Proyecto de fin de carrera
(Universitat Oberta de Catalunya)

"Sistema para la protección de la privacidad de los usuarios de los motores de búsqueda de Internet"

Autor: José Manuel Puga Moreira
Titulación: Ingeniería Informática
Área: Seguridad
Consultor: Jordi Castellà-Roca
Fecha: 12/06/2011

Introducción

- Los WSE (World Search Engines) compiten por ofrecer los mejores resultados.
- Necesitan para ello “conocer” quién hace la consulta.
- Los WSE mantienen un perfil de cada usuario.
- De esta forma pueden distinguir consultas ambiguas
  - Ej.: Júpiter (planeta) o Júpiter (Dios romano.)
- E incluso ofrecer publicidad personalizada.

Introducción

- Violación de la privacidad de los usuarios.
- Servicio vs Privacidad
- Actualmente existen herramientas para proteger la privacidad.
- Inconvenientes de estas aplicaciones:
  - Usuario anónimo → WSE no puede ofrecer un servicio óptimo.
  - Elevado tiempo de respuesta.

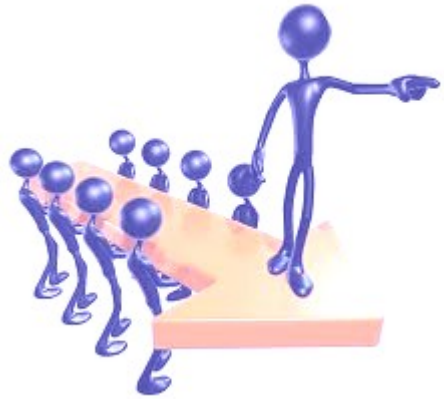
Introducción: justificación

- Obtener un sistema que permita **mantener la privacidad** de un usuario sin que afecte demasiado a los resultados de las búsquedas.

- Conseguir un sistema que **mejore los tiempos** de búsqueda de las herramientas actuales que protegen la privacidad del usuario.

Introducción: objetivos

- El sistema implementado debe ser utilizable en la práctica.
- El WSE no será capaz de obtener un perfil exacto del usuario. Aún así, el perfil que obtenga debe ser útil para proveer un servicio adecuado.



- Deberá ser posible identificar a un usuario que envíe una consulta que vulnere la ley.

Introducción: planificación

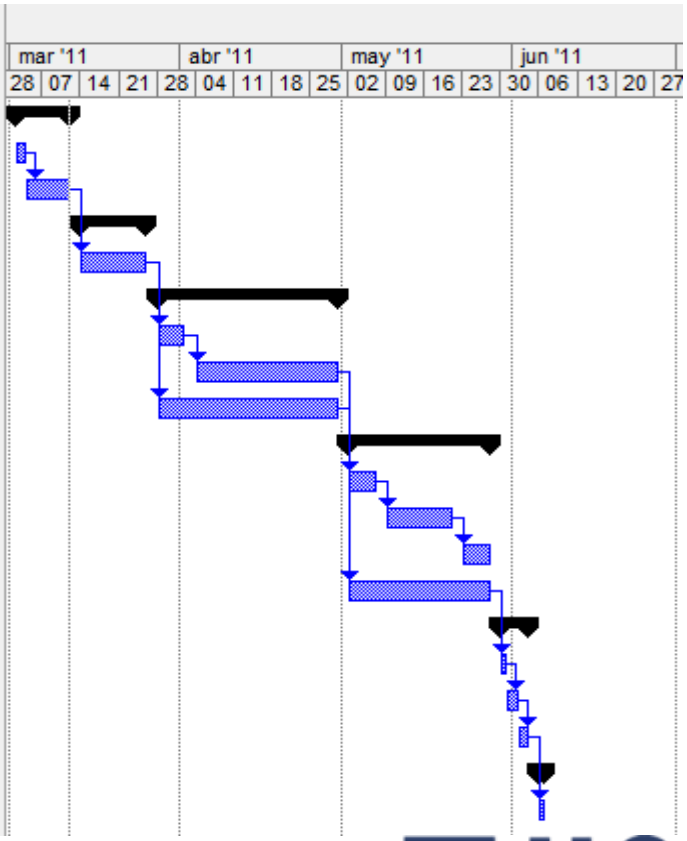
- La aplicación desarrollada se desglosa en fases, en las que cada una utiliza como entrada los productos obtenidos en la fase anterior.
- La evolución de la memoria y de la aplicación sigue un camino paralelo.

