Universidad de Alcalá
Escuela Politécnica Superior

Grado en Ingeniería Informática

Trabajo Fin de Grado

ESCUELA POLITÉCNICA SUPERIOR

Desarrollo de una aplicación para realizar búsquedas de archivos por hash en la red de Ares

Autor: Jesús Domínguez Belinchón

Tutor: José Javier Martínez Herráiz

2013

UNIVERSIDAD DE ALCALÁ

Escuela Politécnica Superior

Grado en Ingeniería Informática

Trabajo Fin de Grado

Desarrollo de una aplicación para realizar búsquedas de archivos por hash en la red de Ares

Autor: Jesús Domínguez Belinchón

Director: José Javier Martínez Herráiz

TRIBUNAL:

Presidente:

Vocal 1:

Vocal 2:

CALIFICACIÓN: ……

FECHA: ……

A mis padres, sin su cariño y su ayuda nada de esto habría sido posible.

Tan sólo has de creer que eres capaz de hacer algo y tendrás las fuerzas suficientes para conseguirlo.

Agradecimientos

A mis padres, por darme los medios para conseguir mis metas, por apoyarme en cada decisión que he tomado y por enseñarme lo que es la vida.

A mi familia, a los que están y a los que ya no están conmigo, por creer en mí y animarme en este camino.

A José Javier Martínez, por confiar en mí y guiarme en este proyecto.

A Manuel Sánchez Rubio, por las valiosísimas lecciones que me ha dado y que no se aprenden en los libros.

A mi novia, Beatriz, por regalarme su sonrisa cada día y recorrer conmigo el camino de la vida.

A José Luis, Daniel y Sara, por acompañarme durante todos estos años, por las horas de estudio juntos, por sacarme una sonrisa cada día y por compartir conmigo recuerdos inolvidables.

Resumen

En la actualidad, las redes P2P (Peer-to-peer) son una de las principales formas de compartición de archivos entre usuarios que se utilizan en internet. De esta forma, los usuarios comparten con el resto del mundo archivos alojados en su ordenador.

Muchas aplicaciones como Emule, BitTorrent o Ares son utilizadas a diario por millones de usuarios en todo el mundo para compartir y descargar millones de archivos.

El objetivo de este TFG es buscar archivos en la red de Ares a través de su hash y registrar en una base de datos a los usuarios que tengan dichos archivos.

Abstract

Currently, P2P networks (Peer-to-peer) are the main way to share files between internet users. These networks allow users to share files stored in your computer with the rest of the world

Many applications like eMule, BitTorrent and Ares are daily used by millions of users around the world to share and download millions of files.

The purpose of this TFG is to search files in the Ares' network through its hash identifier. Users with these files will be stored in a external database.

Palabras clave / Keywords:

P2P, Ares, Hash, XAMPP, Delphi

Índice General

Introducción **18**
1.1 Resumen 18
1.2 Objetivos 20
1.3 Conceptos Previos 21
1.3.1 Redes P2P 21
1.3.2 Clasificación de las redes P2P 22
1.3.3 Diferencias entre arquitecturas P2P y Cliente-servidor 24
1.3.4 Hash 26
1.3.5 Servidor web 27
1.3.6 Ares 28

Descripción del sistema **29**
2.1 Arquitectura de la red de Ares 29
2.2 Identificación de usuarios 31
2.3 Identificación de archivos 31
2.4 Búsqueda de archivos y fuentes 32
2.5 Compartición de archivos 33
2.6 Tecnologías y lenguajes utilizados 35
2.6.1 XAMPP 35
2.6.2 Borland Delphi 7 39
2.6.3 Delphi 39
2.6.4 SQL 41
2.6.5 HTML 42
2.6.6 Adobe Photoshop CS5 42
2.6.7 Wireshark 43

Trabajo realizado **46**
3.1 Servidor 46
3.2 Análisis de tramas 46
3.3 Diseño y creación de la base de datos 52
3.4 Ares 54
3.4.1 Compilación 54
3.4.2 Modificaciones realizadas 64
3.4.3 Diagrama de comunicación entre Ares y el servidor web 66
3.5 Servicio web 67
3.5.1 Control de acceso 68
3.5.2 Pestaña 1: Crear base de datos 69
3.5.3 Pestaña 2: Subir fichero 70
3.5.4 Pestaña 3: Descarga Base_datos.txt 71
3.5.5 Pestaña 4: Mostrar resultados 72

Manual de usuario **75**
4.1 Instalación del servidor 75
4.2 Pasos previos 76
4.2.1 Inicialización del servidor Apache y de MySQL 76
4.2.2 Gestión de usuarios 78
4.2.3 Interfaz web 82
4.2.4 Creación de la base de datos 84
4.2.5 Subida de archivos a la base de datos 86
4.2.6 Descarga del fichero de configuración 87
4.3 Ares modificado 88
4.4 Gestión del fichero de hashes 92
4.5 Gestión de resultados 92
4.6 Preguntas más frecuentes 96
4.7 Glosario 98

Requisitos del sistema y presupuesto 101
5.1 Requisitos del sistema 101
5.1.1 Hardware 101
5.1.2 Software 101
5.2 Presupuesto 102

Conclusiones 104

Bibliografía 106

Índice de Figuras

Figura 1: Topologías de redes P2P 24
Figura 2: Ejemplos de topología Cliente-servidor y P2P 25
Figura 3: Interfaz de Ares 2.1.8 28
Figura 4: Localización de la carpeta de descargas de Ares 34
Figura 5: Principales componentes de XAMPP 35
Figura 6: Panel de control de XAMPP 36
Figura 7: Interfaz de phpMyAdmin 39
Figura 8: Interfaz de Adobe Photoshop CS5 43
Figura 9: Interfaz de Wireshark 44
Figura 10: Captura de Wireshark de la IP de usuario en Hexadecimal 45
Figura 11: Hash insertado en el código 47
Figura 12: Información del archivo a descargar 47
Figura 13: Captura de tramas al realizar petición 48
Figura 14: Detalle del paquete enviado 48
Figura 15: Captura del fichero SNodes.dat 49
Figura 16: Captura de respuestas de los nodos 50
Figura 17: Detalle de descarga en Cliente de Ares modificado 50
Figura 18: IP en hexadecimal del usuario 51
Figura 19: Conversión de la IP en Hexadecimal 52
Figura 20: Esquema de la base de datos "ares" 54
Figura 21: Captura componentes ActiveX 56
Figura 22: Vista de herramientas administrativas 57
Figura 23: Ventana configuración administrador ODBC 57
Figura 24: Vista con los valores introducidos 58
Figura 25: Ventana de directorios del proyecto 59
Figura 26: Captura del proceso de instalación de JCL 59
Figura 27: Ventana emergente JCL 60
Figura 28: Captura del proceso de instalación de JVCL 60
Figura 29: Localización del menú Component 61
Figura 30: Ventana de importación de componente ActiveX 62
Figura 31: Rutas en Library Path 63
Figura 32: Rutas en Browsing Path 63
Figura 33: Diagrama de comunicación entre Ares y el servidor web 67
Figura 34: Interfaz inicial del servicio web 68

Figura 35: Formulario de acceso 69
Figura 36: Interfaz de la pestaña "Crear base de datos" 70
Figura 37: Interfaz de la pestaña "Subir fichero" 71
Figura 38: Formato del archivo Excel de hashes 71
Figura 39: Interfaz de la pestaña "Descargar Base_datos.txt" 72
Figura 40: Formato del archivo "Base_datos.txt" 72
Figura 41: Interfaz de la pestaña "Mostrar Resultados" 73
Figura 42: Vista Web de la información 73
Figura 43: Pantalla de instalación de XAMPP 76
Figura 44: Panel de control de XAMPP 77
Figura 45: Panel de control de XAMPP con Apache y MySQL funcionando 77
Figura 46: Interfaz de selección de idioma de XAMPP 78
Figura 47: Interfaz de phpMyAdmin 79
Figura 48: Vista de la pestaña "Privilegios" de phpMyAdmin 79
Figura 49: Captura con los datos del usuario administrador introducidos 80
Figura 50: Captura con los privilegios de usuario seleccionados 80
Figura 51: Captura de los comandos introducidos en el CMD 82
Figura 52: Vista del directorio htdocs con el servicio web 83
Figura 53: Interfaz principal del servicio web 83
Figura 54: Formulario de acceso con las credenciales introducidas 84
Figura 55: Introducción del valor 1 a la variable "*maxfiles*" 84
Figura 56: Ventana de confirmación de la base de datos 85
Figura 57: Formulario de acceso a phpMyAdmin 85
Figura 58: Bases de datos de phpMyAdmin 85
Figura 59: Vista de las tablas de la base de datos "ares" desde phpMyAdmin 86
Figura 60: Captura previa a la subida del archivo Excel 86
Figura 61: Ventana de confirmación de carga de hashes en la base de datos 87
Figura 62: Icono de descarga 87
Figura 63: Ventana con el fichero de configuración descargado 87
Figura 64: Directorio con el ejecutable y el fichero de configuración 88
Figura 65: Estado de la pestaña "Transfer" al iniciar Ares 89
Figura 66: Pestaña "Transfer" con los hashes descargados 89
Figura 67: IPs de los usuarios que tienen el archivo 90
Figura 68: Pestaña "Control Panel" de Ares 91
Figura 69: Estado de los archivos buscados 91
Figura 70: Pestaña "Mostrar resultados" del servicio web 92
Figura 71: Vista web del informe general 93
Figura 72: Ventana de selección de país 93
Figura 73: Aspecto de "Informe_pais.pdf" 94

Figura 74: Ventana para introducir el hash 94
Figura 75: Aspecto de "Informe_hash.pdf" 95
Figura 76: Ventana para introducir el nombre de usuario 95
Figura 77: Aspecto de "Informe_username.pdf" 96
Figura 78: Localización de SCM en el panel de control de XAMPP 97
Figura 79: Localización del botón "Eliminar" 98

Índice de Tablas

Tabla 1: Diferencias entre arquitectura P2P y arquitectura Cliente-servidor 25
Tabla 2: Datos de conexión 58
Tabla 3: Credenciales de la cuenta de administrador 80
Tabla 4: Sustituciones de código en el fichero Config.inc 81
Tabla 5: Valor de la variable "max_execution_time" 81
Tabla 6: Cambios en php.ini 96
Tabla 7: Salario base de un Ingeniero al día 102
Tabla 8: Salario Total del trabajador 102
Tabla 9: Coste del equipo Informático 103
Tabla 10: Coste de conexión a Internet 103
Tabla 11: Coste licencia de programas 103
Tabla 12: Presupuesto total 103

Capítulo 1

Introducción

1.1 Resumen

En la actualidad, las ***redes P2P*** (Peer-to-peer) son una de las principales formas de compartición de archivos entre usuarios que se utilizan en internet.

Muchas aplicaciones como Emule, Skype, BitTorrent o Ares son utilizadas a diario por millones de usuarios en todo el mundo. Pese a que todas estas aplicaciones son P2P, cada una de ellas utiliza una arquitectura y unos protocolos distintos.

La primera aplicación P2P fue *Hotline Connect*, desarrollada en 1996. Después de esta, aparecieron muchas de las que conocemos en la actualidad como **Emule**, originariamente *eDonkey 2000* en el año 2000 y **Ares** en 2002, inspirada en la arquitectura de red P2P Gnutella.

Ares utiliza la red ***Ares Galaxy***, una red P2P pura en la que la conexión se produce directamente entre el usuario que solicita el archivo y el usuario que lo tiene. En esta Red todos los nodos (usuarios) son a su vez clientes y servidores, y no existe un servidor central que maneje las conexiones de red.

La finalidad de esta red es permitir a los usuarios compartir los archivos alojados en su ordenador con el resto de usuarios, ya sean archivos de música, imágenes, videos u otro tipo de documentos.

Un dato importante es que cada archivo tiene asociado un identificador **Hash**, que es un código que identifica de forma univoca a un archivo. El concepto de Hash se explicara más detalladamente en el apartado 1.3.4.

Este TFG pretende modificar un cliente de Ares para realizar búsquedas de archivos por hash en la red Ares Galaxy. El cliente esta creado en lenguaje Delphi, por lo que se utilizara dicho lenguaje de programación para su modificación.

Para poder realizar la búsqueda de archivos por hash, se va a proceder a modificar los tipos de búsqueda que realiza el cliente de Ares y se añadirá esta forma. De esta manera, cuando se solicite un archivo en la red, dicho archivo será buscado en

función al valor de su hash, lo que nos garantiza que vamos a encontrar el archivo que estamos buscando y no habrá equivocaciones con otros archivos.

Muchas veces, en las búsquedas por nombre, se descargan archivos que no son los deseados debido a que algún usuario de la red ha modificado o cambiado el nombre y se lo ha asignado de forma premeditada o equivocada a un archivo. Realizando búsquedas por hash no se producen este tipo de equivocaciones, ya que es un identificador único de cada archivo, como el DNI de una persona.

La información recopilada durante la búsqueda del cliente de Ares será, entre otras, el nombre de usuario, la IP del usuario y el país. Esta información se almacenará en una base de datos y se podrá visualizar desde un servicio web, creado como complemento en este TFG.

El servicio web constara de una interfaz creada con PHP y HTML desde la cual podremos acceder a los resultados obtenidos en las búsquedas del cliente de Ares, así como a otras funcionalidades orientadas a la administración de la base de datos.

El TFG está dividido en varios capítulos. Un primer capítulo introductorio, en el cual se explicara el propósito del TFG, los objetivos y nos familiarizaremos con los términos y conceptos más importantes del sistema creado.

En el segundo capítulo se describe el sistema y las tecnologías utilizadas en este TFG, lo que nos permitirá entender cómo funciona la red de Ares, como es su arquitectura, cómo se realizan las búsquedas de archivos, cómo se identifican los usuarios y cómo se comparten los archivos.

En el tercer capítulo se explicaran las modificaciones que se han realizado sobre el cliente de Ares, el proceso de creación del servicio web y la estructura de la base de datos.

En el cuarto capítulo hay un manual de usuario, que contiene una guía de instalación de los componentes necesarios para el funcionamiento del sistema, los pasos previos a realizar antes de poner en marcha el cliente de Ares Modificado y la explicación detallada de las funcionalidades que nos ofrece el servicio web.

Y, finalmente, en el quinto capítulo se describirán los requisitos mínimos, tanto hardware como software, necesarios para el correcto funcionamiento del sistema y el presupuesto estimado para llevar a cabo la realización del TFG.

1.2 Objetivos

El objetivo principal de este TFG es explicar detalladamente el funcionamiento de la red P2P de Ares y modificar un cliente de Ares para que se puedan realizar búsquedas de archivos de forma univoca a través de su identificador HASH.

De esta forma, se podrían detectar e identificar a los usuarios de la red P2P de Ares que tengan en su posesión archivos (fotos, videos, etc.) de un determinado tipo.

Con el fin de alcanzar el objetivo general expuesto anteriormente, se fueron marcando objetivos a lo largo del desarrollo que unidos conforman el principal. Éstos son los que se exponen a continuación:

- Gestionar un servidor de base de datos y web, conformando un sistema centralizado al cual se conectará la aplicación tanto para registrar información como para leerla de la misma.

- Modificar el código fuente del software Ares 2.1.8 para que se conecte al servidor de base de datos (tanto para leer información como para registrar datos) de forma automatizada.

- Clasificar la información de la base de datos mediante una aplicación web desde la cual se ordena la información registrada de una forma clara y sencilla de tratar. De esta forma se consigue diferenciar entre usuarios con pocos archivos registrados, o aquellos que tienen un volumen considerado de ficheros de pornografía infantil, y por lo tanto es poco probable estar ante un falso positivo. Además desde dicha interfaz web se realizan tareas de gestión tales como añadir nuevos archivos a la base de datos o crear la misma.

- Realizar todo el sistema en un entorno seguro y privado, en el que no puedan acceder usuarios no autorizados a la base de datos, aplicación o información de la interfaz web.

- Realizar un entorno de fácil manejo.

En cuanto a sus posibles utilidades, este TFG puede ser utilizado para futuras investigaciones sobre la red de Ares y para la posible identificación de usuarios que tengan un determinado archivo que estamos buscando.

1.3 Conceptos Previos

1.3.1 Redes P2P

Una red informática P2P se refiere a una red que no tiene clientes ni servidores fijos, sino una serie de nodos que se comportan simultáneamente como clientes y como servidores respecto de los demás nodos de la red. Este modelo de red contrasta con el modelo cliente-servidor, el cual se rige mediante una arquitectura monolítica donde hay una simple comunicación entre un usuario y una terminal, en la que el cliente y el servidor no pueden cambiar de roles.

Las redes de ordenadores P2P son redes que aprovechan, administran y optimizan el uso de banda ancha que acumulan de los demás usuarios en una red por medio de la conectividad entre los mismos usuarios participantes de la red, obteniendo como resultado mucho más rendimiento en las conexiones y transferencias que con algunos métodos centralizados convencionales, donde una cantidad relativamente pequeña de servidores provee el total de banda ancha y recursos compartidos para un servicio o aplicación. Típicamente, estas redes se conectan en gran parte con otros nodos vía "ad-hoc".

Dichas redes son útiles para muchos propósitos, pero se usan muy a menudo para compartir toda clase de archivos que contienen: audio, video, texto, software y datos en cualquier formato digital.

Cualquier nodo puede iniciar, detener o completar una transacción compatible. La eficacia de los nodos en el enlace y transmisión de datos puede variar según su configuración local (firewalls, NAT, routers, etc.), velocidad de proceso, disponibilidad de ancho de banda de su conexión a la red y capacidad de almacenamiento en disco.

En una red P2P se buscan las siguientes características:

- **Escalabilidad**: Se desea que cuantos más nodos estén conectados en una red P2P mejor sea su funcionamiento. Los recursos de la red van aumentando y a diferencia de una arquitectura cliente-servidor con un sistema fijo de servidores, nunca se llegará a una tasa de transferencia de datos lenta por la sobrecarga a servidores fijos.

- **Robustez**: La naturaleza distribuida de las redes P2P también incrementa la robustez en caso de haber fallos en la réplica excesiva de los datos hacia múltiples destinos, y en sistemas P2P puros permitiendo a los nodos encontrar la información sin hacer peticiones a ningún servidor centralizado de indexado.

- **Descentralización**: Estas redes por definición son descentralizadas y todos los nodos son iguales. No existen nodos con funciones especiales, y por tanto ningún nodo es imprescindible para el funcionamiento de la red.

- **Anonimato**: Es deseable que en estas redes quede anónimo el autor de un contenido, el editor, el lector, el servidor que lo alberga y la petición para encontrarlo siempre que así lo necesiten los usuarios.

- **Seguridad**: Es una de las características deseables de las redes P2P menos implementada. Los objetivos de un P2P seguro serían identificar y evitar los nodos maliciosos, evitar el contenido infectado, evitar el espionaje de las comunicaciones entre nodos, creación de grupos seguros de nodos dentro de la red, protección de los recursos de la red... En su mayoría aún están bajo investigación, pero los mecanismos más prometedores son: cifrado multiclave, cajas de arena, gestión de derechos de autor, reputación, comunicaciones seguras, comentarios sobre los ficheros...

1.3.2 Clasificación de las redes P2P

Dependiendo de su arquitectura, las redes P2P se pueden clasificar en tres grupos:

Redes P2P Centralizadas

Este tipo de red P2P se basa en una arquitectura monolítica en la que todas las transacciones se hacen a través de un único servidor que sirve de punto de enlace entre dos nodos y que, a la vez, almacena y distribuye los nodos donde se almacenan los contenidos. Poseen una administración muy dinámica y una disposición más permanente de contenido. Sin embargo, está muy limitada en la privacidad de los usuarios y en la falta de escalabilidad de un sólo servidor, además de ofrecer problemas en puntos únicos de fallo, situaciones legales y enormes costos en el mantenimiento así como el consumo de ancho de banda.

Una red de este tipo reúne las siguientes características:

- Se rige bajo un único servidor que sirve como punto de enlace entre nodos y como servidor de acceso al contenido, el cual distribuye a petición de los nodos.

- Todas las comunicaciones (como las peticiones y encaminamientos entre nodos) dependen exclusivamente de la existencia del servidor.

Redes P2P Puras o Totalmente Descentralizadas

Las redes P2P de este tipo son las más comunes, siendo las más versátiles al no requerir una gestión central de ningún tipo, lo que permite una reducción de la necesidad de usar un servidor central, por lo que se opta por los mismos usuarios como nodos de esas conexiones y también como almacenadores de información. En otras palabras, todas las comunicaciones son directamente de usuario a usuario con ayuda de un nodo (que es otro usuario) quien permite enlazar esas comunicaciones.

Las redes de este tipo tienen las siguientes características:

- Los nodos actúan como cliente y servidor.
- No existe un servidor central que maneje las conexiones de red.
- No hay un enrutador central que sirva como nodo y administre direcciones.

Algunos ejemplos de una red P2P "pura" son: Ares Galaxy (Ares), red Kad (*Kademlia*, usada por Emule), BitTorrent, Gnutella, Freenet y Gnutella2.

Redes P2P Hibridas, semicentralizadas o mixtas

En este tipo de red, se puede observar la interacción entre un servidor central que sirve como hub y administra los recursos de banda ancha, enrutamientos y comunicación entre nodos pero sin saber la identidad de cada nodo y sin almacenar información alguna, por lo que el servidor no comparte archivos de ningún tipo a ningún nodo. Puede incorporar más de un servidor que gestione los recursos compartidos, pero también en caso de que el o los servidores que gestionan todo caigan, el grupo de nodos sigue en contacto a través de una conexión directa entre ellos mismos con lo que es posible seguir compartiendo y descargando más información en ausencia de los servidores.

Los nodos son responsables de hospedar la información (pues el servidor central no almacena la información), que permite al servidor central reconocer los recursos que se desean compartir, y para poder descargar esos recursos compartidos a los peers que lo solicitan.

Las terminales de enrutamiento son direcciones usadas por el servidor, que son administradas por un sistema de índices para obtener una dirección absoluta.

Algunos ejemplos de una red P2P híbrida ED2K (Emule) y Direct Connect.

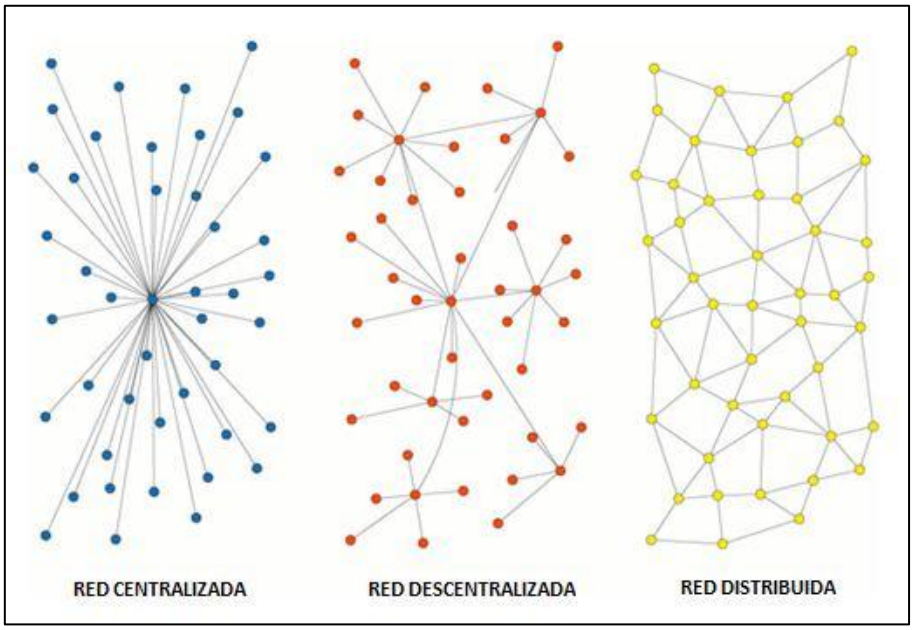

Figura 1: Topologías de redes P2P

1.3.3 Diferencias entre arquitecturas P2P y Cliente-servidor

La arquitectura cliente-servidor es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados **servidores**, y los demandantes, llamados **clientes**. Un cliente realiza peticiones al servidor, quien le da respuesta.

La red cliente-servidor es aquella red de comunicaciones en la que todos los clientes están conectados a un servidor, en el que se centralizan los diversos recursos y aplicaciones con que se cuenta; y que los pone a disposición de los clientes cada vez que estos son solicitados.

Esto significa que todas las gestiones que se realizan se concentran en el servidor, de manera que en él se disponen los requerimientos provenientes de los clientes que tienen prioridad, los archivos que son de uso público y los que son de uso restringido, los archivos que son de sólo lectura y los que, por el contrario, pueden ser modificados, etc.

Por el contrario, la arquitectura P2P, explicada en el apartado 1.3.1, permite el intercambio directo de información entre nodos, realizando cada nodo funciones de cliente y servidor.

La arquitectura P2P conecta un inmenso número de ordenadores de forma aleatoria, y se apoya principalmente en la potencia y ancho de banda de cada uno de los nodos.

A continuación, podemos ver las principales diferencias entre las arquitecturas P2P y Cliente servidor:

Arquitectura P2P	Arquitectura Cliente - Servidor
Funcionan sin clientes-servidores: cualquier nodo o usuario puede ser un cliente o un servidor a la vez.	Requiere un servidor esclavo y un cliente.
Conectado a través de un red.	Conectado a través de un servidor.
Siempre estará activa la transferencia o comunicación debido a los múltiples nodos activos en la red.	Cuando un servidor esta caído o inactivo, las peticiones de los clientes no pueden ser satisfechas.
Estas redes por definición son descentralizadas y todos los nodos son iguales. No existen nodos con funciones especiales.	Se limita a la centralización de un host: Hay que limitar funciones para cada cliente.

