proyecto: Dicha propuesta respondería a la finalidad última del proyecto y pondría límites a los trabajos que se llevaría a cabo más adelante.

2. En base a la arquitectura realizar un estudio e implementar el esquema de base de datos a utilizar: Las conclusiones obtenidas serán en base a las necesidades iniciales del primer estudio y teniendo en cuenta en la medida de lo posible mejoras futuras.

3. Creación de la capa de front de la Aplicación: Esta web deberá disponer de las características necesarias para que se puedan realizar las pruebas necesaria para el testeo de la aplicación. Se tendrán en cuenta características como establecer una interfaz amigable y responsive.

4. Administración del Registro y Login de usuarios en la Aplicación: Creación de la funcionalidad específica para la correcta acción de registro y logueo de usuarios en la aplicación, teniendo en cuenta diversas problemáticas que se puedan dar e intentar anticiparse, así como realizar el mejor control de errores posible.

5. Creación del proceso de Autorización utilizando el estandar oAuth 2.0: Desarrollo de la funcionalidad específica del estándar oAuth 2.0 basándonos en la documentación establecida en la RFC 6749.

6. Creación del front básico de gestión de usuarios de la API: Dicha web deberá disponer de una interfaz simple que permita realizar la autenticación de un usuario en la API.

2.3. Metodología

Este Trabajo Fin de Grado se ha realizado utilizando una metodología de realimentación o feedback basada en la metodología ágil SCRUM. Se basa en que el producto final es el resultado de su propia evolución como consecuencia de las mejoras y problemas que se han ido identificando durante la creación del mismo. En el siguiente esquema podemos observar las diferentes etapas:

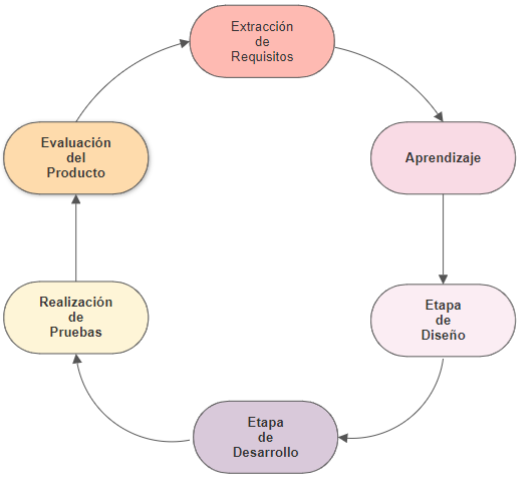

Figura 2.1: Esquema de las diferentes etapas

2.4. Planificación temporal

La duración de la realización del proyecto ha sido entorno a los 6.5 meses. A continuación se adjunta un diagrama de Gantt:

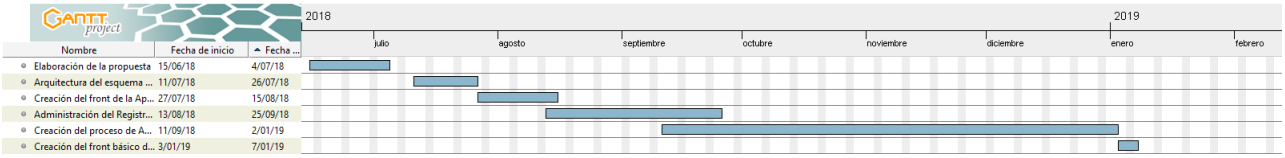

Figura 2.2: Diagrama de Gantt

Desde el 15 de Junio hasta el 7 de Enero la estimación de horas que he necesitado para la finalización del proyecto ha sido entorno a las 400 horas. De ese total de horas la mayoría han sido en las horas posteriores a la vuelta de la jornada laboral y durante los fines de semana, siendo respectivamente la media de horas de 1 a 2 y de 4 a 14.

Capítulo 3

Aspectos del desarrollo

3.1. Sección 1

Hemos hablado de cómo incluir figuras. Pero no hemos dicho nada de tablas. A mí me gustan las tablas. Mucho. Aquí un ejemplo de tabla, la Tabla 3.1.

1	2	3
4	5	6
7	8	9

Cuadro 3.1: Ejemplo de tabla

Capítulo 4

Diseño e implementación

Aquí viene todo lo que has hecho tú (tecnológicamente). Puedes entrar hasta el detalle. Es la parte más importante de la memoria, porque describe lo que has hecho tú. Eso sí, normalmente aconsejo no poner código, sino diagramas.

4.1. Arquitectura general

Si tu proyecto es un software, siempre es bueno poner la arquitectura (que es cómo se estructura tu programa a "vista de pájaro").

Por ejemplo, puedes verlo en la figura 4.1.

Si utilizas una base de datos, no te olvides de incluir también un diagrama de entidad-relación.

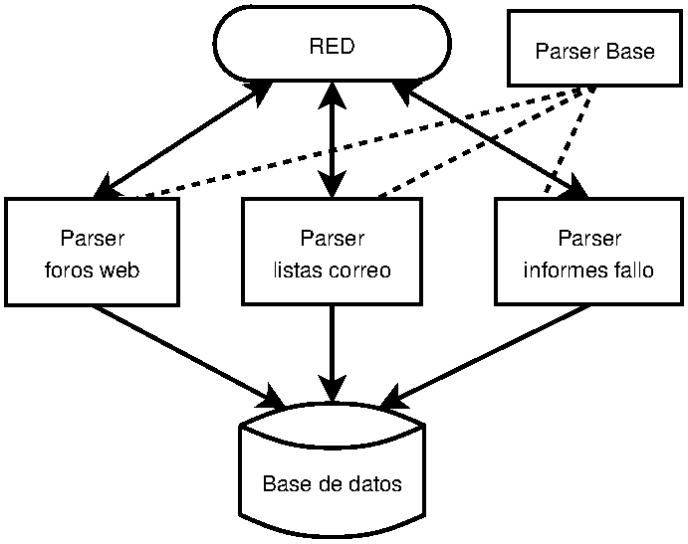

Figura 4.1: Estructura del parser básico

Capítulo 5

Resultados

En este capítulo se incluyen los resultados de tu trabajo fin de grado.

Si es una herramienta de análisis lo que has realizado, aquí puedes poner ejemplos de haberla utilizado para que se vea su utilidad.

Capítulo 6

Conclusiones

6.1. Consecución de objetivos

Esta sección es la sección espejo de las dos primeras del capítulo de objetivos, donde se planteaba el objetivo general y se elaboraban los específicos.

Es aquí donde hay que debatir qué se ha conseguido y qué no. Cuando algo no se ha conseguido, se ha de justificar, en términos de qué problemas se han encontrado y qué medidas se han tomado para mitigar esos problemas.

6.2. Aplicación de lo aprendido

Aquí viene lo que has aprendido durante el Grado/Máster y que has aplicado en el TFG/TFM. Una buena idea es poner las asignaturas más relacionadas y comentar en un párrafo los conocimientos y habilidades puestos en práctica.

1. a
2. b

6.3. Lecciones aprendidas

Aquí viene lo que has aprendido en el Trabajo Fin de Grado/Máster.

1. a

2. b

6.4. Trabajos futuros

Ningún software se termina, así que aquí vienen ideas y funcionalidades que estaría bien tener implementadas en el futuro.

Es un apartado que sirve para dar ideas de cara a futuros TFGs/TFMs.

Apéndice A

Manual de usuario

Bibliografía

[1] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Articial Systems*. Oxford University Press, Inc., 1999.