Fase	Identificación	Descripción
1	Planificación	Se delimita y define el proyecto con respecto al ámbito, el alcance, los objetivos y la elaboración del plan de trabajo.
2	Descripción esquema	Se detalla el funcionamiento del protocolo.
3	Diseño e implementación	- Decisiones de diseño: herramientas. - Diseño de la arquitectura: componentes. - Diagrama de clases. - Desarrollo del código.
4	Pruebas	- Diseño de los juegos de pruebas. - Realización de las pruebas. - Estudio de los resultados.
5	Adecuaciones finales	- Revisión del código - Redacción de las conclusiones - Revisión de la memoria
6	Entrega del PFC	Se realiza la presentación del PFC.

Introducción: planificación

- Cada fase se divide en diferentes tareas.
- La duración de cada fase viene determinada por la fecha de entrega de cada PAC.

	Nombre de tarea	Duración	Comienzo	Fin
1	**Preparación proyecto**	**8 días**	**mié 02/03/11**	**vie 11/03/11**
2	Elaboración del plan de trabajo	2 días	mié 02/03/11	jue 03/03/11
3	Redacción de la introducción	6 días	vie 04/03/11	vie 11/03/11
4	**Descripción esquema**	**10 días**	**lun 14/03/11**	**vie 25/03/11**
5	Redacción de la descripción del esquema	10 días	lun 14/03/11	vie 25/03/11
6	**Diseño e implementación del esquema**	**25 días**	**lun 28/03/11**	**vie 29/04/11**
7	Diseño	5 días	lun 28/03/11	vie 01/04/11
8	Implementación	20 días	lun 04/04/11	vie 29/04/11
9	Redacción del diseño y la implementación	25 días	lun 28/03/11	vie 29/04/11
10	**Pruebas**	**20 días**	**lun 02/05/11**	**vie 27/05/11**
11	Diseño del juego de pruebas	5 días	lun 02/05/11	vie 06/05/11
12	Realización de las pruebas	10 días	lun 09/05/11	vie 20/05/11
13	Estudio de los resultados	5 días	lun 23/05/11	vie 27/05/11
14	Redacción de las pruebas y del estudio	20 días	lun 02/05/11	vie 27/05/11
15	**Adecuaciones finales**	**5 días**	**lun 30/05/11**	**vie 03/06/11**
16	Preparación del código a entregar	1 día	lun 30/05/11	lun 30/05/11
17	Redacción de las conclusiones	2 días	mar 31/05/11	mié 01/06/11
18	Revisión final de la memoria	2 días	jue 02/06/11	vie 03/06/11
19	**Entrega del PFC**	**1 día**	**lun 06/06/11**	**lun 06/06/11**
20	Presentación de la defensa del PFC	1 día	lun 06/06/11	lun 06/06/11



Descripción del esquema

- El esquema se basa en el uso de las redes sociales.
- Las redes sociales mantienen conectados usuarios con intereses comunes.
- Una consulta creada por un usuario puede ser enviada al WSE por el propio usuario o ser reenviada por la red social para que otro usuario la envíe al WSE.
- Este hecho no impide al WSE de crear un perfil del usuario. Pero el perfil que elabore será uno distorsionado.
- El perfil será utilizable (a un determinado nivel) para ofrecer un servicio adecuado al usuario, ya que los grupos de usuarios en una red social comparten intereses y aficiones.