Tabla 1: Diferencias entre arquitectura P2P y arquitectura Cliente-servidor

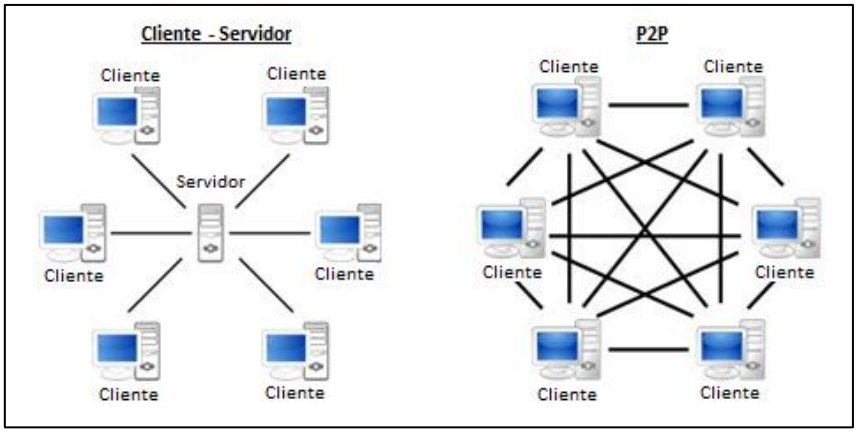

Figura 2: Ejemplos de topología Cliente-servidor y P2P

1.3.4 Hash

Las funciones hash, también llamadas funciones resumen, son algoritmos que generan una salida alfanumérica a partir de una entrada. Esta entrada puede ser texto, una contraseña o un archivo.

La salida de la función hash, denominada popularmente como ***hash***, contiene un resumen de toda la información que se le ha pasado como entrada, es decir, a partir de los datos de entrada genera como salida una cadena alfanumérica que solo puede ser creada con esos datos de entrada.

Las funciones hash tienen dos características principales:

- Son funciones de un único sentido. Esto significa que a partir del hash no se puede calcular el valor de la entrada.

- No hay posibilidad de colisión. Esto significa que dos archivos distintos no pueden tener el mismo valor de hash. Por lo que nos permite identificar un archivo de forma univoca.

En el caso de Ares, cada archivo compartido en su red tiene un hash asociado. Este hash es único para cada archivo, ya que se calcula en función del contenido del mismo. No hay dos hashes iguales para dos archivos distintos, por lo que los archivos se identifican de forma univoca en la red.

SHA-1

La familia SHA (*Secure Hash Algorithm*, Algoritmo de Hash Seguro) es un sistema de funciones hash criptográficas relacionadas de la Agencia de Seguridad Nacional de los Estados Unidos y publicadas por el *National Institute of Standards and Technology* (NIST). El primer miembro de la familia fue publicado en 1993 y es oficialmente llamado SHA. Sin embargo, hoy día, no oficialmente se le llama SHA-0 para evitar confusiones con sus sucesores. Dos años más tarde el primer sucesor de SHA fue publicado con el nombre de **SHA-1**. Existen cuatro variantes más que se han publicado desde entonces cuyas diferencias se basan en un diseño algo modificado y rangos de salida incrementados: SHA-224, SHA-256, SHA-384, y SHA-512 (llamándose SHA-2 a todos ellos).

SHA-1 ha sido examinado muy de cerca por la comunidad criptográfica pública, y no se ha encontrado ningún ataque efectivo.

El algoritmo hash SHA-1 producen una salida resumen de **160 bits** (20 bytes) de un mensaje que puede tener un tamaño máximo de 264 bits, y se basa en principios

similares a los usados por el profesor Ronald L. Rivest del MIT en el diseño de los algoritmos de resumen de mensaje MD4 y MD5.

Un ejemplo de hash SHA-1 es el siguiente:

SHA1(" ") = da39a3ee5e6b4b0d3255bfef95601890afd80709

1.3.5 Servidor web

Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor realizando conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente generando una respuesta en cualquier lenguaje o aplicación del lado del cliente. El código recibido por el cliente suele ser compilado y ejecutado por un navegador web. Para la transmisión de todos estos datos suele utilizarse algún protocolo. Generalmente se utiliza el protocolo HTTP para estas comunicaciones, perteneciente a la capa de aplicación del modelo OSI. El término también se emplea para referirse al ordenador que ejecuta el programa.

El Servidor web se ejecuta en un ordenador manteniéndose a la espera de peticiones por parte de un cliente (un navegador web) y que responde a estas peticiones adecuadamente, mediante una página web que se exhibirá en el navegador o mostrando el respectivo mensaje si se detectó algún error.

El cliente es el encargado de interpretar el código HTML, es decir, de mostrar las fuentes, los colores y la disposición de los textos y objetos de la página; el servidor tan sólo se limita a transferir el código de la página sin llevar a cabo ninguna interpretación de la misma.

Servidor web local

Un Servidor web local es aquel servidor web que reside en una red local al equipo de referencia. El Servidor web local puede estar instalado en cualquiera de los equipos que forman parte de una red local. Es por tanto obvio, que todos los servidores web, son locales a la red local en la que se encuentran, o como mínimo, locales al sistema en el que están instalados.

Cuando un servidor Web se encuentra instalado en el mismo equipo desde el cual se desea acceder puede utilizarse la dirección de *Loopback*, 127.0.0.1:80 en Ipv4 y 127.0.0.1::1 en Ipv6. Los archivos se almacenan en un directorio determinado por la configuración, generalmente modificable.

Existen numerosas aplicaciones que facilitan la instalación automática de servidores web Apache y aplicaciones adicionales como Mysql y PHP (entre otros), de forma conjunta, como XAMPP o EasyPHP. Estas aplicaciones reciben el nombre de LAMP cuando se instalan en plataformas Linux, WAMP en sistemas Windows y MAMP en sistemas Apple Macintosh.

En este TFG se utiliza un servidor web local (Apache, XAMPP), ya que tanto el servidor web como el cliente de Ares modificado están en el mismo ordenador.

1.3.6 Ares

Ares Galaxy, popularmente conocido como Ares, es un programa P2P de compartición de archivos creado a mediados de 2002. Es software libre y está desarrollado en el lenguaje de programación Delphi para el sistema operativo Microsoft Windows. Actualmente también se puede usar la red de Ares en GNU/Linux.

Ares originalmente trabajaba con la red Gnutella, pero seis meses después de su creación, en diciembre de 2002, se optó por empezar a desarrollar su propia red independiente y descentralizada, montada sobre una arquitectura de red P2P de tipo "*nodos hoja*-y-supernodos" ofreciendo un sistema de búsqueda de tipo *broadcasting* inspirada por la arquitectura de la red P2P Gnutella; fue así como empezó a nacer lo que sería la red de Ares Galaxy. Muchos seguidores del programa sostienen que posee velocidades de descarga y búsquedas superiores a las de otros clientes P2P, además de conectar rápido a la red. Los usuarios utilizan o han utilizado Ares principalmente para la compartición de archivos de audio.

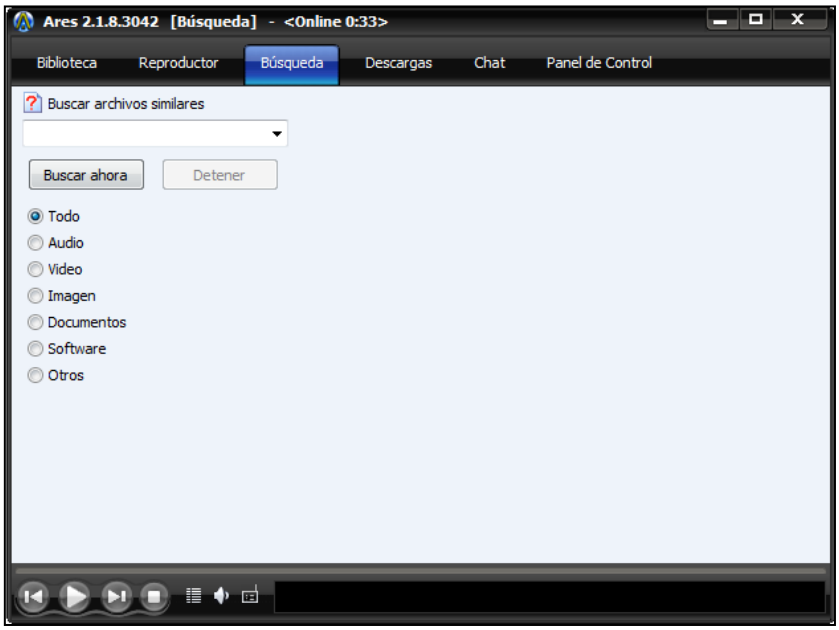

Figura 3: Interfaz de Ares 2.1.8

Capítulo 2

Descripción del sistema

Este capítulo nos permitirá entender cómo funciona la red de Ares, como es su arquitectura, cómo se realizan las búsquedas de archivos y como se identifican tanto los archivos como los usuarios que los poseen.

Por último, se describirán detalladamente las tecnologías utilizadas en este TFG, dando un breve resumen de su historia y usos y explicando sus principales funcionalidades

2.1 Arquitectura de la red de Ares

En este apartado, vamos a explicar la topología de la red de Ares y a describir cada una de las fases que realiza el cliente de Ares desde que lo ejecutamos hasta que se realiza la descarga de un archivo.

Ares utiliza una red P2P descentralizada con nodos agrupados en estrella. Esta forma de agrupación hace que los nodos que forman la estrella sean vecinos unos de otros. Estas agrupaciones son de 5 en 5 nodos.

La primera vez que iniciamos el cliente de Ares, este obtiene una lista que contiene 5 nodos vecinos. A través de estos nodos vecinos, siempre que estén activos, nuestro cliente esta interconectado con el resto de nodos de la red de Ares.

El funcionamiento de la red pasa por tres fases:

- ***Entrada:*** En esta fase un nuevo nodo se conecta a un nodo vecino que ya esté dentro de la red. Cada cliente tiene una lista de nodos que se espera estén siempre conectados y se escoge alguno al azar. Un nodo cualquiera puede estar conectado a varios nodos, y recibir conexiones de nuevos nodos formando una malla aleatoria no estructurada.

- ***Búsquedas:*** Cuando un nodo desea buscar un fichero, le envía un mensaje a todos los nodos vecinos a los que está conectado (broadcasting). Estos buscan localmente si lo ofrecen, y a la vez reenvían la búsqueda a todos los nodos a los que ellos están conectados. Esta estrategia de difusión se llama inundación de

la red, y existen mecanismos para evitar reenvíos infinitos y bucles. Cuando una petición llega a un nodo que ofrece el fichero, se contesta directamente al nodo que inició la búsqueda.

La inundación producida durante esta fase es la debilidad más importante de este protocolo debido a que, si hay muchas búsquedas a la vez, la red se llena de mensajes de búsqueda que los nodos se reenvían entre sí. Aun así, el hecho de que no exista un servidor central que procese las peticiones de búsqueda, hace que el protocolo sea más robusto ante posibles caídas de nodos.

- ***Descargas:*** La descarga se realiza directamente desde los nodos que contestaron a la búsqueda del fichero. Los ficheros pueden partirse en varios trozos servidos por diferentes nodos, y los clientes suelen incluir un sistema de comprobación final de la integridad del fichero.

Una vez explicadas las fases de funcionamiento, vamos a hablar de la gestión y de la velocidad en las descargas de Ares.

Una de las razones por las que la red P2P de Ares es más rápida que el resto de redes es el método que utiliza para gestionar las descargas. Este método consiste en dar mayor prioridad a aquellos nodos cuyo porcentaje de descarga completada sea menor. Con esto lo que se consigue es aumentar la cantidad de “partes” de un archivo descargadas por los usuarios, lo que aumenta la disponibilidad de dichas partes y las fuentes por archivo.

Los archivos se dividen en varias partes, todas del mismo tamaño. Cada vez que un usuario (nodo) se descarga una parte completa, automáticamente se convierte en servidor de dicha parte, por lo que un usuario no tiene que tener un archivo completamente descargado para convertirse en servidor de él.

Darles mayor prioridad a los usuarios que tienen menor porcentaje de descarga de un archivo significa que a los usuarios que tienen menos partes completas de un archivo les aumentamos la capacidad de descarga para que puedan descargar más partes y a su vez poner a disposición del resto de usuarios dichas partes.

Esto mejora enormemente el rendimiento, ya que a la vez que descargas un archivo, compartes las partes completamente descargadas de este. Ahora, podemos hacernos una idea del ritmo al que crecen las fuentes de un archivo y el nivel de compartición que ofrece esta red.

2.2 Identificación de usuarios

Las redes P2P se basan en conexiones directas entre dos usuarios para transferir archivos. Para que esta conexión se produzca se debe conocer la IP del usuario que posee el archivo. Como veremos en el apartado 2.4, cuando realizamos una petición de un archivo en la red, los usuarios que posean dicho archivo, contestan a la petición con su IP, para que el usuario que busca el archivo pueda conectar con él.

El principal problema que existe en la red de Ares es que no existe un ID que identifique unívocamente a un usuario. Cada usuario decide el nombre (username) que quiere tener en la red, por lo que varios usuarios pueden tener el mismo username en la red. En el caso de que el usuario decida no establecer un nombre, Ares le asigna un nombre con el siguiente formato: anon_XXXXXX.

Las X de después del guion bajo representan la IP del usuario codificada en hexadecimal. Esto se explicara más detalladamente en el apartado 3.2, en el cual se analizan las tramas enviadas y recibidas por el cliente de Ares.

Este *username* cambiara de valor cuando cambie la IP al usuario, por lo que no es un identificador único y exclusivo en el tiempo. Por tanto, la asignación de dicho nombre es dinámica y depende de la IP actual de dicho usuario.

Tanto si el usuario elige su nombre, como si se lo asigna Ares, no podemos identificar unívocamente a un usuario. Debemos basarnos en los periodos de uso de dicha IP para relacionar a un usuario concreto con la descarga o tenencia de un archivo, lo que complica bastante la identificación de usuarios.

2.3 Identificación de archivos

Este apartado es uno de los más importantes, ya que nos permite comprobar que un archivo que nos hemos descargado es realmente el que estábamos buscando.

Esto se consigue gracias al ***hash***. El hash de un archivo se puede calcular en cualquier momento y hay varias funciones hash que te lo pueden calcular, siendo las más comunes y seguras MD5 y SHA-1, que es el tipo de hash utilizado en la red de Ares

Todos los ficheros tienen un identificador hash. Este hash es una combinación de números y letras que son capaces de identificar de forma única a un fichero. Un fichero puede tener múltiples nombres, pero esto no altera su valor hash. Esto sucede debido a que el hash se calcula en función del contenido del archivo, y eso nunca cambia. Esto permite a los usuarios encontrar todas las fuentes para un archivo en particular sin importar si alguien le ha cambiado el nombre al mismo. Ares utiliza el

hash SHA-1, tanto para localizar archivos como para identificar errores en las descargas.

Ares también soporta ***hashlinks***, que son otro tipo de identificadores únicos del archivo. A diferencia del hash, tienen la forma de enlaces (links) que comienzan por "arlnk://". Dichos links son publicados en algunas páginas de internet para que los usuarios tengan acceso a ellos. Cuando se utilizan los hashlink, el cliente de Ares busca y descarga un archivo especifico.

Tanto el hash SHA-1 como el hashlinks son identificadores únicos para cada archivo, ya que son generados con operaciones matemáticas realizadas sobre el contenido del archivo.

2.4 Búsqueda de archivos y fuentes

En este apartado vamos a describir el proceso de búsqueda de un archivo y cómo se consiguen las fuentes para realizar la descarga de dicho archivo.

Ya hemos visto en anteriores capítulos la arquitectura de la red *Ares Galaxy* y los métodos de gestión de descargas. Ambas partes afectan por igual a la búsqueda de archivos y obtención de fuentes, ya que la arquitectura juega un papel fundamental en la transmisión de las peticiones y el método de gestión de descargas, explicado en el apartado 2.1, es el culpable de la gran cantidad de fuentes disponibles que tienen los archivos de esta red. Debido a estos pilares fundamentales, casi cualquier archivo que busquemos a través de esta red será encontrado y descargado en un corto periodo de tiempo.

Cuando se realiza una petición de búsqueda, el cliente de Ares envía dicha petición a los cinco nodos vecinos, todos ellos conectados en forma de estrella.

Los nodos vecinos responden al usuario y a la vez reenvían la petición de búsqueda a sus propios nodos vecinos. Cuando una petición llega a un nodo que tiene el fichero, este contesta directamente al nodo que inició la búsqueda. De esta forma se realiza una búsqueda por inundación de la red.

Esta búsqueda por inundación se realiza en función del hash o del hashlink del archivo. Los nodos vecinos devuelven una lista con los usuarios y IPs asociadas a los usuarios que poseen dicho archivo. De este modo, el usuario que realizó la petición de búsqueda, cuenta con todas las fuentes enviadas por los nodos vecinos para realizar la descarga.

Este método de búsqueda de archivos tiene sus pros y sus contras ya que, si muchos usuarios realizan búsquedas de forma simultánea, la red se llena de mensajes que los nodos se reenvían entre sí. Esto hace que la red se sature y algunos nodos puedan no dar abasto debido al número de peticiones recibidas.

Aunque esto parezca muy negativo no lo es, ya que la caída de algunos nodos o la posible saturación de alguno no influyen en la búsqueda debido a que hay muchos nodos en la red que contestaran satisfactoriamente a dichas peticiones.

Una vez que se obtienen los resultados de la búsqueda, el usuario que realizó la petición tendrá una lista de archivos que coinciden con el parámetro de búsqueda que ha introducido y, asociados a cada uno de ellos, un número de fuentes (usuarios) de las cuales conocemos las IPs.

Lo último que tiene que hacer el usuario es seleccionar que archivo se quiere descargar y, automáticamente, se conectara a través de la IP a una o a varias fuentes distintas para comenzar la descarga del archivo.

2.5 Compartición de archivos

En este apartado se explica cómo se lleva a cabo la compartición de archivos en la red de Ares. Dicha compartición resulta muy sencilla ya que los archivos que compartimos se encuentran alojados en la misma carpeta de descargas de Ares.

También podemos autorizar a Ares para que comparta los archivos de otros directorios, pero por defecto se comparten los archivos de la carpeta de descarga de Ares, denominada “My Shared Folder”. Con esto se consigue que cada usuario que se descarga un archivo de la red se convierta automáticamente en servidor de este archivo, para futuras descargas de otros usuarios.

Cada vez que se añade un archivo al directorio de descargas de Ares, el cliente actualiza la lista de archivos compartidos por dicho usuario. Así, cuando un usuario realice una búsqueda, esta petición se va retransmitiendo entre nodos como una inundación, cada nodo comprueba en su lista de archivos compartido en la carpeta de descargas de Ares si lo tiene, y si es positivo, contesta con su IP para que el cliente que realizo la búsqueda pueda abrir una conexión directamente con el nodo que tiene el archivo buscado.

Como hemos dicho antes, cada archivo que se descarga un usuario va a parar a esta carpeta y automáticamente se convierte en servidor de dicho archivo, lo que aumenta constantemente el número de fuentes de un archivo y mejora la disponibilidad del archivo para el resto de usuarios.

Por último, una de las posibilidades que tenemos es cambiar la carpeta de destino de las descargas y asignar otro directorio que nos sea más cómodo. Por defecto, como hemos dicho anteriormente, es “My Shared Folder” pero puede ser cambiada en cualquier momento siguiendo los pasos que vamos a ver a continuación.

Con el cliente de Ares abierto, hacemos clic en la pestaña “Panel de control”. Una vez en este apartado, veremos que aparecen otra serie de pestañas en las cuales podemos modificar algunas de las propiedades del cliente de Ares.

Hacemos clic en la pestaña “Descargas”, en la cual veremos las principales variables implicadas en la gestión de descargas del cliente.

Una vez aquí, en el apartado “Carpeta de descargas” podemos ver la ruta actual de la carpeta de descargas. Si queremos cambiar la carpeta de descargas, lo único que tenemos que hacer es introducir la ruta de la nueva carpeta de descargas sustituyendo la anterior.

Figura 4: Localización de la carpeta de descargas de Ares

2.6 Tecnologías y lenguajes utilizados

2.6.1 XAMPP

XAMPP es un servidor multiplataforma que está formado por el gestor de base de datos MySQL, el servidor Web Apache y los intérpretes para lenguajes de script: PHP y Perl. El nombre proviene del acrónimo de X (cualquiera de los diferentes sistemas operativos), Apache, MySQL, PHP, Perl. El programa actúa como un servidor Web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNULinux, Solaris, y MacOS. XAMPP es un software libre, lo que lo convierte en gratuito y libre para ser copiado conforme los términos de la licencia GNU (Licencia Pública General).

XAMPP también incluye otros módulos como OpenSSL y phpMyAdmin, para incorporar seguridad y gestionar las bases de datos respectivamente. La filosofía detrás de XAMPP es la construcción de una versión fácil de instalar para los desarrolladores que entran al mundo de Apache. Para hacerlo más conveniente, XAMPP está configurado con todas las funciones activadas, pero esta configuración no es buena desde el punto de vista de seguridad y no es suficientemente buena para un ambiente de producción, por este motivo se han modificado numerosos parámetros de su configuración.

Figura 5: Principales componentes de XAMPP

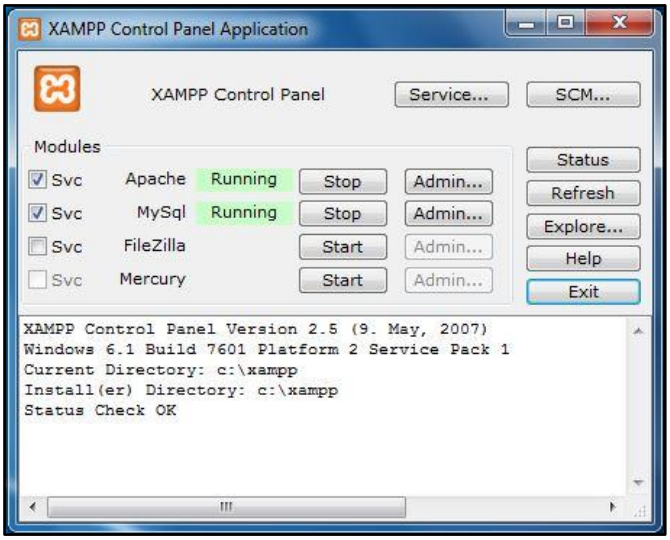

Figura 6: Panel de control de XAMPP

Los principales módulos empleados de XAMPP son los que se explican a continuación:

Apache

El servidor HTTP Apache es un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows y MAC entre otras. El hecho de que sea un sistema multiplataforma lo convierte en el servidor HTTP más utilizado del mundo. Apache es usado principalmente para enviar páginas web estáticas y dinámicas en la World Wide Web. Muchas aplicaciones web están diseñadas asumiendo el entorno y las características propias de Apache, pero éste es usado para muchas otras tareas donde el contenido necesita ser puesto a disposición en una forma segura y confiable.

La arquitectura del servidor Apache es modular. El servidor consta de una sección principal y diversos módulos que aportan muchas de las funcionalidades que podría considerarse básica para un servidor web. Los módulos más interesantes para el desarrollo del proyecto son: el módulo para comunicaciones seguras mediante seguridad en la capa de transporte (SSL y TLS) y un módulo externo para gestión de páginas dinámicas en PHP.

La licencia de software bajo la cual el software de la fundación Apache es distribuido es una parte distintiva de la historia de Apache HTTP Server y de la comunidad de código abierto. La Licencia Apache permite la distribución de derivados de código abierto y cerrado a partir de su código fuente original.

MySQL

MySQL es un sistema de gestión de base de datos (SGBD) veloz, multihilo, multiusuario y robusto. Éste está proyectado tanto para sistemas críticos en producción, soportando intensas cargas de trabajo, como para empotrarse en sistemas de desarrollo masivo de software. El software MySQL tiene licencia dual, pudiéndose usar de forma gratuita bajo licencia GNU o bien adquiriendo licencias comerciales de Sun Microsystems en el caso de no desear estar sujeto a los términos de la licencia GPL.

Existen varias APIs que permiten a aplicaciones externas escritas en diversos lenguajes de programación acceder a las bases de datos MySQL. Entre estos lenguajes podemos encontrar C, C++, Pascal, Delphi, Perl, PHP, Java o TCL. Cada uno de estos utiliza una API específica.

Cualquier consulta a la base de datos se hace por medio del lenguaje **SQL** (Structured Query Language). Éste es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre ellas. Una de sus características es el manejo del álgebra y el cálculo relaciona] permitiendo efectuar consultas con el fin de recuperar información de interés de una base de datos, así como también hacer cambios sobre ella. Es un lenguaje declarativo de "alto nivel" o "de no procedimiento", que gracias a su fuerte base teórica y su orientación al manejo de conjuntos de registros, y no a registros individuales, permite una alta productividad en codificación y la orientación a objetos. De esta forma una sola sentencia puede equivaler a uno o más programas que utilizados en un lenguaje de bajo nivel orientado a registro.

En este TFG, la base de datos MySQL es utilizada para almacenar distintos tipos de información en varias tablas:

- Rango de IPs que pertenecen a los distintos países.
- Lista con los archivos a buscar que contiene el Hash SHA-1 y el tamaño de cada uno.
- IP de cada usuario que haya sido registrado y su *Username*.
- Lista con un identificador asignado a cada hash y otro identificador para la IP que lo tiene.

PHP

PHP es un lenguaje interpretado de propósito general ampliamente usado y está diseñado especialmente para desarrollo web y puede ser incrustado dentro de código HTML. Éste es usado principalmente en interpretación del lado del servidor (server-side scripting), tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. Es publicado bajo la PHP License, considerando esta licencia como software libre.

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones. Cuando el cliente hace una petición al servidor pata que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente.

Mediante extensiones es también posible la generación de archivos PDF, Flash, así como im6genes en diferentes formatos; además permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC, DES, Microsoft SQL Server, Fibird y SQLite.

PhpMyAdmin

PhpMyAdmin es una herramienta escrita en PHP con el fin de administrar bases de datos de MySQL a través de páginas web (Figura IV). Actualmente puede crear y eliminar bases de datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios, exportar datos en varios formatos y está disponible en 50 idiomas.

Al igual que PHP, se encuentra disponible bajo la licencia GPL.1151

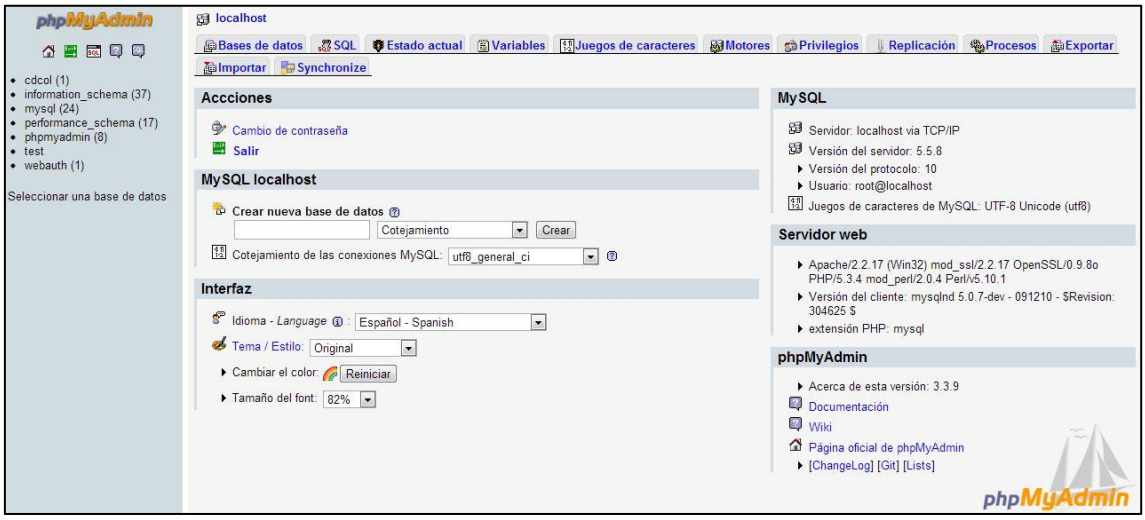

Figura 7: Interfaz de phpMyAdmin

2.6.2 Borland Delphi 7

Borland Delphi, antes conocido como CodeGear Delphi y Inprise Delphi, es un entorno de desarrollo de software diseñado para la programación de propósito general con énfasis en la programación visual. En Delphi se utiliza como lenguaje de programación una versión moderna de Pascal llamada Object Pascal. Es producido comercialmente por la empresa estadounidense CodeGear (antes lo desarrollaba Borland), adquirida en mayo de 2008 por Embarcadero Technologies, una empresa del grupo Thoma Cressey Bravo, en una suma que ronda los 30 millones de dólares. En sus diferentes variantes, permite producir archivos ejecutables para Windows, GNU/Linux y la plataforma .NET.

CodeGear ha sido escindida de la empresa Borland, donde Delphi se creó originalmente, tras un proceso que pretendía en principio la venta del departamento de herramientas para desarrollo.

2.6.3 Delphi

Delphi está basado en una versión orientada a obejtos de Pascal denominada *Object Pascal*.

Borland en los últimos años defendía que el nombre correcto del lenguaje es también *Delphi*, posiblemente debido a pretensiones de marca, aunque en sus mismos manuales el nombre del lenguaje aparecía como *Object Pascal*, por lo que la comunidad de programadores no ha adoptado mayoritariamente este cambio. *Object Pascal* expande las funcionalidades del Pascal estándar:

- Soporte para la programación orientada a objetos también existente desde Turbo Pascal 5.5, pero más evolucionada en cuanto a:

    - Encapsulación: declarando partes privadas, protegidas, públicas y publicadas de las clases.

    - Propiedades: concepto nuevo que luego han adaptado muchos otros lenguajes. Las propiedades permiten usar la sintaxis de asignación para setters y getters (en Delphi setters = write y getters = read).

    - Simplificación de la sintaxis de referencias a clases y punteros.

- Soporte para manejo estructurado de excepciones, mejorando sensiblemente el control de errores de usuario y del sistema.

- Programación activada por eventos (*event-driven*), posible gracias a la técnica de delegación de eventos. Esta técnica permite asignar el método de un objeto para responder a un evento lanzado sobre otro objeto. Fue adoptada por Niklaus Wirth, autor del Pascal Original, e incorporada a otros de sus lenguajes como Component Pascal.

Un uso habitual de Delphi, aunque no el único, es el desarrollo de aplicaciones visuales y de bases de datos cliente-servidor y multicapas. Debido a que es una herramienta de propósito múltiple, se usa también para proyectos de casi cualquier tipo, incluyendo aplicaciones de consola, aplicaciones de web (por ejemplo servicios web, CGI, ISAPI, NSAPI, módulos para Apache), servicios COM y DCOM, y servicios del sistema operativo. Entre las aplicaciones más populares actualmente destaca Skype, un programa de telefonía por IP.

Delphi inicialmente sólo producía ejecutables binarios para Windows: Delphi 1 para Win16 y con Delphi 2 se introdujo Win32.

Una de las principales características y ventajas de Delphi es su capacidad para desarrollar aplicaciones con conectividad a bases de datos de diferentes fabricantes. El programador de Delphi cuenta con una gran cantidad de componentes para realizar la conexión, manipulación, presentación y captura de los datos, algunos de ellos liberados bajo licencias de código abierto o gratuito. Estos componentes de acceso a datos pueden enlazarse a una gran variedad de controles visuales, aprovechando las características del lenguaje orientado a objetos, gracias al polimorfismo.

En la paleta de componentes pueden encontrarse varias pestañas para realizar una conexión a bases de datos usando diferentes capas o motores de conexión.

Hay motores que permiten conectarse a bases de datos de diferentes fabricantes tales como BDE, DBExpress o ADO, que cuentan con manejadores para los formatos más extendidos.

También hay componentes de conexión directa para un buen número de bases de datos específicas: Firebird, Interbase, Oracle, etcétera.

Delphi posee API's para varios tipos de bases de datos como MySQL, PostgreSQL y Microsoft Access entre otros.

En este TFG se utiliza la base de datos de MySQL, en la cual se almacenan todos los datos de cada IP registrada por el cliente de Ares modificado.

2.6.4 SQL

El lenguaje de consulta estructurado o SQL, por sus siglas en inglés *Structured Query Language*, es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en ellas. Una de sus características es el manejo del álgebra y el cálculo relacional que permiten efectuar consultas con el fin de recuperar de forma sencilla información de interés de bases de datos, así como hacer cambios en ella.

SQL explota la flexibilidad y potencia de los sistemas relacionales y permite así gran variedad de operaciones. Es un lenguaje declarativo de "alto nivel que, gracias a su fuerte base teórica y su orientación al manejo de conjuntos de registros, permite una alta productividad en codificación y la orientación a objetos.

Algunas de las principales características de SQL son:

- Lenguaje de definición de datos: El LDD de SQL proporciona comandos para la definición de esquemas de relación, borrado de relaciones y modificaciones de los esquemas de relación.

- Lenguaje interactivo de manipulación de datos: El LMD de SQL incluye lenguajes de consultas basado tanto en álgebra relacional como en cálculo relacional de tuplas.

- Integridad: El LDD de SQL incluye comandos para especificar las restricciones de integridad que deben cumplir los datos almacenados en la base de datos.

2.6.5 HTML

HTML, siglas de HyperText Markup Language (Lenguaje de marcado hipertextual), es el lenguaje predominante para la elaboración de páginas web. Se utiliza para describir y traducir la estructura y la información en forma de texto, así como para complementar el texto con objetos tales como imágenes.

HTML utiliza etiquetas o marcas, que consisten en breves instrucciones de comienzo y final, mediante las cuales se determina la forma en la que debe aparecer en su navegador el texto, así como también las imágenes y los demás elementos, en la pantalla del ordenador. Las etiquetas se identifican porque aparecen encerradas entre los signos "menor que" y mayor que" (< >) y algunas tienen atributos que pueden tomar diferentes valores.

El lenguaje HTML puede ser creado y editado con cualquier editor de textos básico, como puede ser Gedit en Linux, el Bloc de notas de Windows, o cualquier otro editor que admita texto sin formato como GNU o Notepad++.

HTML también permite incluir código PHP y un script (por ejemplo, JavaScript), el cual puede afectar al comportamiento de navegadores web y otros procesadores de HTML, y código PHP.

2.6.6 Adobe Photoshop CS5

Adobe Photoshop es un software de edición gráfica orientado al retoque de imágenes. Está considerado como el mejor programa de manipulación fotográfica del mundo, debido a la multitud de herramientas que posee y a su versatilidad. Con este programa podemos realizar tareas que van desde el retoque más sencillo en una fotografía hasta el diseño de plantillas para páginas web, sin olvidarnos de su utilización en edición y efectos especiales en la mayoría de películas actuales.

Debido a su utilización en el ámbito profesional, *Photoshop* soporta la mayoría de los formatos de imágenes que existen, además de tener formatos propios como PSD y PDD.

Photoshop forma parte de la familia **Adobe Creative Suite** y se puede conseguir de forma individual o en los paquetes de programas de edición profesional de Adobe, como **Adobe Creative Suite Master Collection**.

Fue lanzado al mercado informático en 1990, como un software para los ordenadores **Apple** (*Macintosh*) con sistema operático *MAC OS*, pero desde 1992 se le unió la posibilidad de funcionamiento en el sistema operativo **Windows**.

Su versión más reciente es la CS6 (2012), pero para la realización de este proyecto se ha utilizado la versión CS5, ya que cumplía con todos los requisitos.

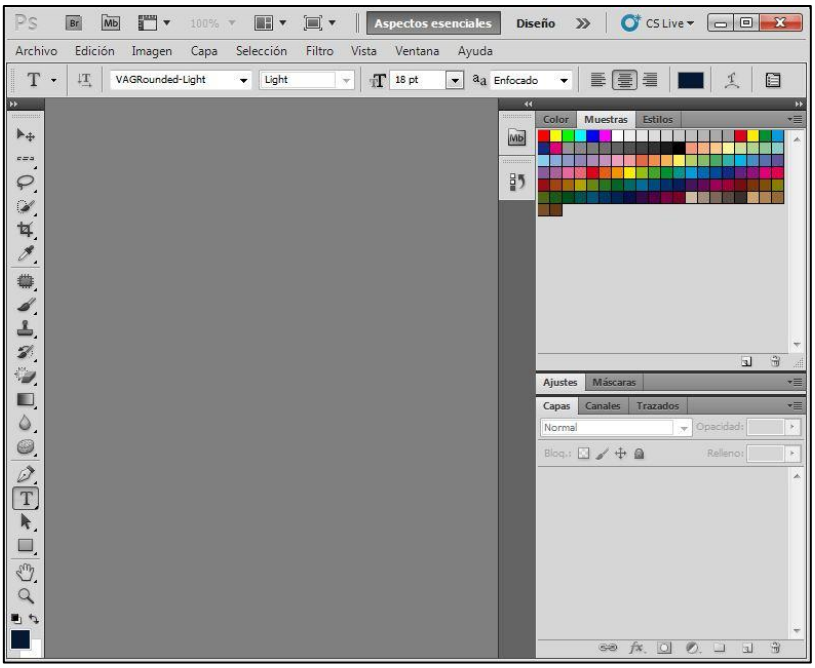

Figura 8: Interfaz de Adobe Photoshop CS5

2.6.7 Wireshark

Wireshark, antes conocido como Ethereal, es un analizador de protocolos utilizado para realizar análisis y solucionar problemas en redes de comunicaciones, para desarrollo de software y protocolos, y como una herramienta didáctica para educación. Cuenta con todas las características estándar de un analizador de protocolos.

La funcionalidad que provee es similar a la de tcpdump, pero añade una interfaz gráfica y muchas opciones de organización y filtrado de información. Así, permite ver todo el tráfico que pasa a través de una red (usualmente una red Ethernet, aunque es compatible con algunas otras) estableciendo la configuración en modo promiscuo. También incluye una versión basada en texto llamada tshark.

Permite examinar datos de una red viva o de un archivo de captura salvado en disco. Se puede analizar la información capturada, a través de los detalles y sumarios por cada paquete. Wireshark incluye un completo lenguaje para filtrar lo que queremos ver y la habilidad de mostrar el flujo reconstruido de una sesión de TCP.

Wireshark es software libre, y se ejecuta sobre la mayoría de sistemas operativos Unix y compatibles, incluyendo Linux, Solaris, FreeBSD, NetBSD, OpenBSD, y Mac OS X, así como en Microsoft Windows.

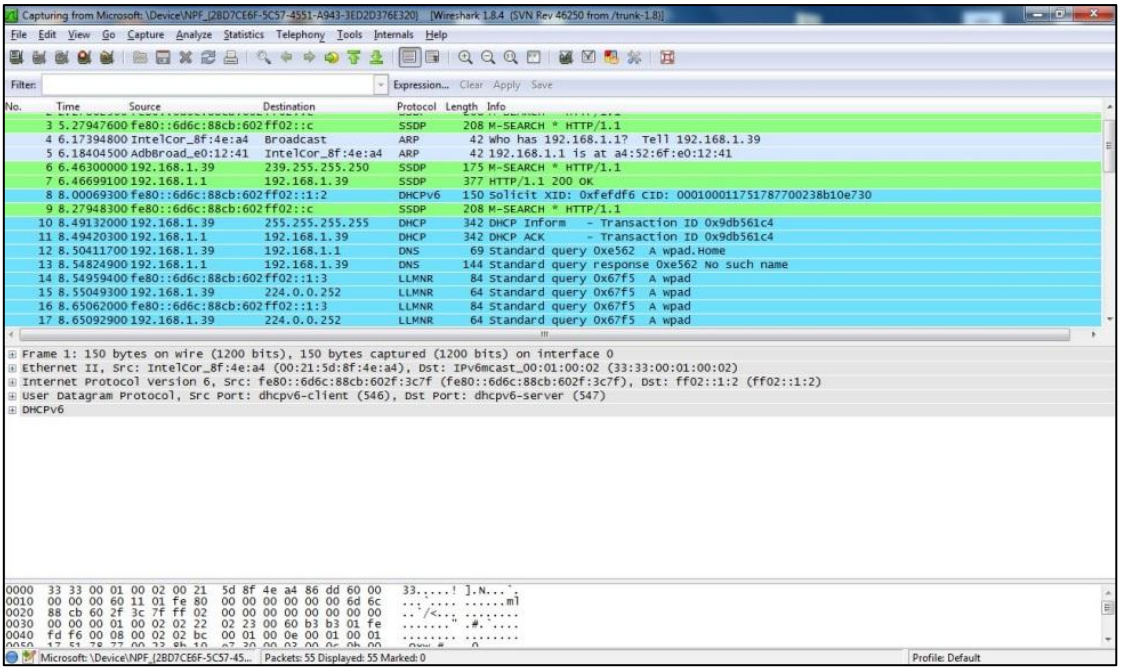

Figura 9: Interfaz de Wireshark

En la realización de este proyecto, se ha utilizado Wireshark para capturar el tráfico mientras se realiza una búsqueda y una descarga de un archivo con el fin de analizar posteriormente tal información. Examinando la captura y mediante el filtrado de información que ofrece Wireshark, se ha conseguido conocer información muy útil de la red de Ares como:

- Conexión con los nodos vecinos.
- Información más detallada de la búsqueda de un archivo por nombre o Hash SHA-1 (en el caso de ALCALARES) mediante el envío de mensajes a los nodos vecinos y éstos a su vez, a sus correspondientes.
- Paquetes recibidos con todas las posibles fuentes que tienen el archivo buscado y pueden compartirlo.
- Datos como el Username o la IP (Figura V) de cada fuente.

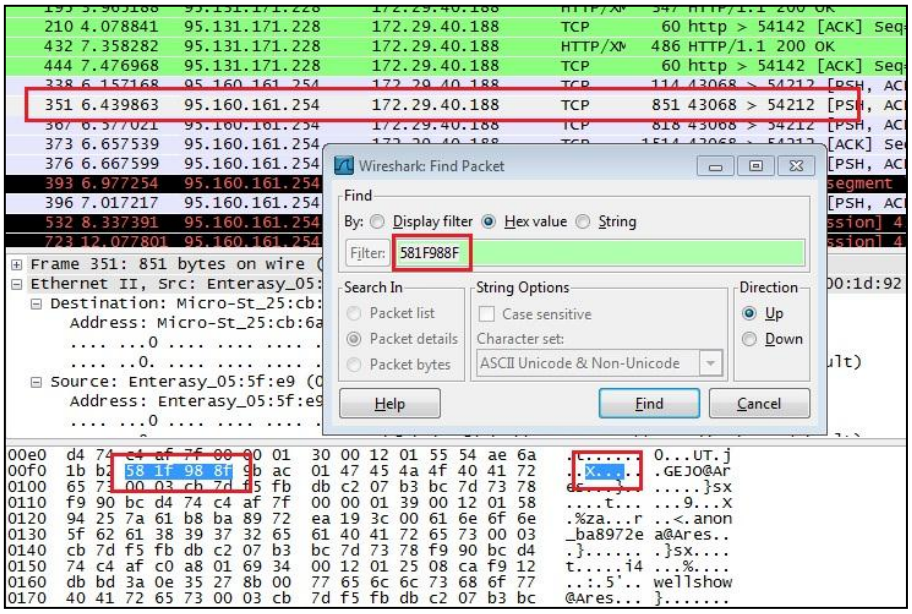

Figura 10: Captura de Wireshark de la IP de usuario en Hexadecimal

Capítulo 3

Trabajo realizado

En este capítulo se explicaran las modificaciones que se han realizado sobre el código del cliente de Ares para que realice las conexiones a la base de datos y la búsqueda por hash en la red de Ares.

Además, vamos a describir el diseño de la base de datos y el proceso de creación del servicio web, explicando cada una de las funcionalidades que nos ofrece.

Como complemento a todo esto, se ha realizado un análisis de las tramas implicadas en la descarga de un archivo, para conseguir comprender mejor el funcionamiento de la red de Ares y entender la función que realiza cada nodo implicado.

3.1 Servidor

El sistema tiene como eje de su funcionamiento un servidor. Este servidor tiene como principales funciones permitir el acceso al servicio web, gracias al funcionamiento de un servidor web, y el almacenamiento de la información recogida por el cliente de Ares modificado, gracias a una base de datos. A través del servicio web podemos gestionar la base de datos, creándola y viendo el contenido almacenado en la misma.

El servidor del sistema se basa en la herramienta XAMPP, explicada en el apartado 2.6.1.

Las tareas relacionadas con el servidor serán gestionadas por el administrador del sistema y la utilización de las distintas herramientas será explicada en el capítulo 4, que contiene el manual de usuario.

3.2 Análisis de tramas

Para comprender el funcionamiento de Ares, se va a proceder a descargar un fichero desde la red y se capturará el tráfico con el analizador de tramas Wireshark, explicado en el apartado 2.6.7.

Para poder descargar un fichero en Ares es necesario como mínimo el Hash y el tamaño del fichero buscado, así que insertamos estos datos directamente en el código fuente de Ares que realiza la descarga de ficheros a partir de hashlinks. El procedimiento en el cual se realiza la inserción del hash y el tamaño del archivo es 'add_weblink' de la clase 'Helper_Hashlinks.pas'.

```
filename:='03CB7DF5FBDBC207B3BC7D7378F990BCD474C4AF';
hash_sha1s:=HexToString('03CB7DF5FBDBC207B3BC7D7378F990BCD474C4AF');
sizec:=11330114;
```

Figura 11: Hash insertado en el código

El archivo que vamos a descargar lo vemos descrito en la figura 12.

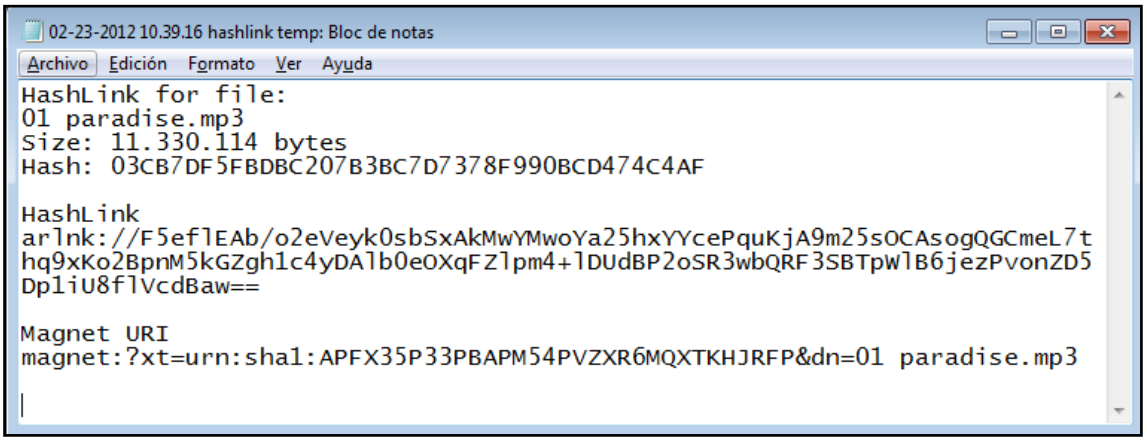

Figura 12: Información del archivo a descargar

Al comenzar a descarga el archivo a través de su hash y tamaño, podemos observar en él envió los siguientes paquetes hacia los nodos vecinos:

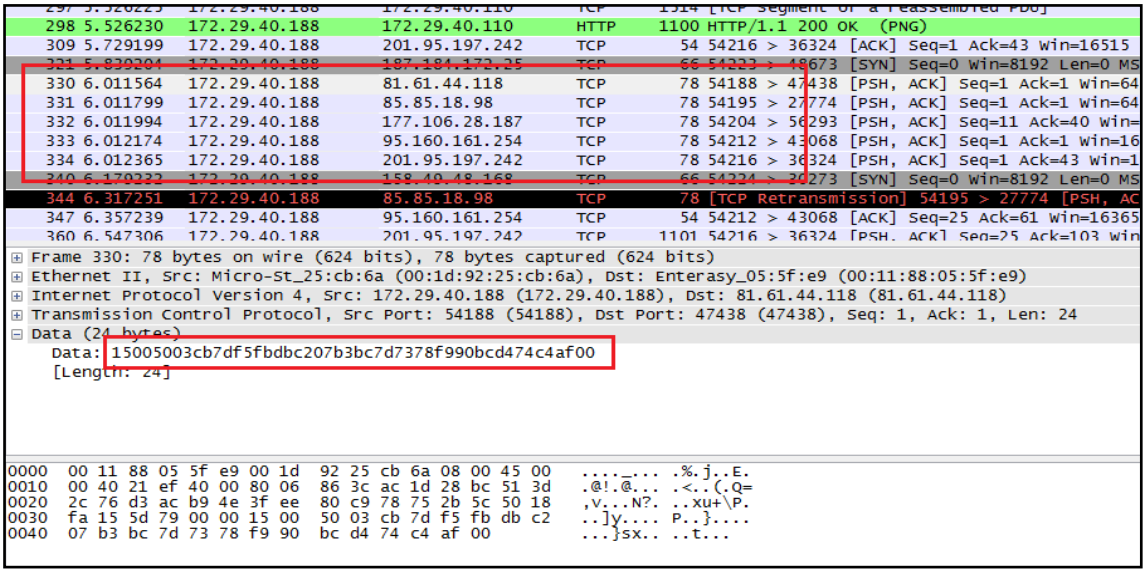

Figura 13: Captura de tramas al realizar petición

Se han enviado cinco paquetes a distintas IPs, en los cuales podemos observar que en el campo Data aparece el hash del archivo entre los bytes 150050 y 00:

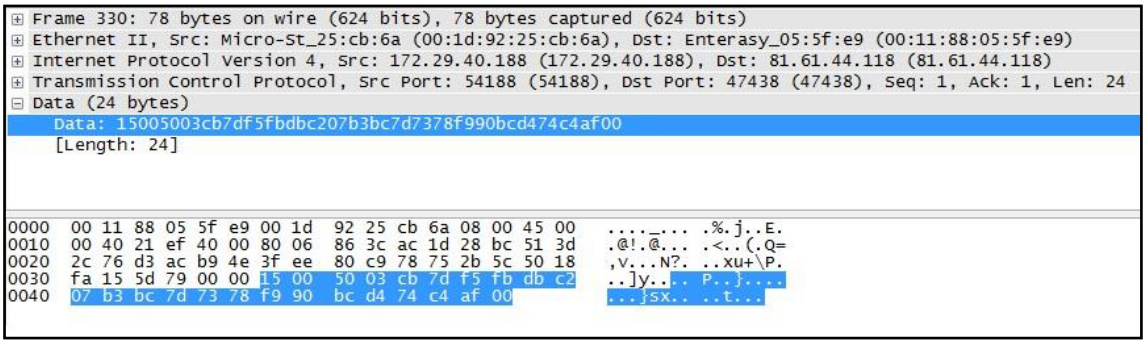

Figura 14: Detalle del paquete enviado

Las IPs a las que se han enviado los paquetes corresponden a las IPs de los 5 nodos vecinos que tiene el cliente de Ares, y están contenidas en el archivo SNodes.dat, cuya ubicación es el directorio:

"C:/Users/Usuario/AppData/Local/Ares/Data/SNodes.dat."

A continuación, en la figura X, aparece una captura extraída del archivo SNodes.dat donde podemos ver una de las IPs.

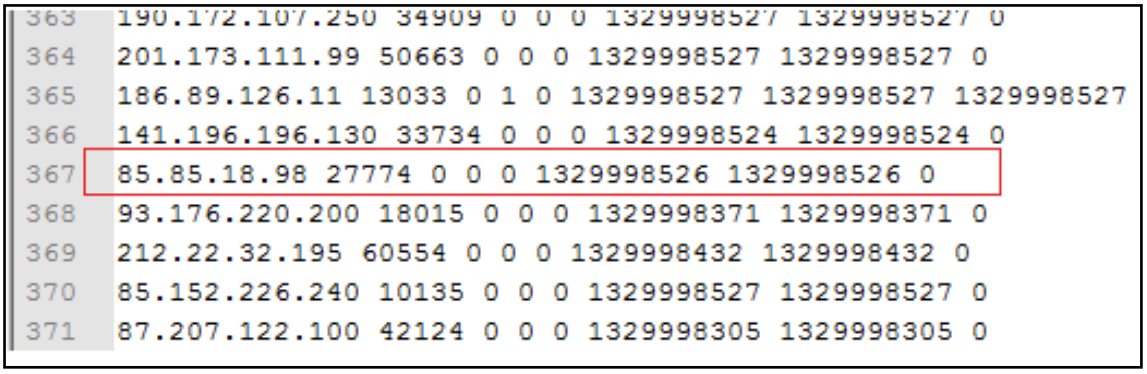

Figura 15: Captura del fichero SNodes.dat

Las IPs de los nodos vecinos a las cuales les hemos enviado la petición de búsqueda son:

- 81.61.44.118
- 85.85.18.98
- 177.106.28.187
- 95.160.161.254
- 201.95.197.242

En los paquetes devueltos por dichas IPs, hemos observado que aparecen los nombres de usuarios que poseen el fichero buscado:

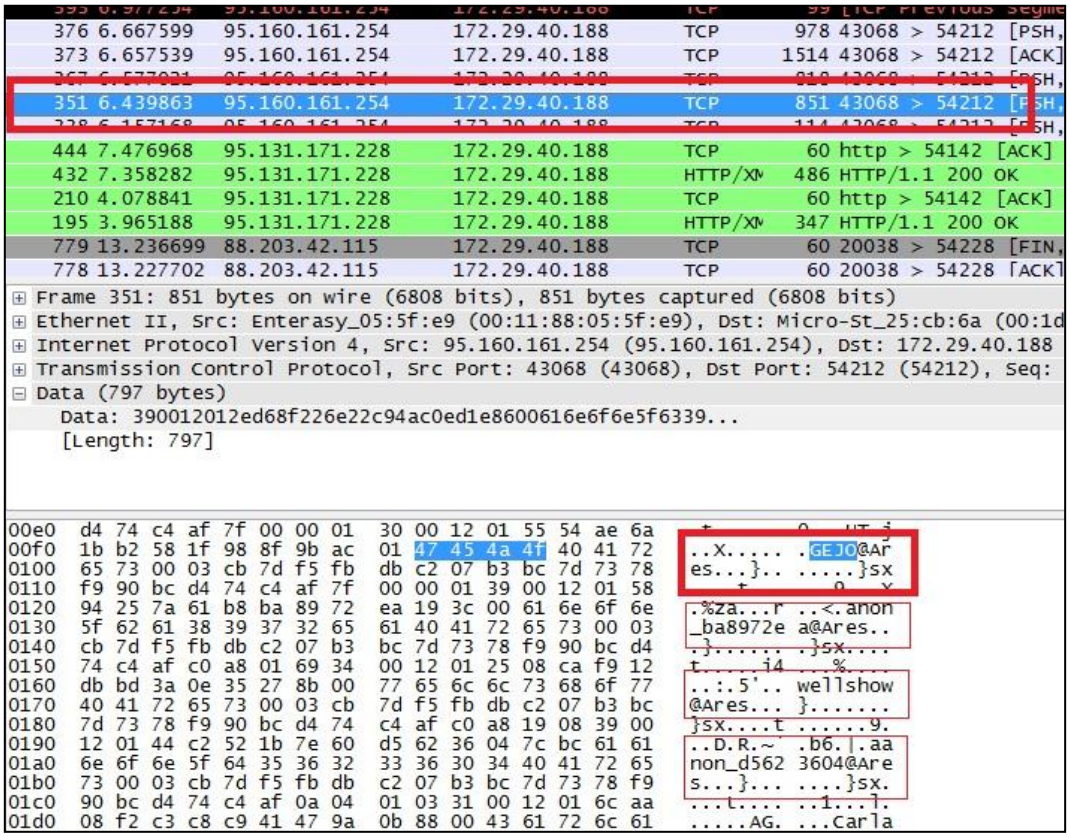

Figura 16: Captura de respuestas de los nodos

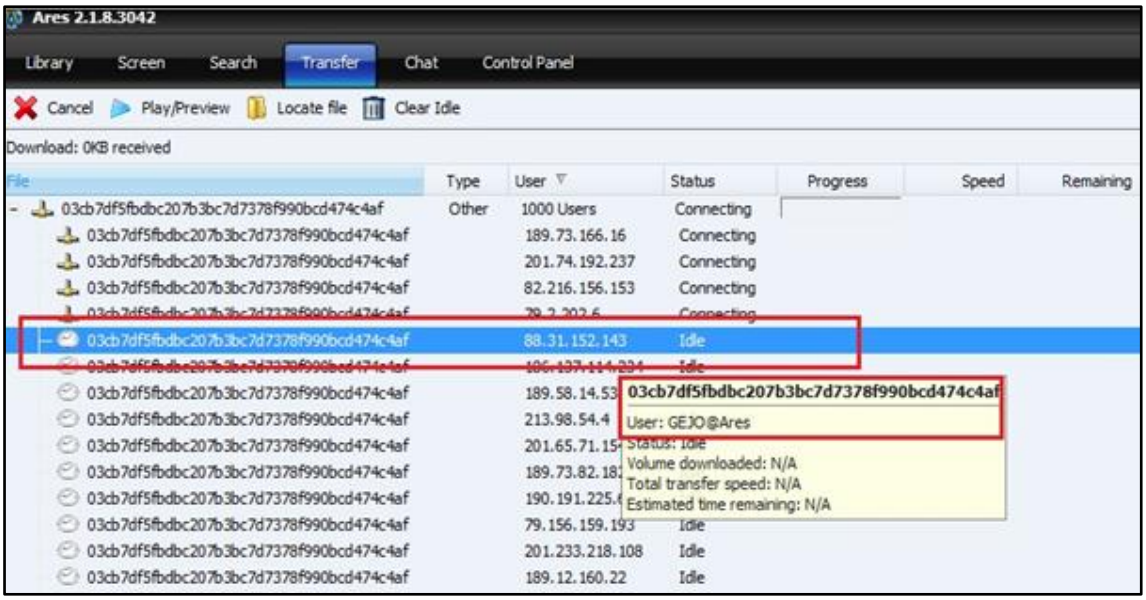

Figura 17: Detalle de descarga en Cliente de Ares modificado

Como podemos observar en las figuras X y X, el usuario “GEJO@Ares” aparece en la lista de fuentes de Ares. Este username también aparece en el paquete devuelto

por los nodos vecinos con los nombres y las IPs de los usuarios que tienen el fichero buscado.

La IP del nodo vecino que nos ha enviado el paquete que contiene al usuario 'GEJO@Ares' es 95.160.161.254. A esta IP le habíamos enviado previamente una petición de búsqueda del fichero, por lo que podemos deducir que cuando nosotros enviamos una petición a nuestros nodos vecinos pidiendo un fichero, nuestros nodos vecinos nos responden con paquetes que contienen los nombres y las IP's codificadas en hexadecimal de usuarios que poseen dicho archivo.

A los usuarios de Ares se les da la posibilidad de escoger un nombre o nickname (el username) al iniciar Ares, si el usuario decide no escoger ningún nombre, se le asignará 'anon_ip codificada@Ares'. La IP está codificada en hexadecimal, así que cuando cambie la IP el nombre de usuario también cambiará. Por ejemplo, en el análisis de tramas anterior vemos a un usuario llamado 'anon_ba8972ea@Ares', si descodificamos la IP en hexadecimal en un conversor hexadecimal a IP (http://sami.on.eniten.com/hex2ip/?) obtenemos la IP: 186.137.114.234.

En la figura X, podemos ver el valor en hexadecimal de la IP de usuario del usuario "GEJO@Ares".

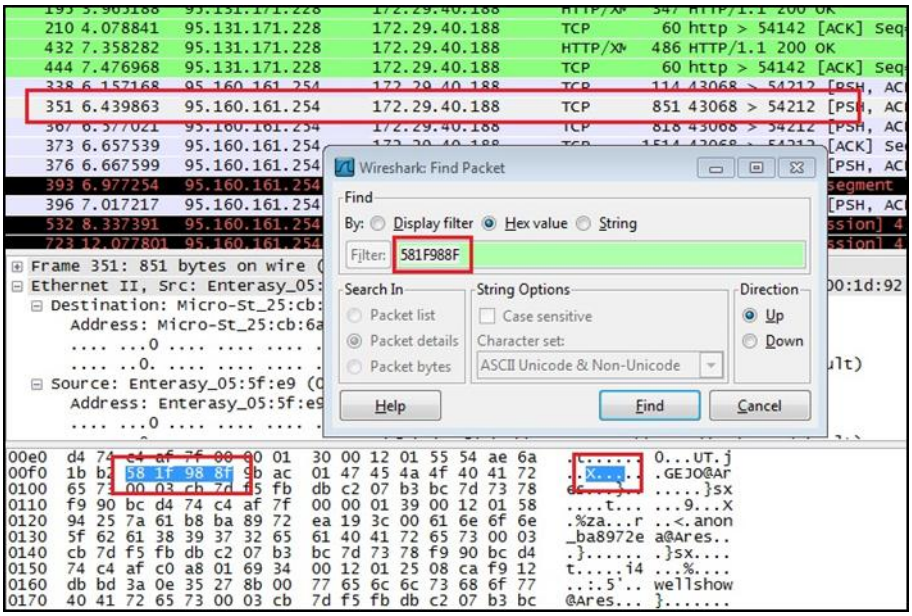

Figura 18: IP en hexadecimal del usuario

La estructura que se repite dentro del paquete es la siguiente:

IP (4 Bytes)	3 Bytes de control	**Nombre de usuario** (1 Byte por carácter)	**. . . } . . . . }sx . . . . t . . . . .**

Esta estructura se repite para cada usuario del paquete y, en cada paquete recibido, aparecen alrededor de 14 usuarios.

Al introducir la IP en hexadecimal del usuario "GEJO@Ares" en un conversor, obtenemos que la IP es **88.31.152.143**.

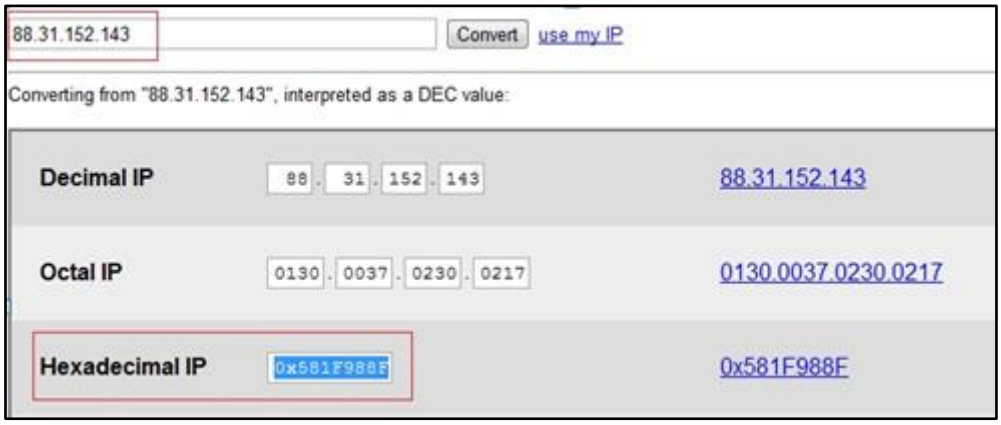

Figura 19: Conversión de la IP en Hexadecimal

3.3 Diseño y creación de la base de datos

La base de datos juega un papel muy importante dentro del sistema, ya que nos permite almacenar los resultados obtenidos durante el funcionamiento del cliente de Ares modificado y, a su vez, nos permite tener un control de los usuarios registrados.

El sistema cuenta con una base de datos MySQL centralizada y gestionada mediante PHPMyAdmin. Ésta se encuentra en el servidor y Ares se conectará a dicha base de datos para obtener los hashes a buscar y escribir sesiones de usuarios que posean dichos hashes. La base de datos está estructurada en tablas, las cuales permiten un acceso concurrente a la información que se encuentra en zonas distintas.

Tablas de la base de datos

Para el correcto funcionamiento del sistema es imprescindible contar con la información de los archivos que queremos buscar, así como de los usuarios a los cuales han sido identificados. La información de los archivos a buscar se sube a la base de datos con un archivo Excel (.xls) que tiene 2 columnas, Hash del archivo y tamaño del archivo. El proceso de subida de este archivo se explicara en el apartado 4.2.5.

Tablas para el registro de datos de ficheros: En ellas se guarda principalmente el hash y el tamaño del archivo a buscar. Estos dos datos son imprescindibles para lanzar las

búsquedas de un archivo en Ares de forma automática. A este grupo pertenecen las siguientes tablas:

- *All_hash*: En esta tabla se guardan todos los ficheros que han sido subidos por un usuario a la base de datos. Se almacenarán el hash y el tamaño del archivo.

- *file_infomation*: En esta tabla se almacena el hash, el tamaño del archivo (en bytes) y un bit que indica si el archivo será el siguiente en ser descargado o no.

Tabla para el registro de datos de usuarios (fuentes/sources): En este tipo de tabla se registran los datos de los usuarios que poseen un archivo de los que estamos buscando con el cliente de Ares modificado. Si un usuario tiene en su ordenador algún archivo de los buscados, se almacena su IP y su username con un ID único (primary key). La tabla encargada de registrar estos datos es *sources_information*.

- *sources_information*: En ella se registra tanto la IP como el nombre de usuario de Ares el cual se le fue detectado uno o varios de los archivos buscados.

Tabla para el registro de datos de sesiones: En este tipo de tablas se establece el registro de datos necesarios para la identificación de las sesiones. En la tabla *sesión_information* se registran:

- File_name: Indica el hash de archivo que un usuario (fuente) tiene guardado en su ordenador.

- Date: Contiene la fecha y hora de la última sesión registrada de dicho usuario y archivo.

- User: Indica que usuario del sistema ha realizado el registro.

- IP: Contiene la dirección IP de la última sesión para dicho archivo y usuario. Este parámetro facilita la localización de IPs dinámicas.

- ID: Identificador único para cada sesión. (Primary Key).

Tabla para indexado de datos: Para aumentar la eficiencia de la base de datos tanto en velocidad como en consumo de recursos, se creado la tabla *sesion_by_ip_by_hash* donde se establece una relación entre **archivo**, **usuario** y **sesión**.

Tabla para la clasificación de IPs: La tabla *Ipdatabase* contiene los rangos de IPs asociadas a cada país. De esta forma podemos relacionar la IP de un usuario con su localización.

La base de datos permite la centralización de los datos tanto para lectura como para escritura de tal forma que tanto la aplicación web como Ares se conectan a ella con el fin de almacenar o recuperar información de la misma.

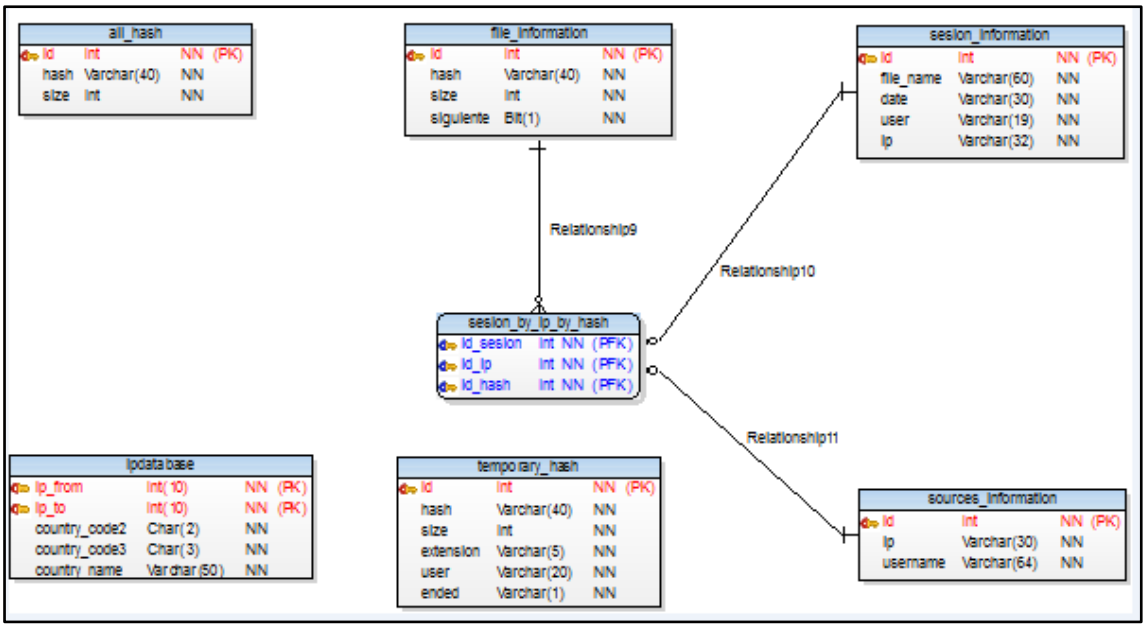

Figura 20: Esquema de la base de datos "ares"

3.4 Ares

3.4.1 Compilación

Lo primero que tenemos que hacer es descargar el código libre del cliente de Ares 2.1.8 desde el enlace:

http://sourceforge.net/projects/aresgalaxy/files/aresgalaxy/AresRegular218_020212/

El proyecto de Ares y los archivos .pas se abren con la aplicación Borland Delphi 7, que nos permite modificar el código de forma cómoda y compilar en dos modos, *Debug* y *Release*. El modo *Debug* permite la depuración de la aplicación mediante el rastreo de errores y crea un ejecutable con dependencia de librerías del ordenador, por lo que no es portable a otros ordenadores que no dispongan de Borland Delphi 7.

El modo *Release*, que es el modo en el que se va a compilar este TFG, crea un ejecutable de la aplicación que puede ser ejecutada en cualquier ordenador.

Una vez modificado el código del cliente de Ares, explicado en el apartado 3.4.2, se debe compilar para obtener el archivo ejecutable correspondiente al cliente de Ares modificado.

Para la compilación, se necesitan los siguientes paquetes externos:

- Borland Delphi 7 Second Edition [9]
- Componentes ActiveXs (adobe Flash Player y Adobe ShockWave Player) [7]
- MySQL Connector ODBC [13]
- Paquete ESBPCS [14]
- Paquete DSPACK231 [10]
- Componente TntWareDelphiUnicode [15]
- Librerías JCL y JVCL [12]
- Paquete EmbebedWB [11]
- Ares VCL's

Después de conseguir todos los componentes y entornos necesarios, creamos en el directorio de la aplicación una carpeta llamada *lib* que tenga todas las librerías de estos componentes.

Una vez que se ha preparado el directorio, empezamos con la instalación e importación de librerías.

Componentes ActiveXs

Empezamos instalando los componentes "ActiveXs", los cuales los podemos descargar desde el link," http://www.adobe.com/es/downloads/", como vemos en la figura 21.

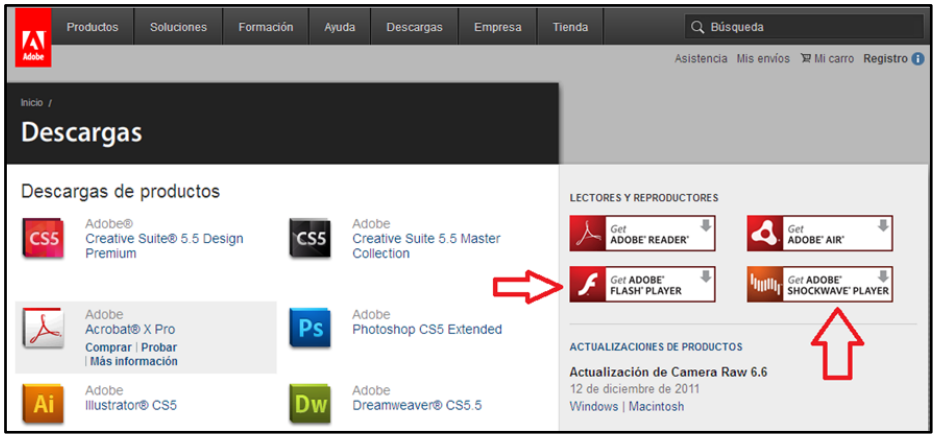

Figura 21: Captura componentes ActiveX

Paquete ESBPCS

El siguiente paquete que hay que instalar es "ESBPCS". Para ello, lo único que tenemos que hacer es ejecutar el Setup.exe, adjuntado en el cd del proyecto.

Instalación MySQL Connector ODBC

Este componente es indispensable, ya que es el que permite la conexión del cliente de Ares modificado con la base de datos.

En primer lugar, ejecutamos el instalador adjuntado en este TFG y llamado "mysql-connector-odbc-5.1.10".

Una vez instalado, abrimos el menú de *Inicio* de Windows, accedemos al *Panel de control* y abrimos *Herramientas administrativas*.

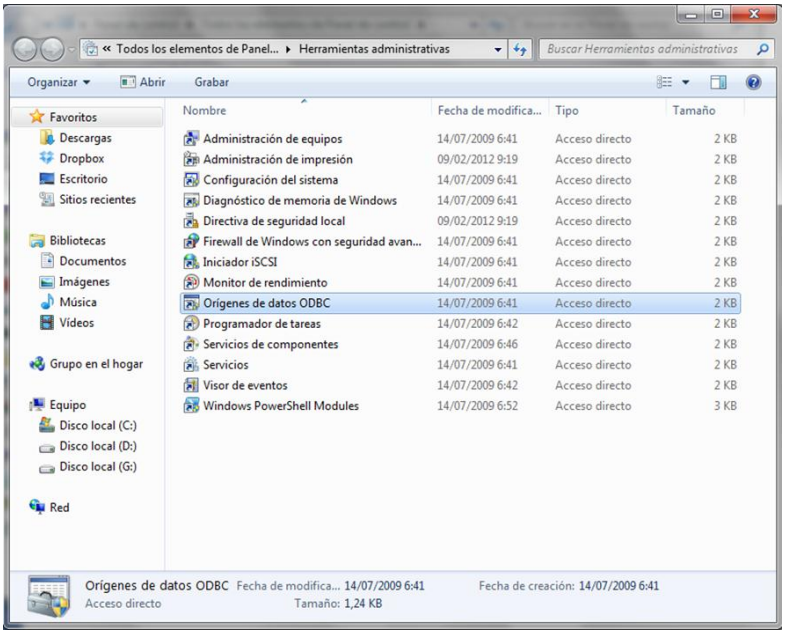

Figura 22: Vista de herramientas administrativas

Estando situado en herramientas administrativas, abrimos "orígenes de datos ODBC", como se muestra en la figura 23, para configurar la conexión.

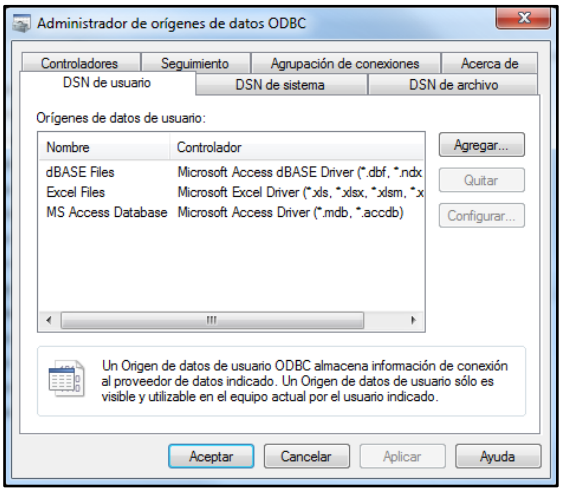

Figura 23: Ventana configuración administrador ODBC

Con la ventana de la figura 23 abierta, hacemos clic en "Agregar" y seleccionamos como origen de datos "MySQL ODBC 5.1 Driver".

Por último, hacemos clic en finalizar y rellenamos los datos de conexión con los valores que aparecen en la tabla 2.

Campo	Valor
Data Source Name	ares
TCP/IP Server	127.0.0.1
Port	3306
User	root
Password	TFG2013
Database	ares

Tabla 2: Datos de conexión

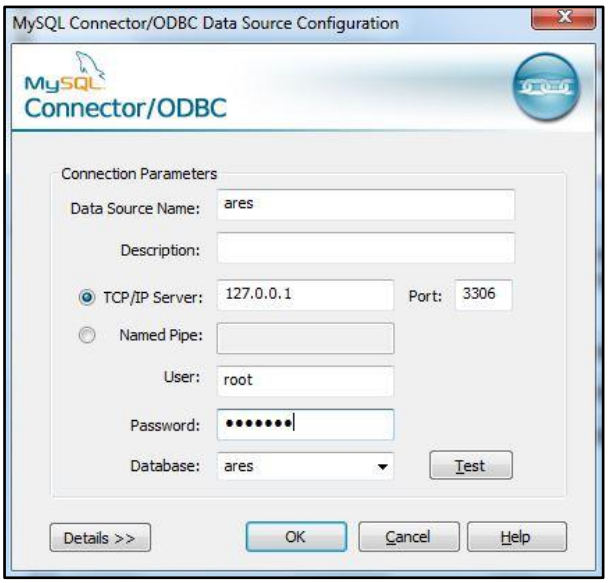

Figura 24: Vista con los valores introducidos

Para termina hacemos clic en "OK" para guardar la configuración de la conexión.

Componente TntWareDelphiUnicode

El componente TntWareDelphiUnicode será el siguiente en ser instalado, para ello ejecutamos el Setup.exe.

Una vez instalado, abrimos el proyecto con Borland Delphi 7 y en el menú principal hacemos clic en *Tools*. Una vez hay seleccionamos *Enviroment Options* y hay hacemos clic en *Library*. En el apartado *Directories* nos tenemos que asegurar de que aparece la librería, tal y como aparece en la figura 25.

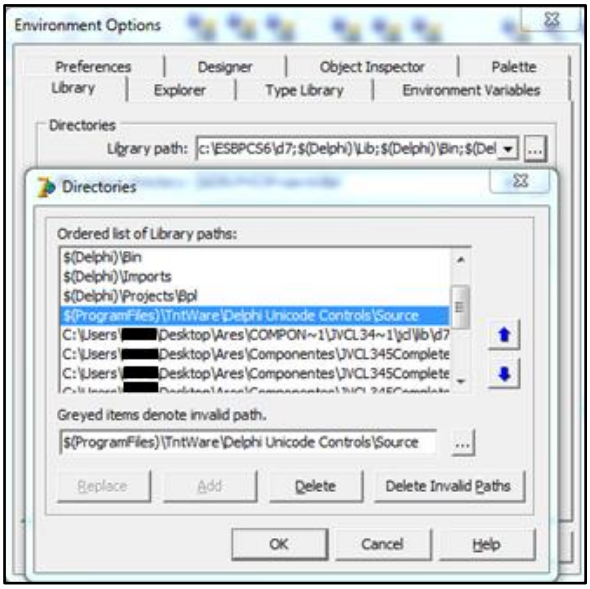

Figura 25: Ventana de directorios del proyecto

Instalación de JCL y JVCL

Para la instalación de ***JCL*** accedemos a la carpeta **JVCL345CompleteJCL221-Build4197/jcl**, y ejecutamos install.bat. A continuación se abrirá el proceso de instalación del componente.

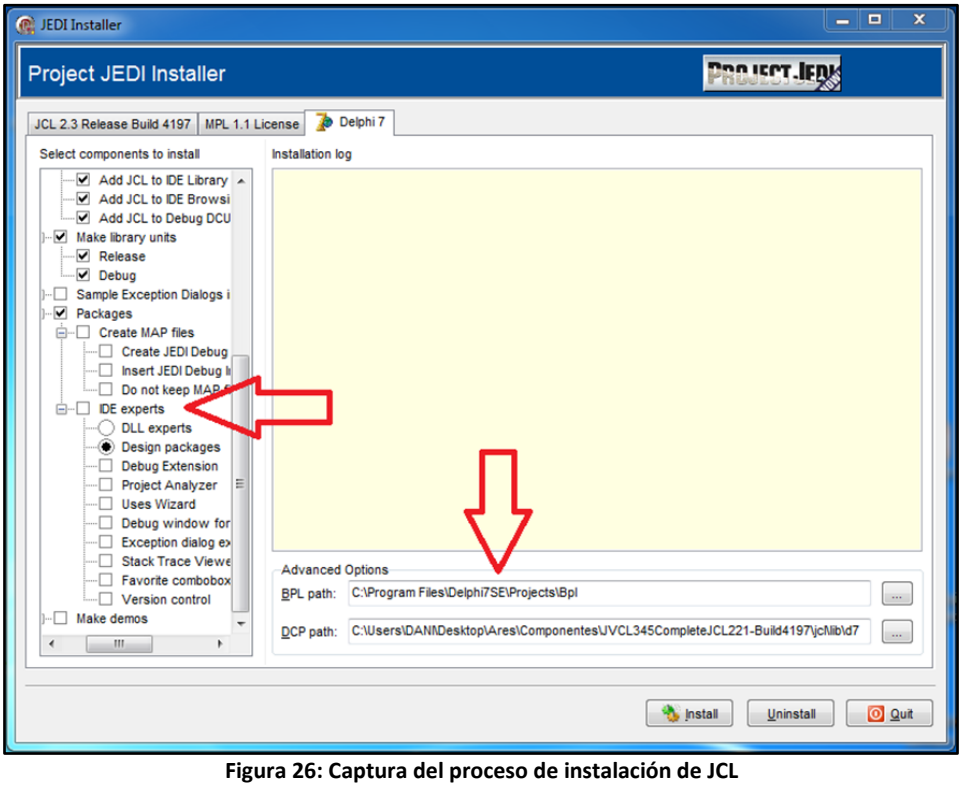

Figura 26: Captura del proceso de instalación de JCL

Como se puede observar en la figura 26, tenemos que deshabilitar la opción ***IDE experts*** y escoger la ruta adecuada similar a las que aparecen en pantalla. Previamente hemos tenido que aceptar los términos de licencia para poder instalarlo.

Hacemos clic en "Install" y aparecerá una ventana emergente como la de la figura 27, en la que tendremos que hacer clic en "No".

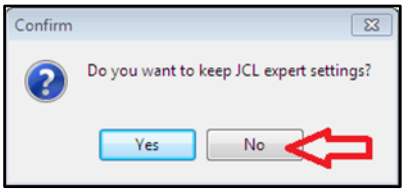

Figura 27: Ventana emergente JCL

A continuación, instalamos el componente JVCL.

Para ello accedemos al directorio **JVCL345CompleteJCL221-Build4197/jvcl**, y ejecutamos Install.bat. A continuación se abrirá el proceso de instalación del componente.

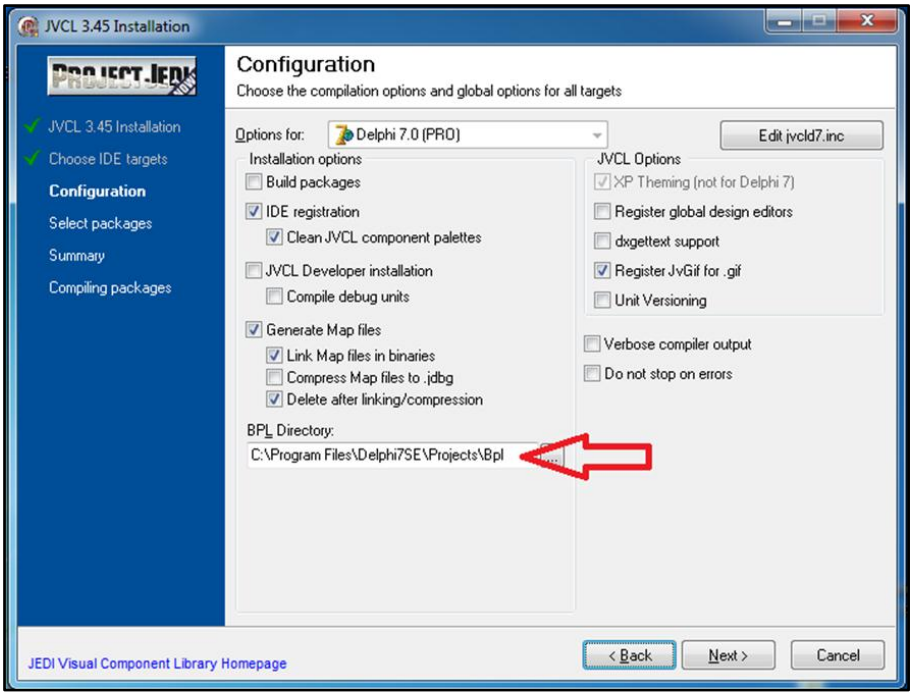

Figura 28: Captura del proceso de instalación de JVCL

Tenemos que tener en cuenta que la ruta del directorio DPL tiene que ser parecida a la de la figura 28. Por último, hacemos clic en "Next" y se completará la instalación de este componente.

Paquete "DSPACK231"

Abrimos la carpeta DSPACK231/packages, adjuntada en el TFG y hacemos lo siguiente:

- Abrimos DirectX9_D7.dpk y lo compilamos.
- Abrimos DSPack_D7.dpk y lo compilamos.
- Abrimos DSPackDesign_D7.dpk lo compilamos e instalamos.

Después, abrimos Borland Delphi 7 y en el menú principal hacemos clic en *Tools*. Una vez hay seleccionamos *Enviroment Options* y hay hacemos clic en *Library*. Una vez aquí añadimos las siguientes rutas tanto en **Library Paths** como en **Browsing Paths**:

- C:\Users\Usuario\Desktop\Ares\Componentes\DSPACK231\src\DirectX9

- C:\Users\Usuario\Desktop\Ares\Componentes\DSPACK231\src\DSPack

Paquete EmbebedWB

Copiamos la carpeta ***EmbeddedWB_D5-XE_Version_14.70.0***, adjuntada en el TFG, a una ruta similar a "C:\Program Files\Delphi7SE\Lib". Por último, abrimos el archivo con ruta "Packages\ EmbeddedWebBrowser_D7.dpk".

Lo compilamos e instalamos. Al cerrar seleccionamos que **NO** guarde los cambios.

Importación del paquete ShockWave ActiveX al entorno Borland Delphi 7

Abrimos el proyecto con el entorno Borland Delphi 7 y hacemos clic en *Component*. En el menú desplegable, hacemos clic *Import ActiveX Control*.

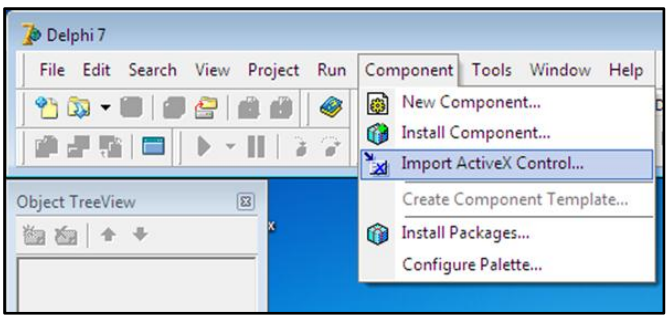

Figura 29: Localización del menú Component

Seleccionamos **Shockwave Flash (Version 1.0)** y le damos a **Install**:

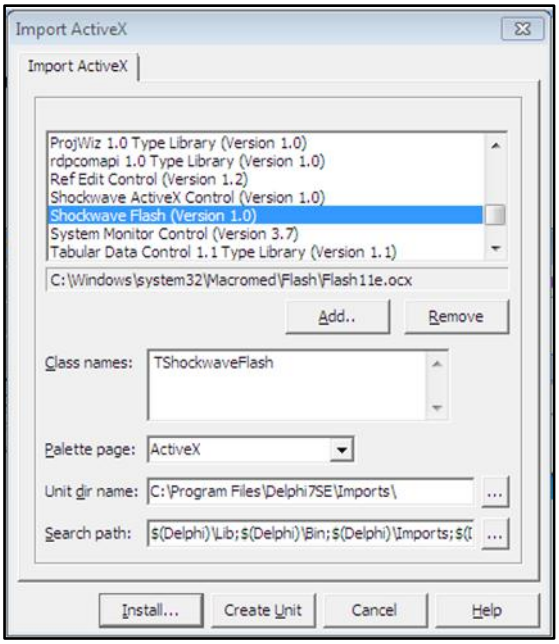

Figura 30: Ventana de importación de componente ActiveX

A continuación, le damos un nombre al paquete, por ejemplo *Shockwave* y hacemos clic en "**OK"**.

Por último, hacemos clic en "**SI"** para que se realice *built* del nuevo paquete importado y guardamos los cambios.

Comprobación de rutas

Abrimos el proyecto con Borland Delphi 7 y en el menú principal hacemos clic en *Tools*. Una vez hay seleccionamos *Enviroment Options* y hay hacemos clic en *Library*.

Una vez aquí, comprobamos que en **Library Path** aparecen las rutas de la figura 31 o similares:

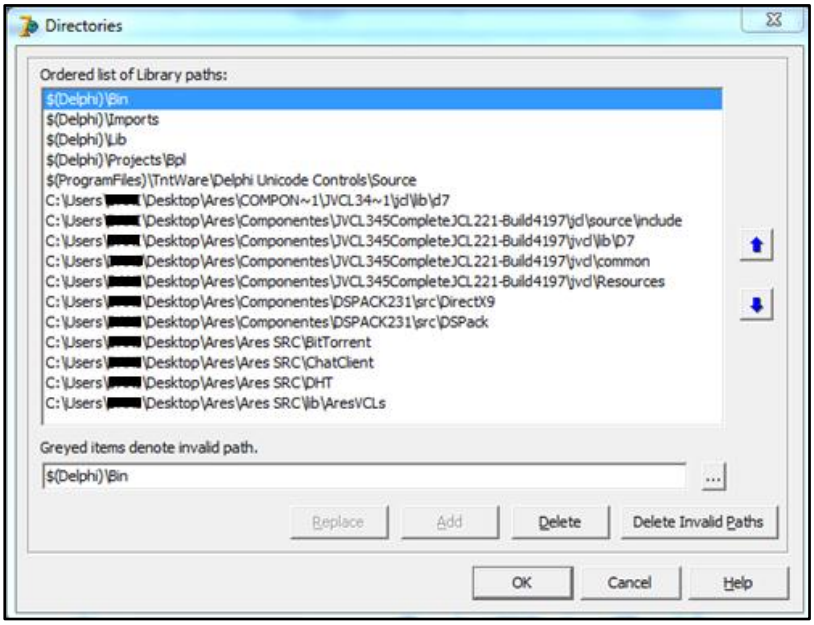

Figura 31: Rutas en Library Path

Y comprobamos que en **Browsing Path** aparecen las rutas de la figura 32 o similares:

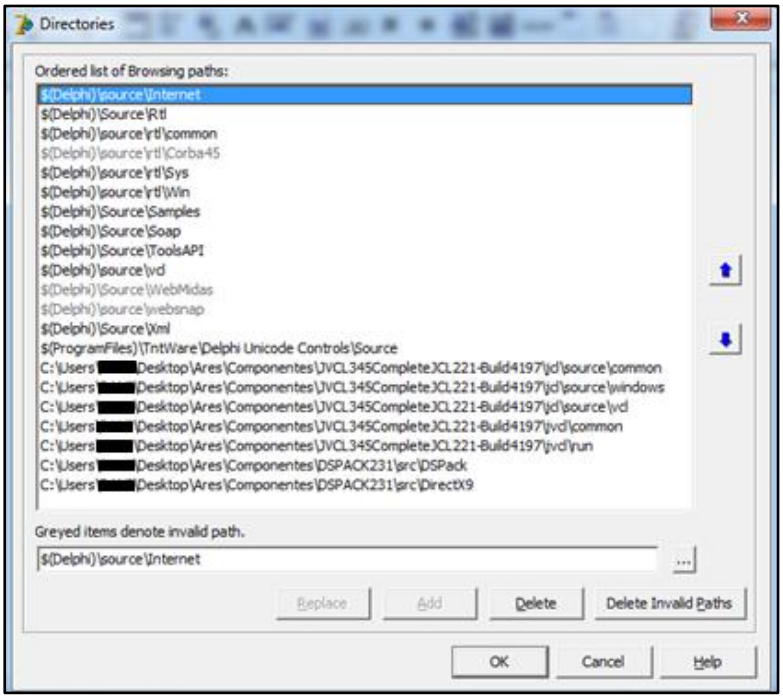

Figura 32: Rutas en Browsing Path

Instalación de Ares VCL's

Para la instalación de este componente, nos vamos a la ruta "Componentes\03 – Ares VCLs (modificadas)". Este directorio ha sido adjuntado en el TFG.

Una vez en este directorio, compilamos e instalamos **arescp.dkp**.

Compilación y obtención del ejecutable

El último paso es compilar el proyecto de Ares. Para ello, abrimos el proyecto situado en la ruta "Ares SRC\Ares.dpr" con el entorno Borland Delphi 7.

Seleccionamos como tipo de compilación ***Release*** y compilamos.

El archivo ejecutable, llamado **Ares.exe**, estará ubicado en la carpeta Ares SRC, siempre y cuando no haya habido errores de compilación.

3.4.2 Modificaciones realizadas

Para conseguir que la aplicación realice las funcionalidades explicadas en anteriores capítulos, se han realizado varias modificaciones en el código fuente de Ares 2.1.8, las cuales se detallan a continuación.

- ***Thread_download.pas:*** Esta clase crea un hilo por cada fichero en la lista de descargas en Transfer. En esta clase se realizan diversas modificaciones relacionadas con la descarga de los archivos buscados y el registro de las IPs en la base de datos.

  - ***tthread_download.check_half_second:*** En este procedimiento comentamos la línea //CheckSources; para evitar que los ficheros a descargar en la pestaña 'Transfer' comiencen a descargar. Evitamos que la descarga inicie debido a que en Ares, un usuario que tiene una parte descargada de un fichero puede actuar como servidor a su vez de más clientes, por lo que cortaremos la descarga en cuanto los nodos servidores estén listos para comenzar con nuestra descarga.

  - ***tthread_download.LeerBaseDatosTxt:*** Este procedimiento ha sido añadido para permitir la conexión a la base de datos. El procedimiento se encarga de leer un fichero donde se detalla el nombre del conector de base de datos, el usuario y la contraseña con la que poder conectarse a esa base de

datos. Después realiza una conexión con la base de datos especificada mediante 'ADOQuery'.

- ***save_sources_bd:*** Este procedimiento ha sido añadido para registrar en la base de datos el nombre de usuario, la IP encontrada, la fecha, el usuario y el hash pasado por parámetro. Este procedimiento comprobará si en la base de datos se encuentran registros similares, si es así los actualiza y si no los inserta. El procedimiento será llamado en el momento en que se establezca conexión para descargar un fichero con los demás nodos servidores de dicho fichero.

- ***tthread_download.AddVisualNewSources:*** Este procedimiento es el que obtiene las IP's de los nodos servidores de un archivo en concreto. El método ha sido modificado para que calcule la fecha en el momento en el que un archivo vaya a comenzar una descarga y llame al procedimiento anteriormente descrito 'save_sources_bd' para registrar la información asociada a la descarga (username, IP, hash, fecha y usuario).

- ***tthread_download.parser_nickname:*** Esta función ha sido añadida para formatear un string pasado por parámetro (que normalmente será el username) y devolverlo en un formato válido para realizar correctamente la sentencia SQL.

- ***tthread_download.formateaString:*** Esta función ha sido añadida para concatenar con comillas a ambos lados una cadena introducida por parámetro.

- ***Ufrmmain.pas:*** Esta clase es la clase principal de Ares y se ocupa de cargar periódicamente nuevos hashes a la lista de descargas en transfers accediendo a la base de datos de ares.

  - ***tthread_download.LeerBaseDatosTxt:*** Este procedimiento ha sido añadido para permitir la conexión a la base de datos. El procedimiento se encarga de leer un fichero donde se detalla el nombre del conector de base de datos, el usuario y la contraseña con la que poder conectarse a esa base de datos. Después realiza una conexión con la base de datos especificada mediante 'ADOQuery'.

  - ***Tares_frmmain.cargadorHash:*** Este procedimiento ha sido añadido para poder cargar hashes periódicamente a la pestaña transfer de Ares desde la base de datos leída en el método 'LeerBaseDatosTxt'. El procedimiento cargará la cantidad de hashes especificada en la variable

'cantidad_buscados' (establecida a 5 hashes) cada periodo de tiempo (definido por el reloj Timer1Timer).

- ***Tares_frmmain.Limpiarlista:*** Este procedimiento ha sido añadido para limpiar la lista de hashes a descargar en la pestaña transfer. Este método se ejecuta a cada pulso de reloj para limpiar la lista de hashes buscados y cargar nuevos hashes mediante el método 'cargadorHash'.

- ***Tares_frmmain.Timer1Timer:*** Este reloj ha sido añadido para cargar nuevos hash a la lista de transfer. Para ello utiliza los métodos ClearIdle2Click, ClearIdle1Click, cargadorHash y LimpiarLista.

- ***Tares_frmmain.Timer2Timer:*** Este reloj ha sido añadido para comprobar constantemente que la lista de archivos buscados en transfer esté limpia y no queden archivos cancelados sin eliminar. Para ello utiliza los métodos ClearIdle2Click y ClearIdle1Click.

Varias de las modificaciones comentadas anteriormente aseguran que el proceso de descarga del archivo no se lleve a cabo. El cliente de Ares modificado busca a los usuarios portadores del archivo, y obtiene sus IPs, pero no descarga el archivo.

De esta forma, nos aseguramos de no aparecer en la base de datos, ya que si tenemos alguno de los archivos que estamos buscando nos convertimos automáticamente en sospechosos y apareceremos en la base de datos, lo que crearía cierta confusión.

3.4.3 Diagrama de comunicación entre Ares y el servidor web

En este apartado, se va a describir la comunicación entre el cliente de Ares modificado y el servidor web, en el cual esta albergada la base de datos.

Una vez que iniciamos el cliente de Ares, este se conecta a la base de datos para descargar 5 hashes de la base de datos. Una vez descargados, procederá a realizar la búsqueda de estos archivos por la red de Ares durante varios minutos.

Una vez transcurridos estos minutos, el cliente de Ares modificado realiza una serie de consultas para saber dónde alojar el registro de las nuevas fuentes. Para ello primero recupera la sesión actual sobre la que se está trabajando, para saber si alguna de las fuentes nuevas que va a registrar ya existían en la base de datos. A continuación, recupera las fuentes de dicha sesión. Por último realiza el registro de las nuevas fuentes obtenidas en la búsqueda, teniendo en cuenta si ya existían o no.

Este proceso se puede ver con más claridad en el diagrama de comunicación de la figura 33.

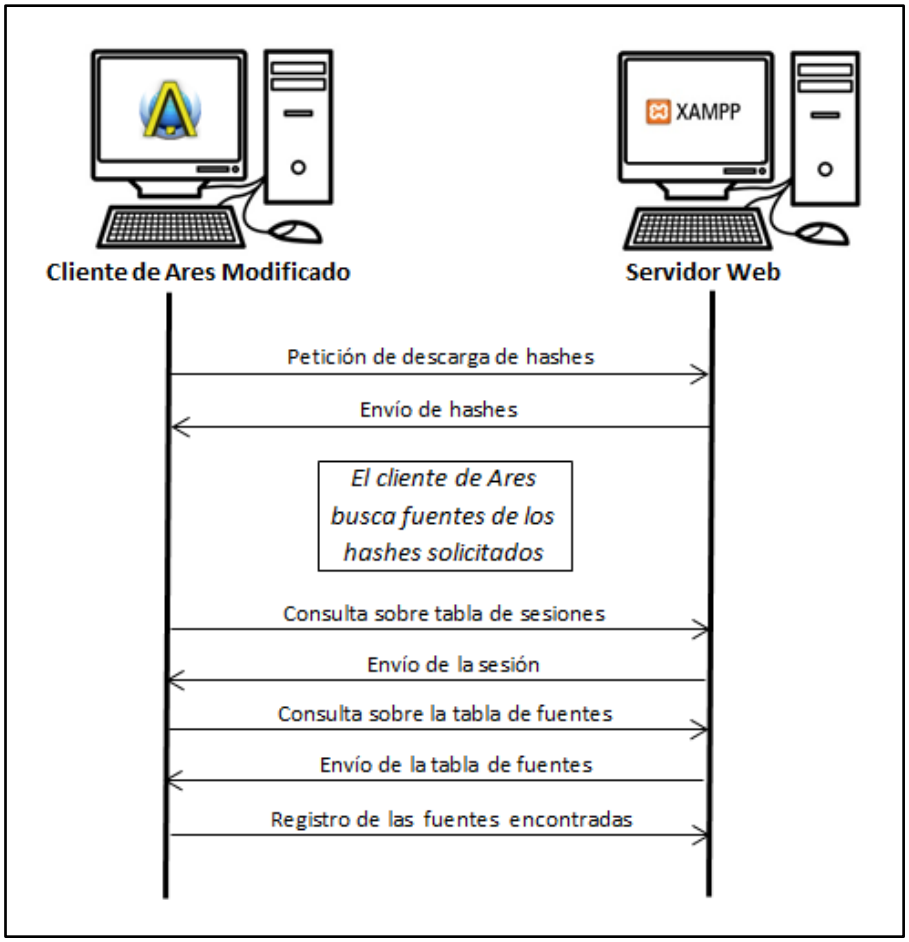

Figura 33: Diagrama de comunicación entre Ares y el servidor web

Una vez realizado el registro de fuentes, se descartan los hashes buscados y el cliente de Ares se descarga los 5 siguientes hashes, volviendo a repetir el proceso de comunicación visto en la figura 33.

3.5 Servicio web

El servicio web consta de una interfaz web, y su función principal es intermediar entre la base de datos y el usuario administrador del sistema. Dicha interfaz permite la visualización y gestión de la información almacenada en la base de datos, así como la modificación de ciertos parámetros y contenidos.

La figura 34 muestra la apariencia del interfaz web, en la que podemos ver el formulario de inicio de sesión y las distintas funcionalidades que ofrece distribuidas en pestañas.

Figura 34: Interfaz inicial del servicio web

La apariencia ha sido creada a través de la aplicación de retoque de imágenes Adobe Photoshop, HTML y PHP. Se ha elegido la distribución en pestañas por ser la más sencilla de utilizar para los usuarios, ya que permite cambiar rápidamente lo que se está viendo en pantalla sin cambiar de ventana.

El servicio web posee varios ficheros PHP que son ejecutados por el servidor según se van necesitando. El primer *script* que se ejecuta es "Gestion_bd.php", que es el fichero principal y a partir del cual podemos navegar por toda la interfaz. Cada fichero PHP ha sido programado para que verifique la autenticación del usuario antes de proceder a realizar cualquier otra acción.

3.5.1 Control de acceso

Todas las funcionalidades que se ofrecen desde la interfaz web están ubicadas en el servidor web central pero para, tener acceso a ellas, es necesario estar previamente registrado como administrador del sistema. Para asegurarnos de que el sistema es manipulado únicamente por el personal autorizado, se ha añadido un control de acceso de usuarios. Con esto conseguimos que solo los usuarios autorizados puedan acceder al servicio web.

El control de usuarios se realiza mediante sesiones. En las aplicaciones web creadas con PHP, las sesiones sirven para almacenar toda la información relativa a un

usuario mientras dura su navegación por la página web. Mientras un usuario navega por la página web, la sesión abierta almacena los valores de las variables de la página.

Cada usuario que realiza un *login* en el servicio web abre una sesión, y esta sesión es independiente de la sesión de otros usuarios. El formulario de acceso se puede ver en la figura 35.

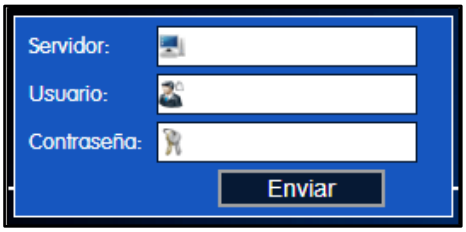

Figura 35: Formulario de acceso

En la sesión de un usuario se almacena su nombre de usuario y su contraseña de acceso. Estas credenciales de usuario deben coincidir con las credenciales de acceso a la base de datos.

Por último, para que cada usuario conserve el identificador de sesión, PHP ha sido configurado para que la variable de sesión sea almacenada en forma de cookie.

Para aumentar el grado de seguridad, se ha establecido un periodo de validez de la sesión. El tiempo máximo de duración de una sesión es 15 minutos desde que se realiza el *login*. Una vez transcurrido este tiempo, si el usuario quiere seguir navegando por la aplicación deberá iniciar sesión nuevamente.

3.5.2 Pestaña 1: Crear base de datos

La creación de las tablas y las relaciones de la base de datos se hace de forma automática desde esta pestaña. Solo el administrador de la base de datos puede realizar esta operación. Para la creación de la base de datos se requiere introducir un valor numérico para la variable *maxfiles*. Esta variable representa el número máximo de ficheros que puede tener un usuario sin que sea considerado sospechoso.

Figura 36: Interfaz de la pestaña "Crear base de datos"

Cuando se hace clic en el botón “Crear base de datos…”, se realiza una llamada a un fichero PHP llamado *crear_BD.php*, que contiene las consultas que se le envían a MySQL, para la creación de las tablas y las relaciones entre ellas. Una vez ejecutadas las consultas encargadas de crear la base de datos, se ejecuta otro fichero SQL que introduce los rangos de las IPs por país en la tabla “Ipdatabase”.

EL lenguaje PHP admite consultas en lenguaje SQL gracias a las extensiones del propio PHP.

La base de datos se actualiza si ya existía previamente. También se puede modificar el valor de la variable *maxfiles*, de tal forma que si la base de datos ya existía y elegimos un nuevo valor para la variable, al ejecutar la acción de crear, se mostrara el mensaje de existencia de la base de datos y la actualización del valor *maxfiles*.

3.5.3 Pestaña 2: Subir fichero

En la pestaña “Subir fichero” se realiza una función fundamental para el funcionamiento del sistema, la carga de hashes en la base de datos. En esta pestaña se sube a la base de datos el archivo Excel que contiene los hashes y tamaños de los archivos a buscar y es indispensable para el funcionamiento del cliente de Ares modificado.

Figura 37: Interfaz de la pestaña "Subir fichero"

Solo el usuario administrador, previamente registrado, puede subir este archivo, que tiene que tener una apariencia similar al de la figura 38.

	A	B
1	7EFCC90A43E1C72A4D839A6A9C5C323470CABEE1	7.658.156
2	F2A5F1DB6BA0F16B7BD102411C8059054DEE3409	3.116.766
3	C349B4D00FAF8BC27F2B2FD66A7B416D78476DA5	34.449.426
4	B498668832A2783E8B8E3B0BF7DD87A4517F5A1F	8519680
5	B5D9DFE2B4B672BC67E338EE36C2855A191786EF	3343133
6	BF7844490C4E5DCAF89302A957D4F530A4FD18D1	8516866
7	7761EA54FFE963876AB517F6B6A532AE6219A35A	5.168.354
8	E9451DDFF7770F4F2970759113B4432382923915	3.070.336
9	FF9E3F03284BE64A403511BBF6F286D57AD469F8	6.045.368
10	A776F8BD30923C29B95851D11E9B2450F6468C02	8387359

Figura 38: Formato del archivo Excel de hashes

Además del hash, el cliente de Ares necesita el tamaño de los archivos para realizar la búsqueda, como se puede ver en la columna B de la figura 38.

Por último, en esta pestaña también podemos añadir un archivo con nuevos hashes o actualizar los actuales. Si al subir un fichero Excel se detecta que la tabla *All_hash* ya tenía hashes, se añaden los nuevos hashes a la base de datos sin borrar los anteriores. El fichero encargado de la carga y actualización de los hashes es "*Actualizar_BD.php*".

3.5.4 Pestaña 3: Descarga Base_datos.txt

Para que el cliente de Ares modificado funcione necesita poder conectarse con la base de datos. Para ello, necesita un archivo llamado "Base_datos.txt" que contiene

el nombre de la base de datos en el conector MySQL ODBC, el nombre de usuario administrador y la contraseña del usuario administrador.

El archivo “Base_datos.txt” lo descargamos en esta pestaña, haciendo clic en el icono de descarga, que se puede apreciar en la figura 39.

Figura 39: Interfaz de la pestaña "Descargar Base_datos.txt"

Para generar el archivo, se ha realizado un fichero llamado “*fichero_cong.php*” que es ejecutado cuando hacemos clic en el icono de descarga de la pestaña.

El formato de dicho archivo lo podemos ver en la figura 40.

```
Conector_ODBC:ares
usuario:root
Contraseña:TFG2013
```

Figura 40: Formato del archivo "Base_datos.txt"

El archivo “Base_datos.txt” se descargará en la carpeta de descargas establecida por defecto en el navegador web. Cuando lo tengamos descargado, hay que situarlo en el mismo directorio que el archivo ejecutable del cliente de Ares, llamado “Ares.exe”, ya que el programa lo buscara en su mismo directorio.

3.5.5 Pestaña 4: Mostrar resultados

La última pestaña, nos permite gestionar y mostrar de forma estructurada la información registrada por el cliente de Ares modificado en la base de datos del servidor web. Esta funcionalidad nos permite visionar la información en formato Web y en formato PDF.

Figura 41: Interfaz de la pestaña "Mostrar Resultados"

El formato web muestra la información a través de la interfaz, como se puede apreciar en la figura 42. Los datos aparecen en una tabla que contiene los campos IP del usuario, número de archivos encontrados a este usuario, nombre del usuario en la red de Ares y el país al cual pertenece la IP.

Esta funcionalidad ha sido implementada completamente en PHP y las consultas a la base de datos se han realizado en lenguaje SQL, utilizando la API de PHP para este lenguaje. El país al cual pertenece la IP se obtiene cruzando el valor de la IP del usuario con la tabla “IPdatabase”.

Información General Base de Datos Ares

Existen 31 ficheros en la base de datos

Se han detectado 13985 usuarios diferentes

IP	ARCHIVOS ENCONTRADOS	NOMBRE DE USUARIO	ORIGEN
189.115.18.5	1	raulvianna@Ares	BRAZIL
189.200.193.252	4	anon_bdc8c1fc@Ares	MEXICO
186.251.23.43	2	acontece@Ares	BRAZIL
92.58.92.71	1	anon_5c3a5c47@Ares	SPAIN
177.3.101.145	2	yaas@Ares	COLOMBIA
80.102.16.52	2	lunallena1411@Ares	SPAIN
189.101.252.31	2	anon_bd65fc1f@Ares	BRAZIL
190.193.66.198	2	anita@Ares	ARGENTINA
189.1.177.122	1	OctavioRibeiro7@Ares	BRAZIL
200.127.34.222	1	anon_c87f22de@Ares	ARGENTINA
189.250.236.43	1	anon_bdfaec2b@Ares	MEXICO
201.248.102.100	2	aguila@Ares	VENEZUELA
190.72.26.64	1	anon_be481a40@Ares	VENEZUELA
201.252.44.195	2	habi@Ares	ARGENTINA
201.254.87.123	2	anon_c9fe577b@Ares	ARGENTINA
190.206.23.211	1	anon_bece17d3@Ares	VENEZUELA

Figura 42: Vista Web de la información

Para el formato PDF, se ha utilizado la librería *fpdf* que nos permite exportar en formato PDF la información extraída desde la base de datos mediante la utilización de consultas SQL.

En ambos formatos, ya sea Web o PDF, solo aparecerán los usuarios considerados sospechosos, es decir, los usuarios que tengan un número de archivos mayor o igual al valor introducido en la variable "*maxfiles*".

Capítulo 4

Manual de usuario

En este capítulo se explica el proceso de instalación del servidor web, la configuración previa que se debe realizar para su correcto funcionamiento y el uso de las distintas funcionalidades que ofrece la aplicación a los usuarios finales.

Este manual está dividido en 7 partes: Instalación y configuración del servidor, pasos previos a la ejecución de Ares, funcionalidades de la aplicación, gestión de archivos, los distintos tipos de visionado de los resultados recogidos por la aplicación, resolución de problemas y dudas más frecuentes y un glosario para la explicación de los términos más técnicos.

4.1 Instalación del servidor

Como se ha explicado en anteriores capítulos, un componente indispensable de esta aplicación desarrollada es el servidor web. Las principales funciones de este componente son almacenar de forma estructurada la información recogida por la aplicación y realizar las funciones propias de un servidor web, permitiéndonos acceder a una interfaz creada en HTML y PHP desde la que podremos crear la base de datos, visualizar los resultados y realizar el resto de tareas de administración del sistema.

El servidor del sistema está basado en la herramienta XAMMP (ver apartado 2.6.1), el cual dispone de todos los componentes y funcionalidades requeridas para la realización de este TFG.

Debido a las múltiples incompatibilidades entre componentes, versión que debemos instalar es la 1.7.4 ya que permite el correcto funcionamiento de Apache server y MySQL.

Al iniciar la instalación, seleccionamos como destino de la instalación la ruta “c:/xampp”.

El siguiente paso es elegir las opciones de instalación, por lo que seleccionaremos instalar Apache server, MySQL y Filezilla como un servicio. Con esto

conseguimos que cada vez que iniciemos el ordenador se inicialicen los servicios de todos los componentes.

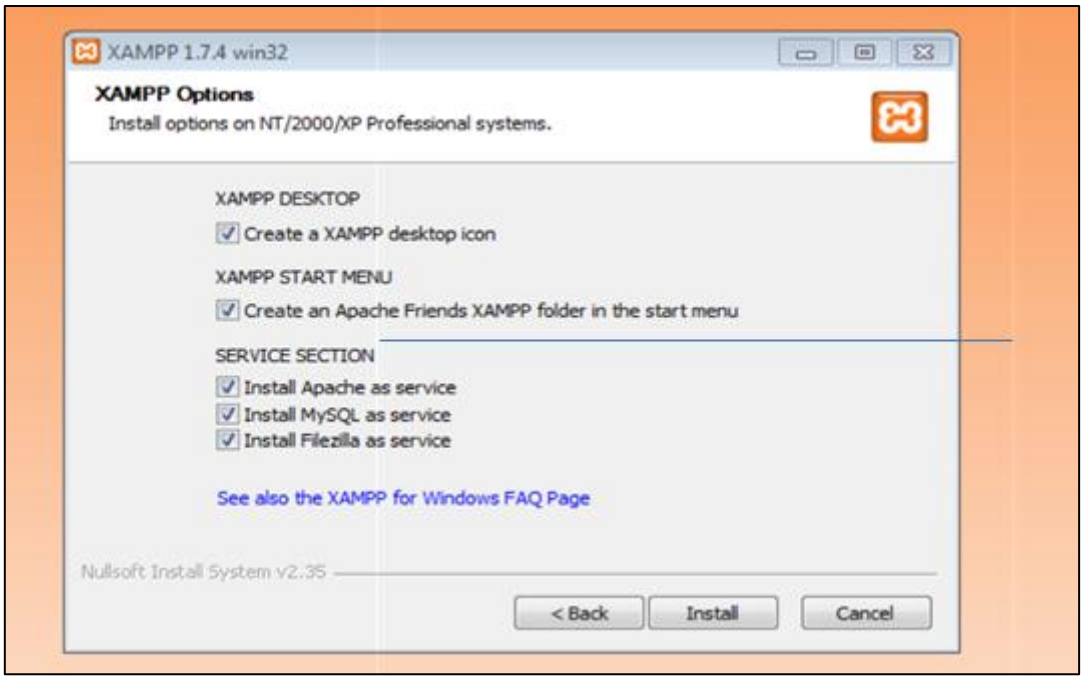

Figura 43: Pantalla de instalación de XAMPP

Presionamos el botón Install y esperamos a que termine la instalación de todos los componentes. Una vez finalizada, comprobara el funcionamiento de todos los servicios instalados y terminara por completo.

4.2 Pasos previos

Antes de poner en funcionamiento el cliente de Ares modificado, hay que llevar a cabo una serie de pasos que mejoraran la seguridad y que permitirán el correcto funcionamiento de todos los componentes del sistema.

4.2.1 Inicialización del servidor Apache y de MySQL

Para que el sistema funcione, el servidor Apache y MySQL deben estar activos, de lo contrario la interfaz web, la base de datos y el cliente de Ares modificado no estarán conectados entre sí.

Para iniciar ambos componentes tenemos que abrir el XAMPP Control Panel (Panel de control de XAMPP), desde donde se pueden iniciar y parar todos los servicios relacionados con la aplicación.

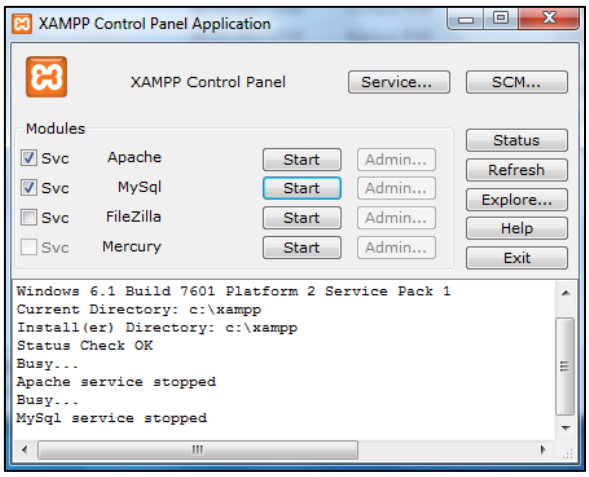

Figura 44: Panel de control de XAMPP

Una vez abierto el panel de control, hacemos clic sobre los botones Start correspondientes a Apache y MySQL. Si todo funciona correctamente, aparecerá el mensaje "Running" al lado de cada uno de los componentes, como se muestra en la figura 45.

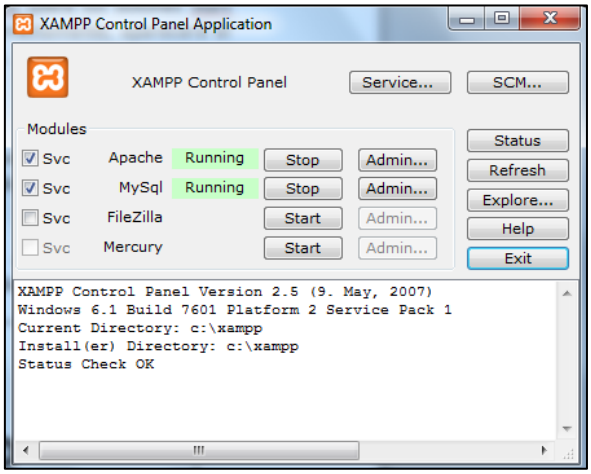

Figura 45: Panel de control de XAMPP con Apache y MySQL funcionando

Para comprobar el correcto funcionamiento del servidor web, accedemos a la dirección loopback introduciendo en el navegador web una de las siguientes direcciones: "http://localhost" ó "http://127.0.0.1".

Si el servidor web está funcionando, debería aparecer la página de administración de XAMPP, que nos permite seleccionar el idioma como podemos ver en la figura 46.

Figura 46: Interfaz de selección de idioma de XAMPP

Elegimos el idioma “Español”.

4.2.2 Gestión de usuarios

Para aumentar la seguridad, vamos a realizar una serie de cambios que afectan a los usuarios administradores del servidor web. Con esto nos garantizamos que solo las personas autorizadas para la administración del sistema puedan realizar modificaciones en la base de datos y visualizar los resultados a través de la interfaz web.

El usuario que viene por defecto es “root” y no tiene contraseña, por lo que cualquier persona con acceso físico al ordenador podría modificar o visualizar los resultados con permisos de administrador.

Para evitar esto, vamos a crear un nuevo usuario “root”, con todos los permisos y privilegios del administrador, pero le vamos a añadir una contraseña para aumentar la seguridad.

Para ello, abrimos un navegador web e introducimos la dirección “http://127.0.0.1”. Una vez abierta la página de administración del XAMPP, hacemos clic en “phpmyadmin”, situado en el menú lateral izquierdo.

Esta interfaz se puede ver en la figura 47.

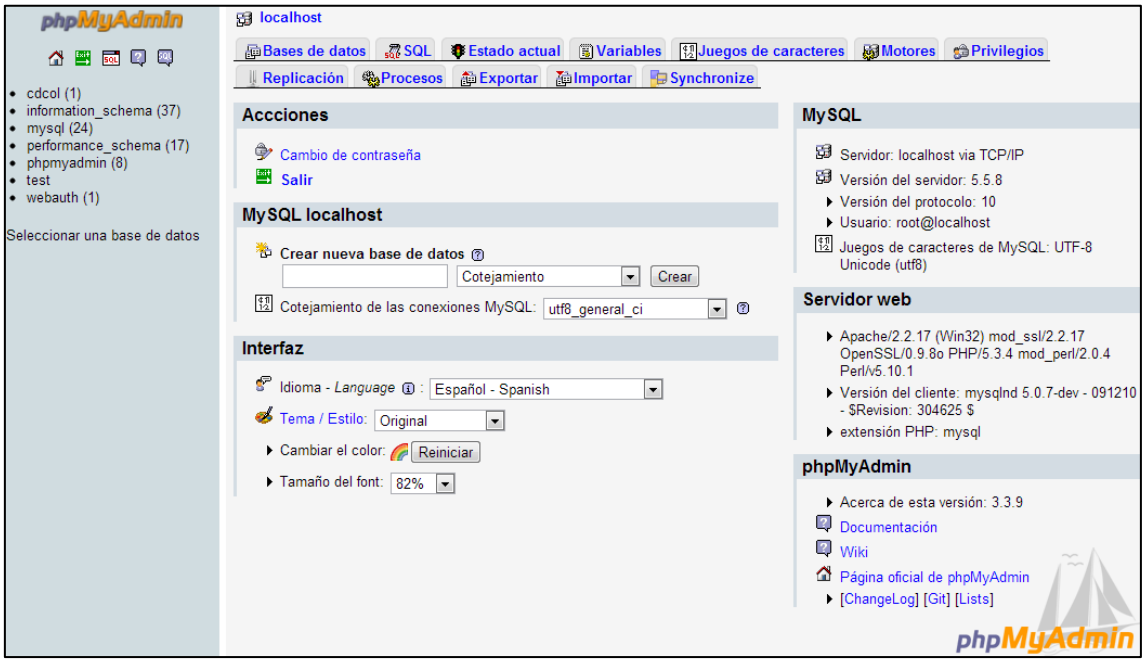

Figura 47: Interfaz de phpMyAdmin

Una vez dentro, accedemos a la pestaña “Privilegios”.

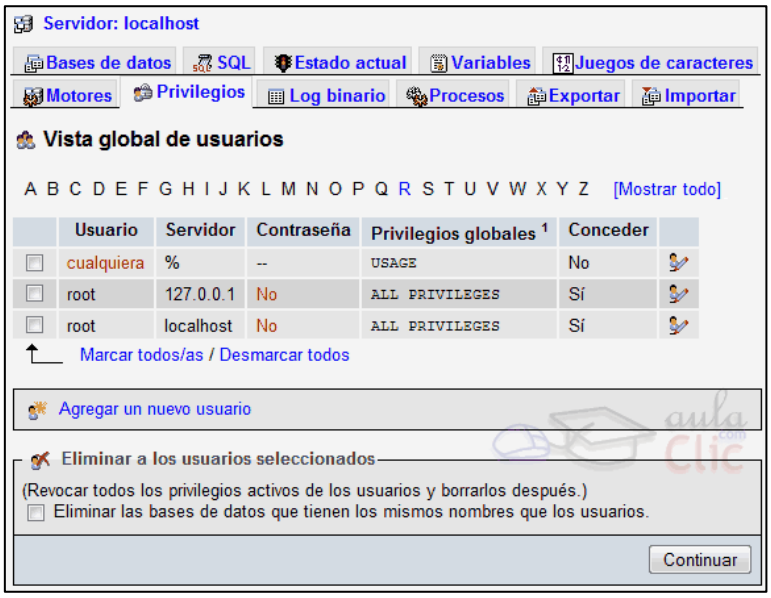

Figura 48: Vista de la pestaña "Privilegios" de phpMyAdmin

Una vez aquí, hacemos clic en “Agregar un nuevo usuario” y rellenamos los campos mostrados en la tabla 3.

Nombre de usuario:	**root**
Servidor:	**%**
Contraseña:	**TFG2013**

Tabla 3: Credenciales de la cuenta de administrador

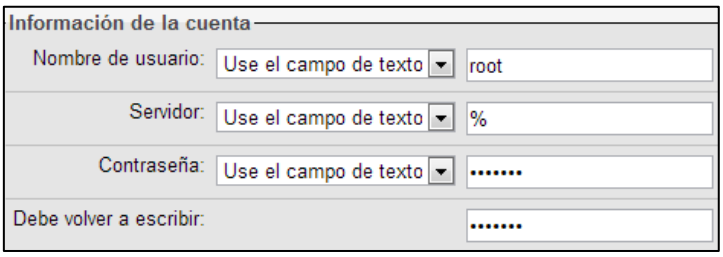

Figura 49: Captura con los datos del usuario administrador introducidos

Por último, marcamos todos los privilegios y pulsamos en “Continuar”.

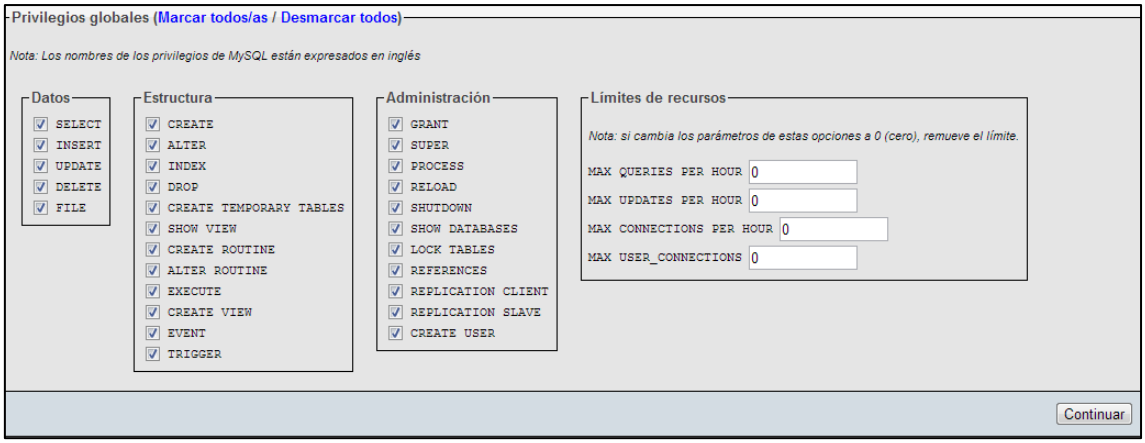

Figura 50: Captura con los privilegios de usuario seleccionados

Una vez creado el nuevo usuario administrador, vamos a modificar los ficheros de configuración de PHPMyAdmin. Estas modificaciones las realizamos para asegurarnos de que se pide siempre la contraseña de acceso al administrador. Si no llevamos a cabo estas modificaciones, nada de lo que hemos hecho anteriormente servirá.

Nos situamos en el directorio: “C:\xampp\phpMyAdmin”. Con un editor de texto abrimos el archivo “Config.inc” y sustituimos las líneas de código antiguas por las nuevas, como muestra la tabla 4.

Código antiguo	Código nuevo
`$cfg['Servers'][$i]['auth_type'] = 'config';`	`$cfg['Servers'][$i]['auth_type'] = 'cookie';`
`$cfg['Servers'][$i]['password']='';`	`$cfg['Servers'][$i]['password'] = 'TFG2013';`

Tabla 4: Sustituciones de código en el fichero Config.inc

Guardamos los cambios y listo.

Para modificar el siguiente archivo, nos situamos en el directorio: "C:\xampp\php".

Una vez situados en dicho directorio, abrimos el archivo "php.ini" y procedemos a cambiar el valor de la variable "max_execution_time" por el mostrado en la tabla 5.

Código antiguo	Código nuevo
`max_execution_time = 30`	`max_execution_time = 30000`

Tabla 5: Valor de la variable "max_execution_time"

Esta variable controla el tiempo máximo de ejecución de una petición sobre el servidor web. Para evitar posibles errores, aumentamos el tiempo de espera a un valor de 30 segundos, suficiente para procesar sin errores cualquier petición.

Por último, hay que modificar la contraseña de acceso a MySQL. Este paso es muy importante, ya que al estar MySQL y PHPMyAdmin alojados dentro del mismo administrador (XAMPP) ambos administradores tiene que tener el mismo nombre (root, por defecto) y la misma contraseña (en nuestro caso TFG2013).

Para cambiar la contraseña del administrador de MySQL, abrimos el símbolo de sistema de Windows como administrador (también conocido como *cmd*).

Una vez abierto, nos situamos en el directorio de MySQL ejecutando el siguiente comando:

```
cd C:\xampp\mysql\bin
```

Para finalizar, cambiamos la contraseña introduciendo el comando:

```
mysqladmin –u root password TFG2013
```

El resultado de la ejecución de los comandos anteriores se puede ver en la figura 51.

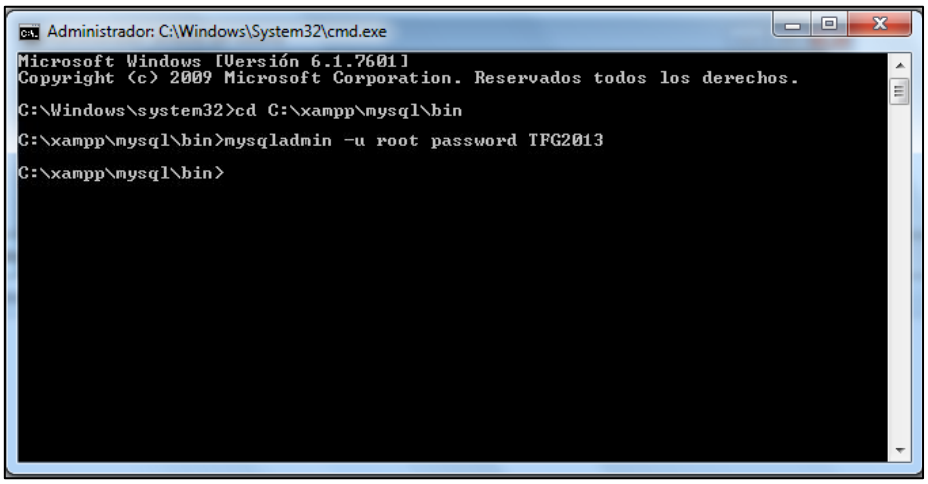

Figura 51: Captura de los comandos introducidos en el CMD

4.2.3 Interfaz web

Después de crear un nuevo usuario administrador y aumentar la seguridad del sistema, pasamos al servicio web.

El servicio web creado en este TFG (explicado en el apartado 3.4) consta de una interfaz web desde la cual podemos crear la base de datos, subir archivos y visualizar resultados, entre otras funcionalidades.

Para tener acceso al servicio creado, tenemos que situarlo dentro del servidor web. Para ello, nos situamos en el directorio “C:\xampp\htdocs” y pegamos en su interior la carpeta “SW_TFG”, la cual contiene el servicio web creado en este TFG.

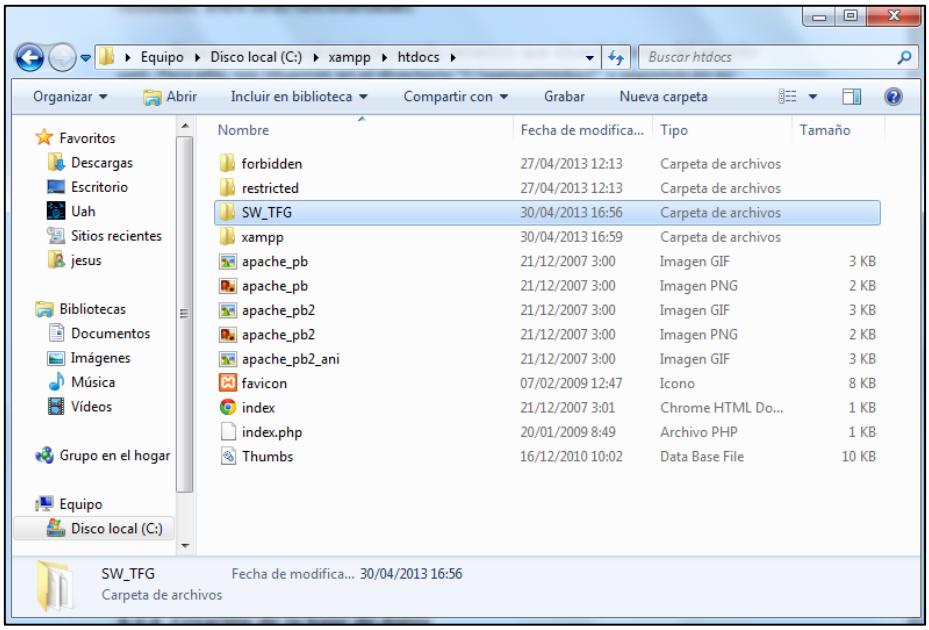

Figura 52: Vista del directorio htdocs con el servicio web

Una vez situado el servicio web en el directorio, abrimos un navegador web e introducimos la dirección "http://127.0.0.1/SW_TFG/Gestion_bd.php" para acceder a la interfaz web.

Figura 53: Interfaz principal del servicio web

4.2.4 Creación de la base de datos

Para crear la base de datos desde el servicio web, hay que estar autenticado como usuario administrador del sistema (root). Accedemos a la interfaz web a través de un navegador web, introduciendo la dirección vista en el anterior apartado, "http://127.0.0.1/SW_TFG/Gestion_bd.php".

Una vez abierta la interfaz web, nos autentificamos en el servicio con las credenciales del administrador, establecidas en el apartado 4.2.2.

Para acceder como administrador, introducimos la IP del servidor (en nuestro caso nuestro propio ordenador, 127.0.0.1) y rellenamos con las credenciales del administrador el formulario de autenticación y acceso (root/TFG2013).

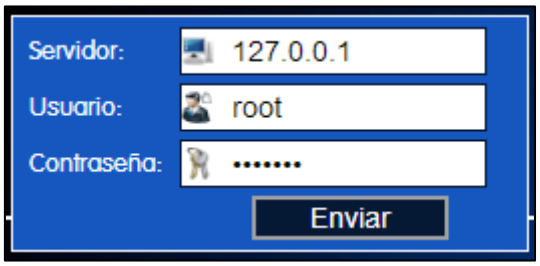

Figura 54: Formulario de acceso con las credenciales introducidas

Una vez rellenado, hacemos clic en el botón "enviar". Si se ha cumplimentado correctamente todos los campos, nos aparecerá el mensaje "Login OK" a la derecha del formulario de autenticación.

Una vez logueados, hacemos clic en la pestaña "Crear Base de datos" y rellenamos el campo numérico correspondiente al número de archivos por usuario para ser considerado sospechoso (variable *maxfile*), como vemos en la figura 55.

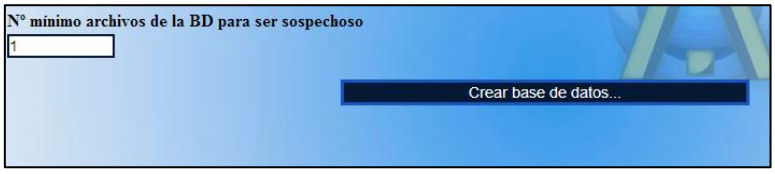

Figura 55: Introducción del valor 1 a la variable "*maxfiles*"

Por último, hacemos clic en el botón "Crear base de datos…". Si la base de datos se ha creado correctamente, aparecerá una ventana emergente con el mensaje "La base de datos se ha creado correctamente", como la que podemos ver en la figura 56.

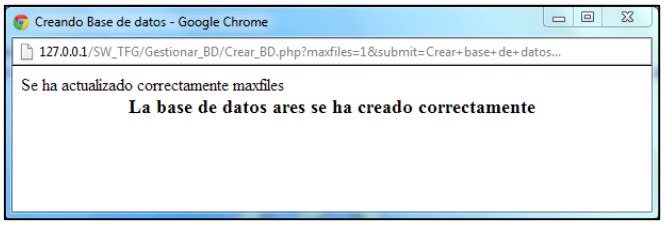

Figura 56: Ventana de confirmación de la base de datos

La nueva base de datos creada puede ser administrada desde PHPMyAdmin. Para ello, introducimos en el navegador la dirección "http://127.0.0.1/xampp" y accedemos a la herramienta PHPMyAdmin, situada en el menú lateral izquierdo. Nos logueamos con las credenciales de administrador (root/TFG2013), como se aprecia en la figura 57.

Figura 57: Formulario de acceso a phpMyAdmin

Una vez dentro, podemos ver la base de datos "ares" en la zona izquierda de la pantalla, junto con el resto de bases de datos.

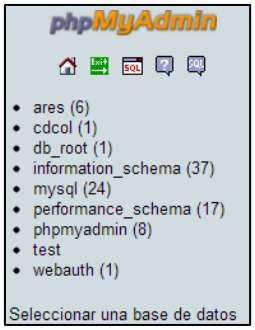

Figura 58: Bases de datos de phpMyAdmin

Si queremos acceder al contenido almacenado en las tablas de la base de datos, hacemos clic sobre “ares” en el menú izquierdo y aparecen las 6 tablas de la base de datos.

Figura 59: Vista de las tablas de la base de datos "ares" desde phpMyAdmin

4.2.5 Subida de archivos a la base de datos

Una vez creada la base de datos “ares”, debemos subir el fichero que contiene los hashes de los archivos que queremos buscar en la red P2P de Ares a través del cliente modificado.

Para subir el fichero Excel (.xls) que contiene los hashes y tamaños de los archivos, debemos estar logueados en el servicio web (visto en el apartado 4.2.4).

Una vez logueados, hacemos clic en la pestaña “Subir fichero”. Estando en dicha pestaña, hacemos clic en el botón “Seleccionar archivo” y seleccionamos el fichero Excel (.xls) que contiene los hashes.

Figura 60: Captura previa a la subida del archivo Excel

Por último, hacemos clic en el botón “Subir el fichero a la base de datos” y si se ha subido correctamente aparecerá una ventana emergente con la mostrada en la figura 61.

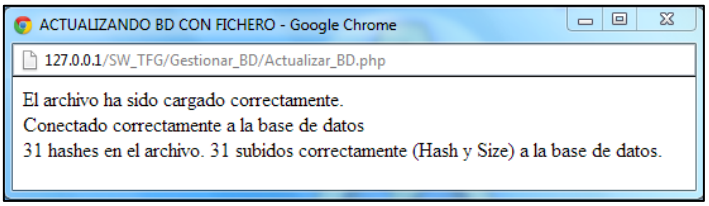

Figura 61: Ventana de confirmación de carga de hashes en la base de datos

Una vez subido el fichero, los hashes y el tamaño de los archivos son introducidos en la tabla "all_hash" de la base de datos "ares". Esto lo podemos comprobar desde el administrador de XAMPP (http://127.0.0.1/xampp), dentro de la herramienta PHPMyAdmin (como vimos anteriormente en el apartado 4.2.4).

4.2.6 Descarga del fichero de configuración

Por último, antes de poner en funcionamiento el cliente de Ares modificado, hay que descargar el fichero de configuración de la base de datos. Dicho fichero contiene el nombre de la base de datos, el nombre de usuario y la contraseña.

Como hemos explicado en anteriores capítulos, este fichero es utilizado por el cliente de Ares modificado para autenticarse frente a la base de datos antes de realizar cualquier operación, ya sea de carga o descarga.

Para descargarnos el fichero de configuración hay que estar logueado en el servicio web.

Una vez logueados en el servicio web, hacemos clic en la pestaña "Descargar Base_datos.txt". Una vez en esta pestaña, hacemos clic en el icono de descarga, figura 62, y se abrirá una ventana emergente como la de la figura 63.

Figura 62: Icono de descarga

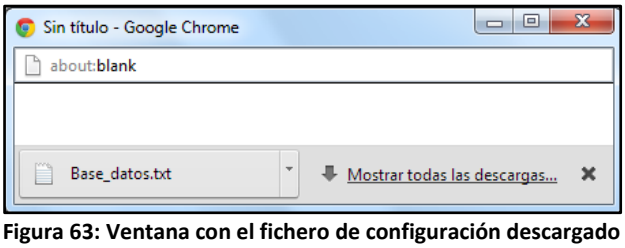

Figura 63: Ventana con el fichero de configuración descargado

El fichero “Base_datos.txt” lo encontraremos en la carpeta de descargas que tengamos establecida por defecto en nuestro navegador web. Lo más común es que el fichero se descargue en el directorio “C:\Users\usuario\Descargas”.

Por último, debemos colocar el fichero “Base_datos.txt” dentro de la carpeta “Ares_Modificado_TFG”. Si no colocamos el fichero en dicha carpeta, el cliente de Ares modificado no funcionara, ya que el fichero de configuración debe estar en el mismo directorio que el ejecutable “Ares.exe”. Lo podemos ver en la figura 64.

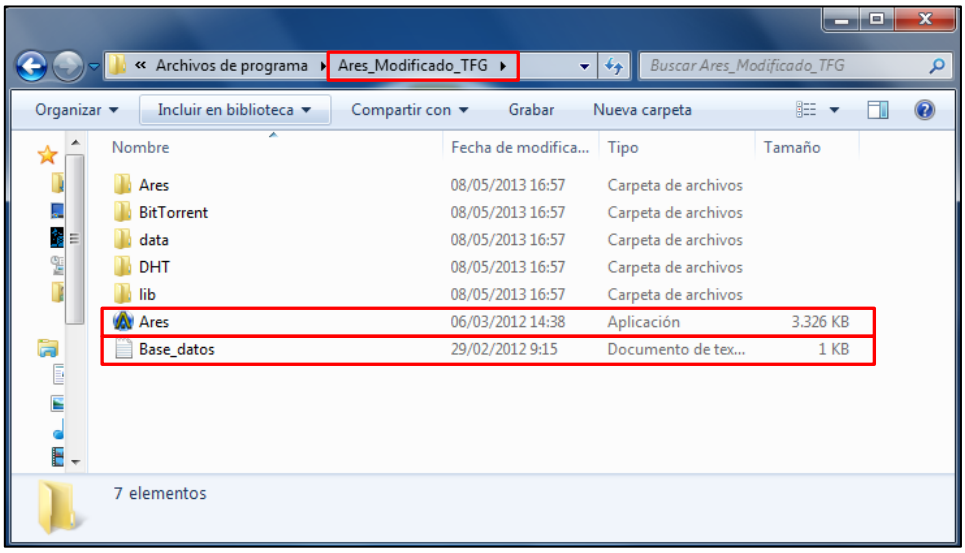

Figura 64: Directorio con el ejecutable y el fichero de configuración

4.3 Ares modificado

Una vez realizados todos los pasos previos y necesarios para el funcionamiento del sistema, podemos ejecutar el cliente de Ares modificado. Antes de poner en funcionamiento Ares, hay que asegurarse de que el fichero de configuración de la base de datos se ha descargado correctamente y de que está situado en el mismo directorio de la base de datos, tal y como se explica en el apartado 4.2.6.

Ahora sí, hacemos doble clic en el archivo ejecutable llamado “Ares.exe”. Una vez abierto, nos situamos en la pestaña “Transfer”. Inicialmente, esta pantalla esta vacía, tal y como se muestra en la figura 65.

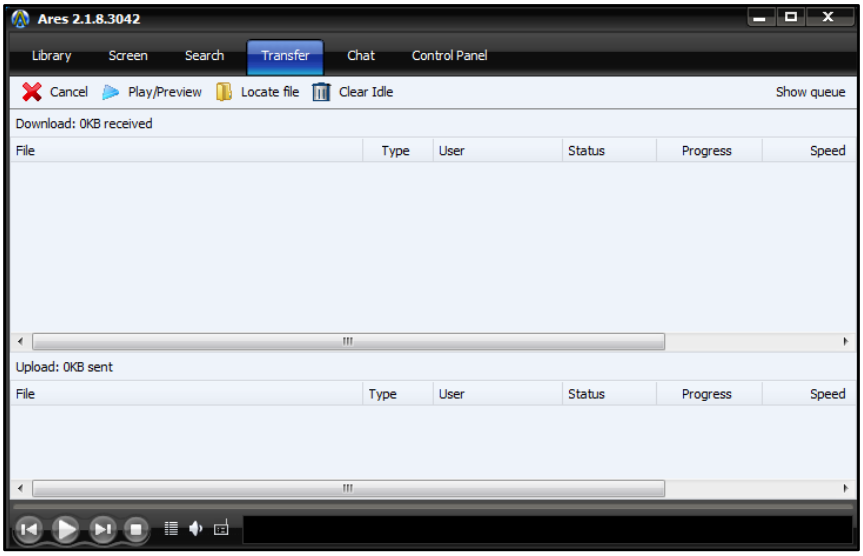

Figura 65: Estado de la pestaña "Transfer" al iniciar Ares

Al cabo de unos segundos, el cliente de Ares modificado se descargará los hashes desde la base de datos y se pondra a buscar dichos arrchivos por la red, como podemos apreciar en la figura 66.

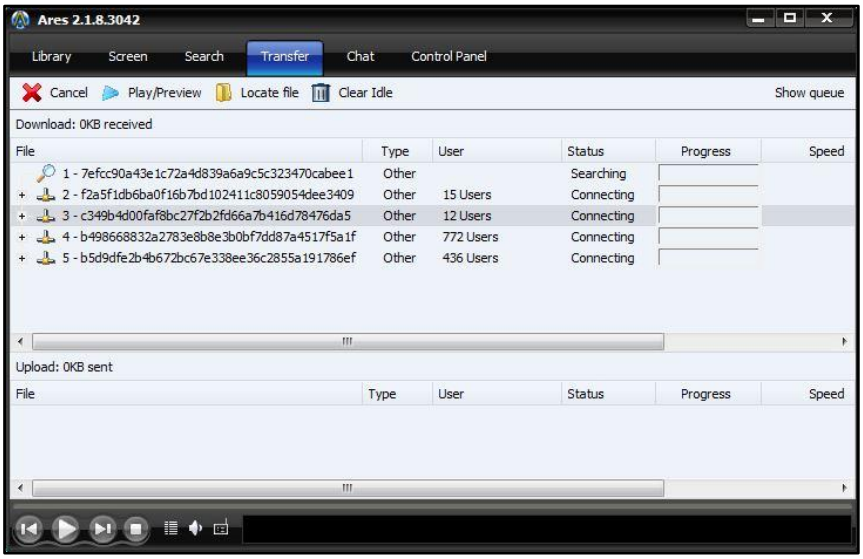

Figura 66: Pestaña "Transfer" con los hashes descargados

En la figura superior podemos ver que los archivos no aparecen con su nombre original, ya que el cliente de Ares ha sido modificado para que muestre el hash del archivo que está buscando, el cual obtiene de la base de datos.

El número que acompaña en la interfaz a cada hash se corresponde con la posición que tienen en la tabla de la base de datos, su identificador en la tabla.

El cliente solo funciona correctamente si está abierta la pestaña “Transfer”. Si nos situamos en cualquier otra pestaña, el cliente puede dejar de conectarse con la base de datos o incluso puede dejar de funcionar. Es importante que el cliente de Ares modificado pase la mayor parte del tiempo situado en la pestaña “Transfer”, tal y como se muestra en la figura 66.

El cliente de Ares busca estos archivos durante un periodo de entre 4 y 5 minutos. Una vez cumplido este tiempo, el cliente de Ares descarta la búsqueda de los archivos actuales, registra las fuentes encontradas en la base de datos y se descarga cinco nuevos hashes para volver a comenzar la búsqueda.

Las fuentes encontradas para cada archivo se muestran en la columna llamada “User”. Si queremos ver las IPs de los usuarios encontrados para un archivo, basta con desplegar el contenido de cada hash, haciendo clic en el botón “+”. La información se mostrará como se muestra en la figura 67.

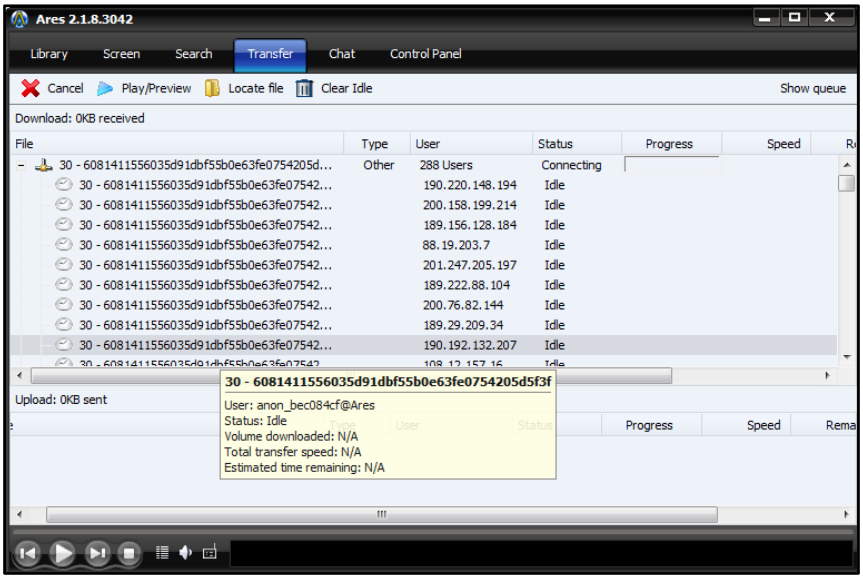

Figura 67: IPs de los usuarios que tienen el archivo

Para que el cliente de Ares pueda realizar las búsquedas y capture fuentes, debemos asegurarnos de que tiene acceso a internet. Para ello nos situamos en la pestaña “Control Panel”. En dicha pestaña debe aparecer el mensaje “Connected” y a continuación el nombre de usuario que nos ha asignado el propio cliente, tal y como se puede ver en la figura 68.

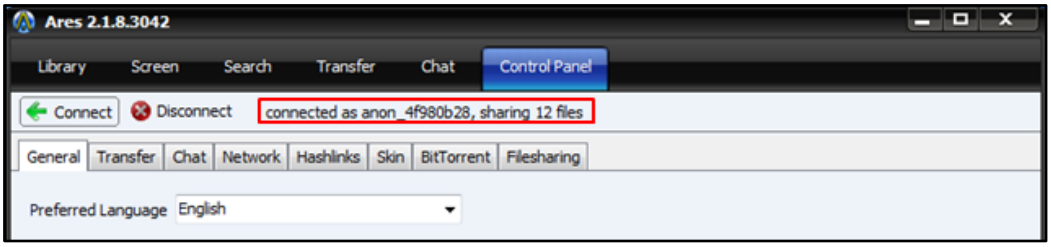

Figura 68: Pestaña "Control Panel" de Ares

Los archivos que estamos buscando pueden aparecer con dos estados distintos. Estos estados aparecen en la columna "Status" con el valor *Searching* ó *Connecting*, como podemos apreciar en la figura 69.

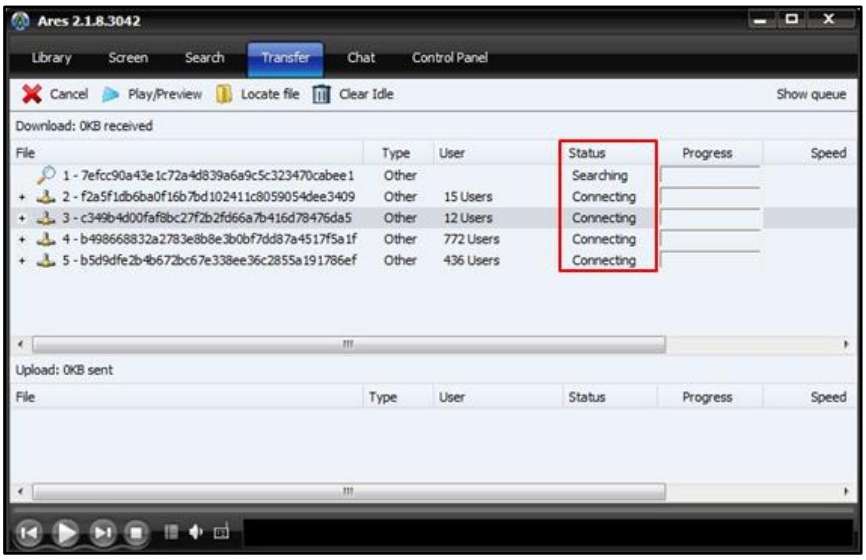

Figura 69: Estado de los archivos buscados

Si un archivo aparece acompañado del estado *Searching*, significa que está buscando fuentes y que no ha encontrado ninguna. Si por el contrario tiene un estado *Connecting*, significa que ha encontrado fuentes, aunque sigue buscando más.

El resto de funcionalidades del cliente de Ares modificado en este TFG son las mismas que las de cualquier cliente de Ares, exceptuando la descarga de archivos. El programa está programado para no descargar archivos, solo localiza fuentes. De esta forma evitamos que nosotros mismos podamos aparecer en la base de datos como portadores de algún archivo.

4.4 Gestión del fichero de hashes

Los hashes almacenados en la base de datos son gestionados desde la interfaz del servicio web. Como vimos en el apartado 3.5.3, desde la pestaña “Subir fichero” podemos cargar los hashes en la base de datos e incluso actualizar los hashes cargados actualmente.

En el apartado 4.2.5 hemos visto como subir el fichero Excel (.xls) de hashes a la base de datos, pero también podemos actualizar los hashes que hemos subido e incluso subir nuevos.

Una vez cargada la base de datos, si queremos subir un nuevo fichero Excel para aumentar el número de hashes, lo único que tenemos que hacer es subirlo como vimos en el apartado 4.2.5. Durante este proceso, se comprobara si alguno de los hashes del nuevo fichero ya existía en la base de datos. Si ya existía, no se sube y pasa al siguiente. De esta forma se actualiza el contenido de la base de datos con los nuevos hashes que no estaban en la base de datos.

4.5 Gestión de resultados

Para poder visualizar los resultados y la información recopilada por el cliente de Ares modificado, tenemos que acceder a la pestaña “Mostrar resultados” del servicio web.

Figura 70: Pestaña "Mostrar resultados" del servicio web

Desde esta pestaña, tenemos 4 opciones de ver los resultados, cada una distinta y orientada a objetivos distintos.

La primera forma de visualizar los resultados es en formato web, haciendo clic en el botón “WEB”. Esta forma de visualización consiste en la apertura de una nueva ventana en el navegador web en la cual se nos muestra los resultados almacenados en la base de datos “ares” de forma general. Como se puede ver en la figura 71, primero se muestra del número de hashes (ficheros) que hay cargados en la base de datos y se nos informa de la cifra de usuarios distintos (fuentes) capturados por el cliente de Ares modificado. Además, aparece una tabla que contiene el nombre de usuario, la IP asociada a dicho usuario, el número de archivos encontrados a este usuario y el país de origen de la IP.

Existen 31 ficheros en la base de datos

Se han detectado 21186 usuarios diferentes

IP	ARCHIVOS ENCONTRADOS	NOMBRE DE USUARIO	ORIGEN
189.115.18.5	1	raulvianna@Ares	BRAZIL
189.200.193.252	4	anon_bdc8c1fc@Ares	MEXICO
186.251.23.43	2	acontece@Ares	BRAZIL
92.58.92.71	1	anon_5c3a5c47@Ares	SPAIN
177.3.101.145	2	yaas@Ares	COLOMBIA
80.102.16.52	2	lunallena1411@Ares	SPAIN
189.101.252.31	2	anon_bd65fc1f@Ares	BRAZIL
190.193.66.198	5	anita@Ares	ARGENTINA
189.1.177.122	1	OctavioRibeiro7@Ares	BRAZIL
200.127.34.222	1	anon_c87f22de@Ares	ARGENTINA
189.250.236.43	1	anon_bdfaec2b@Ares	MEXICO
201.248.102.100	2	aguila@Ares	VENEZUELA
190.72.26.64	1	anon_be481a40@Ares	VENEZUELA
201.252.44.195	2	habi@Ares	ARGENTINA
201.254.87.123	2	anon_c9fe577b@Ares	ARGENTINA
190.206.23.211	1	anon_bece17d3@Ares	VENEZUELA
41.214.194.172	1	anon_29d6c2ac@Ares	JAPAN
189.234.69.142	1	anon_bdea458e@Ares	MEXICO
201.161.190.61	2	fuerto@Ares	MEXICO
181.116.192.68	1	anon_b574c044@Ares	COLOMBIA
177.96.35.185	4	Fran@Ares	COLOMBIA

Figura 71: Vista web del informe general

Otra forma de visualizar los datos es mostrarlos por país. Si hacemos clic en el botón “Informe Pais”. Al hacer clic en este botón aparece una venta emergente como la de la figura 72, en la cual tenemos que seleccionar el país del cual queremos obtener el informe.

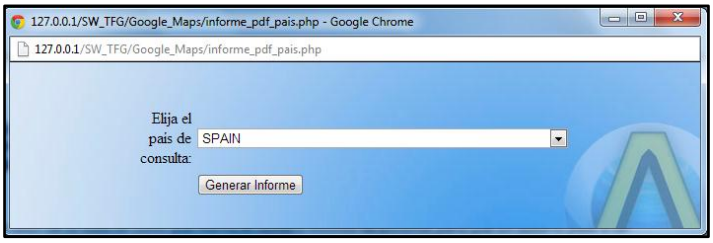

Figura 72: Ventana de selección de país

Una vez elegido el país, se realizaran varias consultas en la base de datos para obtener las IPs registradas de ese país y recuperar los nombres de usuarios y archivos encontrados a dichos usuarios. Esta información será descargada a nuestro ordenador en forma de archivo PDF, como podemos observar en la figura 73.

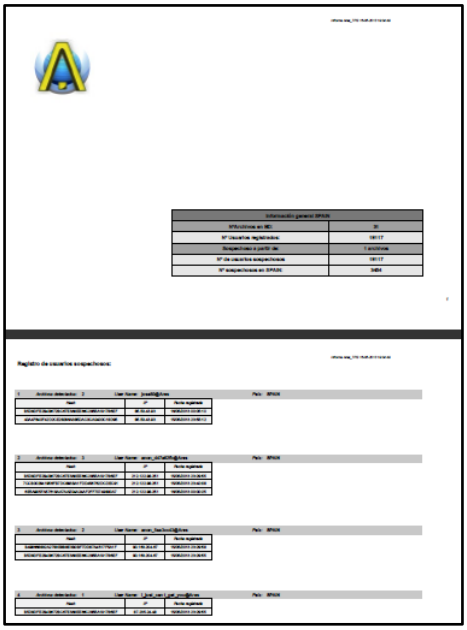

Figura 73: Aspecto de "Informe_pais.pdf"

La siguiente opción que se ofrece para ver los datos registrados es mostrar a los usuarios que tienen un determinado archivo. Para ello, si se hace clic en el botón "Informe Hash" se abre una ventana emergente como la de la figura 74, en la cual tenemos que introducir el hash del archivo del cual queremos conocer a los usuarios que lo poseen.

Figura 74: Ventana para introducir el hash

El informe resultante presenta a los usuarios que tienen el archivo, acompañados del hash del archivo, la IP del usuario y la fecha en la que les fue encontrado el archivo. Las primeras páginas de dicho informe PDF se pueden ver en la figura 75.

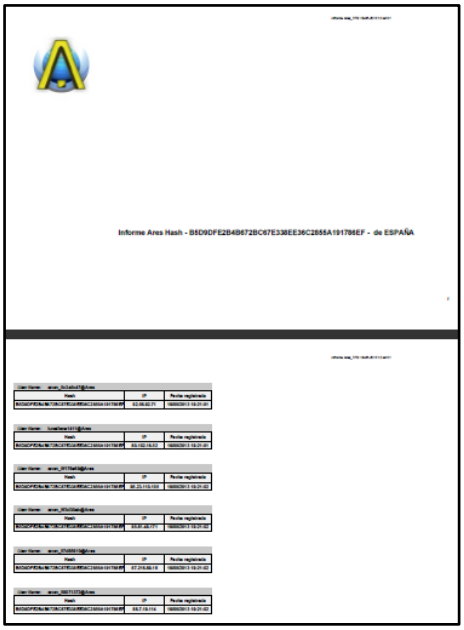

Figura 75: Aspecto de "Informe_hash.pdf"

La última forma de visualizar los resultados consiste en un informe sobre un usuario concreto, en el cual se nos muestran todos los archivos que tiene en su poder dicho usuario. Este informe se obtiene haciendo clic en el botón “Informe Username”. Después de hacer clic, se abrirá una ventana como la de la figura 76, en la cual tendremos que introducir el nombre del usuario del cual queremos ver la información.

Figura 76: Ventana para introducir el nombre de usuario

El nombre de usuario que tenemos que introducir es el *username* que utiliza dicho usuario en la red de Ares. El informe resultante contiene el hash de los archivos encontrados a dicho usuario, la IP del usuario y la fecha exacta en la cual se le encontraron cada uno de los archivos. El archivo PDF resultante lo podemos observar en la figura 77.

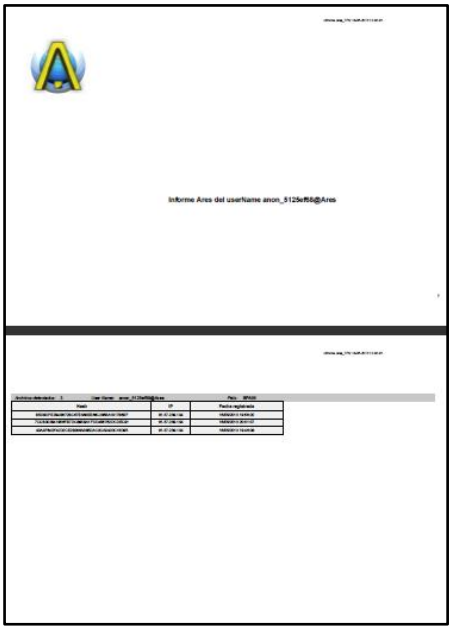

Figura 77: Aspecto de "Informe_username.pdf"

4.6 Preguntas más frecuentes

- **Problemas para acceder al servidor:** Si la herramienta XAMPP se ha instalado correctamente y el servidor Apache está en funcionamiento, desde la misma máquina se puede acceder al servidor desde un navegador web introduciendo http://127.0.0.1 o localhost. Si el servidor está en otro ordenador, hay que introducir su IP dinámica en un navegador web.

- **Problemas de acceso a phpMyAdmin:** Si no puede acceder a phpMyAdmin, no puede acceder al gestor de la base de datos. Este problema se soluciona editando el fichero *php.ini*, ubicado en C:\xampp\php. Una vez abierto el archivo, hay que descomentar las líneas mostradas en la tabla 6.

Código antiguo	Código nuevo
;extension=php_mssql.dll ;extension=php_mysql.dll ;extension=php_mysqli.dll	extension=php_mssql.dll extension=php_mysql.dll extension=php_mysqli.dll

Tabla 6: Cambios en php.ini

Después de realizar las modificaciones anteriores, asegúrese de guardar los cambios en el archivo y de reiniciar el servidor Apache para que se ejecute dicho cambio.

- **Localización del archivo config.inc:** Este fichero se encuentra en el directorio de XAMPP. La ruta del fichero es *C:\xampp\phpMyAdmin\config.inc*.

  Dicho fichero contiene la configuración cargada actualmente en el servidor.

- **Localización del archivo php.ini:** Este archivo se encuentra en el directorio de XAMPP. Por defecto el directorio XAMPP se encuentra en C:, por lo que la ruta del archivo es *C:\xampp\php\php.ini*.

  Otra forma de conocer la ubicación del archivo es acceder desde un navegador a la ruta localhost/xampp y ejecutar el menú *phpinfo*. Una vez en esta ubicación, se verifica la ubicación del archivo en *Loaded Configuration File* que nos indica el path del fichero.

  El archivo php.ini contiene la configuración cargada actualmente para PHP.

- **Cómo reiniciar el servidor Apache:** Cada vez que se modifique el archivo php.ini se debe reiniciar el servidor Apache para que se ejecuten los cambios. Para ello hay que abrir el panel de control de XAMPP y hacer clic sobre "SCM". Se abrirá una ventana con los servicios locales que se están ejecutando, buscamos Apache y reiniciamos el servicio. El proceso tardara varios segundos y puede que salten mensajes de error en los que se indica que Apache HTTP dejo de funcionar.

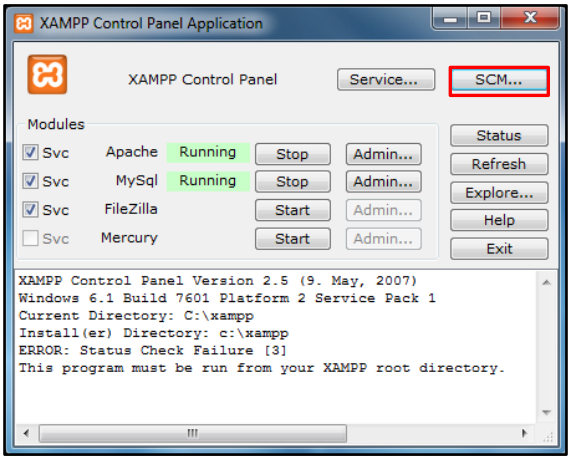

Figura 78: Localización de SCM en el panel de control de XAMPP

- **Borrado de la base de datos:** Esta acción hace que se pierdan todos los datos almacenados en la base de datos, por lo que si no quiere perderlos absténgase de realizarla. Para borrar la base de datos hay que acceder a phpMyAdmin desde cualquier navegador web introduciendo la dirección

http://localhost/phpmyadmin/. Una vez aquí hay que introducir las credenciales de administrador (Nombre y contraseña) para acceder. Una vez que somos administradores del sistema, hacemos clic sobre la base de datos "ares", que se encuentra en el menú lateral izquierdo. Por último, hacemos clic en "Eliminar" que se encuentra en la zona superior de la pantalla y remarcado con letras rojas. Si quiere volver a crear la base de datos, vaya al apartado 4.2.4 del manual de usuario.

Figura 79: Localización del botón "Eliminar"

- **Cómo crear archivo Excel con extensión valida:** Debido al gran número de aplicaciones y versiones existentes para crear archivos de Excel, a continuación explicamos cual es la forma correcta de guardarlo. Teniendo Microsoft Excel abierto, cuando hayamos rellenado correctamente las columnas hash y tamaño, hacemos clic en "*Guardar como*" y elegimos el formato "*Microsoft Excel 97/2000/XP (.xls)*".

- **Error de conexión en interfaz web por falta de permisos:** Esto ocurre porque la sesión iniciada en el servicio web ha caducado. Por motivos de seguridad, la sesión caduca cuando pasan 15 minutos desde el comienzo. Lo único que tenemos que hacer para solucionarlo es registrarnos nuevamente en el formulario de la interfaz web. Si tiene dudas de como iniciar una sesión o registrarse en la interfaz web, acudir al apartado 4.2.4 del manual de usuario.

4.7 Glosario

- **Autenticación:** Proceso de verificación de la identidad digital de un usuario. Durante este proceso se comprueba que un usuario es quién dice ser y se verifican los permisos que tiene asignados.

- **Broadcasting:** Es un tipo de difusión de la información utilizado en las redes informáticas. Consiste en que la información emitida por un nodo llega a todos los nodos de la red. Ver *Difusión por inundación*.

- **CMD:** También conocido como *Símbolo del sistema*. Es el intérprete de comandos de Windows. Desde esta consola podemos ejecutar diversas instrucciones que nos permiten administrar el sistema.

- **Difusión por inundación:** es una forma de transmisión de información donde un nodo emisor envía información a una multitud de nodos receptores de manera simultánea, sin necesidad de reproducir la misma transmisión nodo por nodo.

- **Dirección IP:** Es un número que identifica de manera lógica y jerárquica a una interfaz de un dispositivo (ordenador) dentro de una red. Cada usuario que se conecta desde su ordenador a Internet utiliza una dirección IP. Esta dirección IP es de asignación variable, esto quiere decir que puede cambiar cada vez que se conecta. A esto se lo conoce como dirección IP Dinámica.

- **Fichero Excel:** Fichero propio de la aplicación Microsoft Excel, perteneciente al paquete de Microsoft Office. La extensión de dicho fichero es *.xls*.

- **Hash:** Clave hexadecimal que identifica de forma única a un archivo. Aunque un archivo tenga distintos nombres, o un usuario haya modificado el nombre de dicho archivo, el identificador hash sigue siendo el mismo.

- **Interfaz web:** Apariencia o presentación gráfica que permite al usuario navegar de forma cómoda e intuitiva. Las interfaces sirven de intermediarias entre los usuarios y los sistemas de información que funcionan a más bajo nivel. Permiten al usuario recuperar la información deseada y navegar por el sistema sin tener altos conocimientos del mismo.

- **Localhost:** Nombre reservado que tienen todos los ordenadores, routers o dispositivos con tarjeta de red para referirse a sí mismo. Este nombre es traducido como la dirección IP 127.0.0.1.

- **Login /loguear:** Autenticación verificando que el usuario que intenta acceder al sistema es quien dice ser y tiene permisos de acceso.

- **Loopback:** Es una interfaz de red virtual. Las direcciones del rango '127.0.0.0/8' son direcciones de loopback. La más utilizada de estas direcciones es la '127.0.0.1' por ser la primera de dicho rango y la correspondiente al *localhost*.

- **Maxfile:** Parámetro que indica el número máximo de ficheros que puede tener un usuario registrado por el sistema sin que sea considerado sospechoso.

- **P2P:** *Peer-to-Peer*, red punto a punto o red entre iguales. Es una red de ordenadores que funciona sin el concepto cliente y servidor fijo, es decir, todos los nodos pertenecientes a la red funcionan simultáneamente como clientes y servidores. Además, todos los nodos son considerados como iguales, a nivel de permisos y prioridades en la red.

- **Path:** Palabra muy utilizada en el ámbito de la informática para referirnos a la ruta en la cual está un archivo o un directorio.

- **Pestaña:** Elemento de una interfaz web o de un programa que permite cambiar rápidamente lo que se está viendo sin cambiar de ventana. Permite cargar varios elementos separados dentro de una misma ventana y así es posible movernos entre ellos con mayor velocidad y comodidad.

- **Petición:** Solicitud enviada por un usuario a otro usuario de la red para satisfacer una necesidad. Habitualmente las peticiones son utilizadas para solicitar archivos, direcciones o para autenticarse ante un servidor.

- **Root:** Nombre convencional de la cuenta de usuario que posee todos los derechos y permisos. Es también conocido como administrador o superusuario.

- **Servicio web:** Tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Los servicios web permiten intercambiar datos entre aplicaciones desarrolladas en lenguajes de programación diferentes y ejecutados en sistemas operativos distintos.

- **Servidor:** Es un ordenador que, formando parte de una red, provee de servicios al resto de ordenadores de la red, denominados clientes.

- **Símbolo del sistema:** ver ***CMD***.

Capítulo 5

Requisitos del sistema y presupuesto

En este capítulo se describirán los requisitos mínimos, tanto hardware como software, necesarios para el correcto funcionamiento del sistema y el presupuesto estimado para llevar a cabo la realización del TFG.

5.1 Requisitos del sistema

5.1.1 Hardware

Tanto la herramienta XAMPP, que contiene el servidor web y la base de datos, como el cliente de Ares funcionan en el mismo ordenador..

El ordenador encargado de albergar tanto el servidor como el cliente modificado de Ares necesita tener, como mínimo, las siguientes características a nivel de hardware:

- **Procesador:** *Intel Dual-Core 2 GHz o similar.*
- **Memoria RAM:** *2 Gb.*
- **Espacio en disco:** *30 Gb*
- **Conexión a Internet.**

La aplicación funciona correctamente en ordenadores con las características anteriores. Se admiten configuraciones distintas, siempre y cuando sean superiores a las mencionadas anteriormente. No obstante, la aplicación puede funcionar en ordenadores con características más bajas, pero no se asegura su correcta ejecución en largas tandas de funcionamiento.

5.1.2 Software

El ordenador que va a ejecutar dicha aplicación, necesita tener como mínimo las siguientes características a nivel de software, entre las que se incluyen los programas más importantes:

- **Sistema Operativo:** *Microsoft Windows 7 de 32-Bits.*
- **Navegador web:** *Chrome (Recomendado) ó Firefox.*
- **Editor de texto:** N*otePad ++.*
- **Visor de documentos PDF:** *Adobe Reader.*
- **Visor de documentos Excel:** *Microsoft Excel.*
- **Conector MySQL:** MySQL Connector ODBC.
- **Entorno de programación:** *Borland Delphi 7.*
- **Herramienta XAMMP con Apache Server y MySQL.**
- **Cliente de Ares modificado.**

Se garantiza el funcionamiento del servicio web en los navegadores Chrome y Firefox. Si se utiliza cualquier otro navegador web, la interfaz puede presentar algunos errores, solo a nivel de apariencia. La funcionalidad no presentaría dichos errores.

5.2 Presupuesto

El presupuesto de este Trabajo Fin de grado se va a dividir en varias partes: Mano de obra en función de los días trabajados, coste de equipos informáticos, costes de conexión a internet y licencias de las diversas aplicaciones.

Mano de obra

Tipo de trabajador	Salario (Euros/día)
Ingeniero	*80*

Tabla 7: Salario base de un Ingeniero al día

La realización de este Trabajo Fin de Grado ha durado 2 meses con una media de 30 días al mes, lo que hace un total de 60 días de los cuales 42 son laborables.

La media de horas al día dedicada a la realización de este Trabajo Fin de Grado ha sido de 4 horas, lo que hace un total de 160 horas.

Tipo de Trabajador	Días	Salario (Euros)	Coste Total (Euros)
Ingeniero	*42*	*80*	*3360*

Tabla 8: Salario Total del trabajador

Coste de equipos informáticos

En este apartado, se calcula el coste destinado a los equipos informáticos utilizados en este proyecto.

Equipo	Descripción	Antigüedad (años)	Valor Actual (Euros)
Portátil LG R510 Core 2 Duo 2.4 GHz	*Equipo utilizado para el desarrollo del sistema y la realización de pruebas.*	*3*	*300*

Tabla 9: Coste del equipo Informático

Coste de conexión a internet

Concepto	Tarifa (Euros/mes)	Meses	Coste (Euros)
ADSL 10Mb	*24.90*	*2*	*49,80*

Tabla 10: Coste de conexión a Internet

Licencia de aplicaciones

Aplicación	Tipo de licencia	Descripción	Tarifa (Euros/mes)	Coste (Euros)
Adobe Photoshop CS5	*Gratuita. 30 días de prueba*	*Servicios limitados*	*0*	*0*
Adobe Photoshop CS5	*1 mes*	*Todos los servicios disponibles*	*24.99*	*24.99*

Tabla 11: Coste licencia de programas

Presupuesto total

Concepto	Coste (Euros)
Total mano de obra	*3360*
Total equipos informáticos	*300*
Total conexión a Internet	*49.80*
Total licencia de aplicaciones	*24.99*
Suma total	*3734,79*
IVA (21%)	*784,31*
Importe total	**4519,10**

Tabla 12: Presupuesto total

Capítulo 6

Conclusiones

A lo largo de este Trabajo Fin de Grado se ha explicado el desarrollo de un sistema para localizar a los usuarios poseedores de ciertos archivos en la red de Ares. Se ha cumplido con dicho objetivo principal del trabajo y, además, se ha conseguido un entorno intuitivo y fácil de manejar por lo que el usuario final de esta aplicación no requiere altos conocimientos en ámbitos de la informática para manejar correctamente el sistema.

El sistema está formado por servidor web, base de datos, interfaz web y el cliente de Ares modificado. Todos los componentes interactúan entre sí para hacer posible el correcto funcionamiento del sistema. La interfaz web hace posible la visualización de los resultados almacenados en la base de datos. El cliente de Ares es el encargado de recopilar la información de los usuarios de la red y de almacenarla en la base de datos.

Para facilitar el manejo del sistema, este Trabajo Fin de Grado incluye un manual de usuario que describe los pasos previos que hay que realizar y explica las principales funcionalidades que nos ofrece el sistema. Además, el manual incluye un glosario con la explicación de los términos más técnicos y un apartado de preguntas frecuentes, que ayuda a solucionar los errores más comunes que se producen en la aplicación.

Una vez terminada la parte de desarrollo, se han dedicado un par de semanas a realizar pruebas al sistema, obteniendo resultados muy satisfactorios. Dentro de las pruebas realizadas, se ha podido observar que el cliente de Ares modificado ha capturado más de treinta mil usuarios distintos, de los cuales más de mil tenían 5 o más archivos de los buscados. Los archivos buscados en la red de Ares durante estas pruebas se componen de música y películas actuales, lo que aumentaba significantemente el número de fuentes.

Los usuarios detectados provienen de todas las partes del mundo, por lo que la mayor parte de las fuentes obtenidas están localizas en Estados Unidos y países europeos, entre los que destacan Reino Unido. Alemania y España.

Por último se procedió a la exportación de los tres tipos de informes que nos ofrece el servicio web, realizándose satisfactoriamente la creación de todos ellos.

Bibliografía

Libros

[1] Cantú, M. (2003). *La Biblia de Delphi 7*, Primera edición, Anaya multimedia.

[2] Charte, F. (1996). *Programación con Delphi*, Tercera edición, Anaya multimedia.

[3] Charte, F. (2005). *La biblia de HTML*, Anaya multimedia.

[4] Holzner, S. (2009). *PHP manual de referencia*, McGraw-Hill.

[5] Kurose, J. y Ross, K. (2010). *Redes de computadores: Un enfoque descendente*, Pearson Addison Wesley.

[6] Naramore, E., Glass, M., y Le Scouarnec, Y. (2004). *Desarrollo web con PHP, Apache y MySQL*, Primera edición, Anaya multimedia.

Enlaces de componentes

[7] ActiveX, http://www.adobe.com/es/downloads/

[8] Ares 2.1.8, http://sourceforge.net/projects/aresgalaxy/files/aresgalaxy/AresRegular218_020212/

[9] Borland Delphi 7, http://www.embarcadero.com/products/delphi

[10] DSPACK231, http://es.sourceforge.jp/projects/sfnet_dspack/downloads/dspack/DSPAck%202.31/DSPACK231.zip/

[11] EmbeddedWB, http://sourceforge.net/projects/embeddedwb/

[12] JCL y JVCL, http://www.delphi-jedi.org/

[13] MySQL Connector ODBC, http://dev.mysql.com/downloads/connector/odbc/

[14] Paquete ESBPCS , http://www.esbconsult.com/esbpcs/downloads.htm

[15] TntWare Delphi Unicode, http://www.axolot.com/TNT/

[16] XAMPP, http://www.apachefriends.org/en/xampp-windows.html

Enlaces de documentación

[17] Algoritmo SHA-1, Wikipedia (2013), http://es.wikipedia.org/wiki/Secure_Hash_Algorithm

[18] Ares Galaxy, Wikipedia (2013), http://es.wikipedia.org/wiki/Ares_Galaxy

[19] Consultas SQL en Delphi, http://cs.uns.edu.ar/~gis/ebd/Archivos/clases%20practicas/claseMySQL-Delphi.pdf

[20] Consultas SQL en Delphi, http://delphiallimite.blogspot.com.es/2007/10/creando-consultas-sql-con-parmetros.html

[21] Delphi, http://delphi.about.com/

[22] Diferencias entre Arquitecturas de Red, http://www.slideshare.net/yeinier/diferencia-entre-cliente-servidor-y-p2-p

[23] Función Hash, http://www.genbetadev.com/seguridad-informatica/que-son-y-para-que-sirven-los-hash-funciones-de-resumen-y-firmas-digitales

[24] Función Hash, http://www.monografias.com/trabajos76/funciones-hash-criptografia/funciones-hash-criptografia2.shtml

[25] Función Hash, Wikipedia (2013), https://es.wikipedia.org/wiki/Funci%C3%B3n_hash

[26] Funcionamiento Ares Galaxy, http://foro.elhacker.net/programacion_general/protocolo_ares_galaxy-t315776.0.html

[27] Funcionamiento de Ares Galaxy, http://aresgalaxy2010junio.blogspot.com.es/2010/06/ares-galaxy.html

[28] Funcionamiento de Ares Galaxy, http://espanol-articulos.com/como-funciona-la-red-de-ares-galaxy/

[29] Funcionamiento de Ares Galaxy, http://planetaneutro.blogspot.com.es/2008/09/como-funciona-ares-tutorial.html

[30] Funcionamiento de Ares Galaxy, http://www.vidadigitalradio.com/compartir-descargar-ares/

[31] Historia de Ares, http://5tar7eam1nfinity.blogspot.com.es/2012/05/ares-p2p-historia.html

[32] Lenguaje Delphi, http://www.larevistainformatica.com/Delphi.htm

[33] Lenguaje Delphi, Wikilibros (2013), http://es.wikibooks.org/wiki/Lenguaje_Delphi

[34] Lenguaje HTML, Wikipedia (2013), http://es.wikipedia.org/wiki/HTML

[35] Lenguaje SQL, Wikipedia (2013), http://es.wikipedia.org/wiki/SQL

[36] Manual de Ares, http://blog.uptodown.com/ares-a-fondo-descubre-todo-lo-que-te-ofrece-el-cliente-p2p-mas-descargado-de-la-historia/

[37] Manual de Ares, http://www.slideshare.net/lantejuela/manual-de-ares

[38] Manual de PHP, http://php.net/manual/es/langref.php

[39] Manual de XAMPP, http://www.apachefriends.org/es/faq-xampp.html

[40] Redes P2P, http://www.elotrolado.net/wiki/Todo_sobre_P2P

[41] Redes P2P, Wikipedia (2013), http://es.wikipedia.org/wiki/Peer-to-peer

[42] Servidor Web, http://www.misrespuestas.com/que-es-un-servidor-web.html

[43] Servidor Web, Wikipedia (2013), http://es.wikipedia.org/wiki/Servidor_web

[44] Tutoriales de Ares y Ares Galaxy, http://www.tutorial-enlace.net/tutoriales-Ares/tipo-Iniciacion%20o%20explicacion-pagina-1-Fecha.html

[45] XAMPP, http://es.wikipedia.org/wiki/XAMPP

Universidad de Alcalá
Escuela Politécnica Superior

ESCUELA POLITÉCNICA SUPERIOR