Descripción del esquema

- **Tipos de usuarios:**
  - *Honesto*. Sigue el protocolo propuesto.
  - *Egoísta*. Aquel que debiendo enviar una consulta al WSE la reenvía a uno de sus vecinos, o que descarta todas las consultas recibidas de sus vecinos.

- **Protocolo**:
  - ¿Como decide un usuario que crea una consulta si reenviarla o enviarla al WSE? → ***Función de selección de usuario***
  - ¿Como decide un usuario que recibe la consulta si reenviarla o enviarla al WSE? → ***Función de reenvío de consulta***
  - ¿Como aislar los usuarios egoístas? → ***Función del nivel de egoísmo***

Descripción del esquema: función del nivel de egoísmo

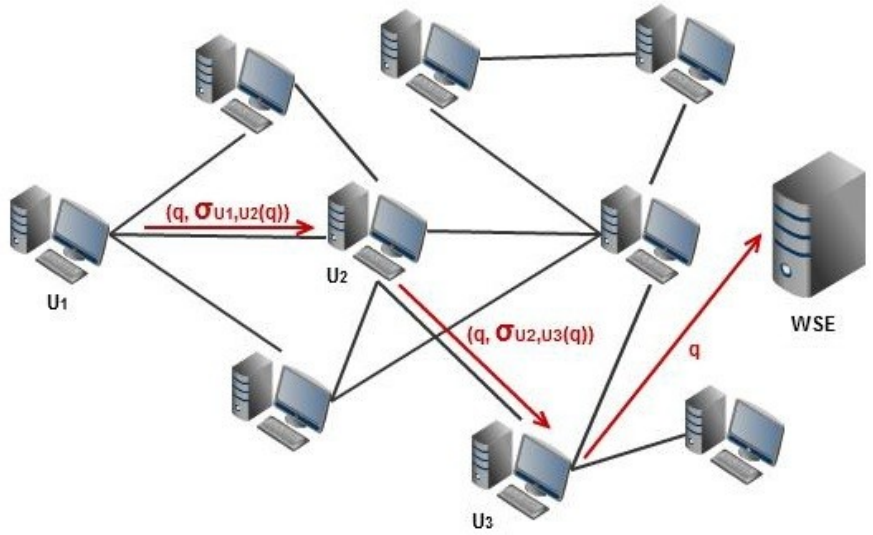
- **Objetivo**: penalizar a los usuarios egoístas.
- Cada usuario asigna una probabilidad de aceptar consultas a cada uno de sus vecinos.
- Probabilidad inicial = 1.
- **Ejemplo de funcionamiento**:
  - Un usuario U_1 envía una consulta a uno de sus vecinos U_2.
  - Si U_2 acepta la consulta:
    - U_1 aumenta la probabilidad de aceptar consultas de U_2.
    - U_2 disminuye su probabilidad de aceptar consultas de U_1.
  - Si U_2 rechaza la consulta:
    - U_1 disminuye su probabilidad de aceptar consultas de U_2.

Descripción del esquema: firma digital

- Una consulta de un usuario puede ser enviada por cualquier miembro del grupo de la red social.
- Este hecho puede provocar que un usuario envíe al WSE una consulta que vulnere la ley y que él no haya creado.
- Para solventar el problema el protocolo utiliza certificados para probar la transacción entre dos usuarios.
- En un entorno real sólo tendría validez legal la firma electrónica avanzada, en donde la clave pública de cada usuario estaría certificada por una CA (Autoridad Certificadora).
- Para los certificados de transacción se utiliza XML “Enveloped”.

Descripción del esquema: firma digital

- U_1 crea una consulta que será enviada a U_2.
- U_1 envía el par $(q, \sigma_{U1,U2}(q))$ donde $\sigma_{U1,U2}(q)$ es el certificado que prueba la transacción entre U_1 y U_2.
- El certificado es firmado por U_1 utilizando su clave secreta.
- El usuario U_2 recibe la consulta y el certificado, verificando la firma con la clave pública de U_1.
- Si todo es correcto U_2 guarda en disco el certificado.



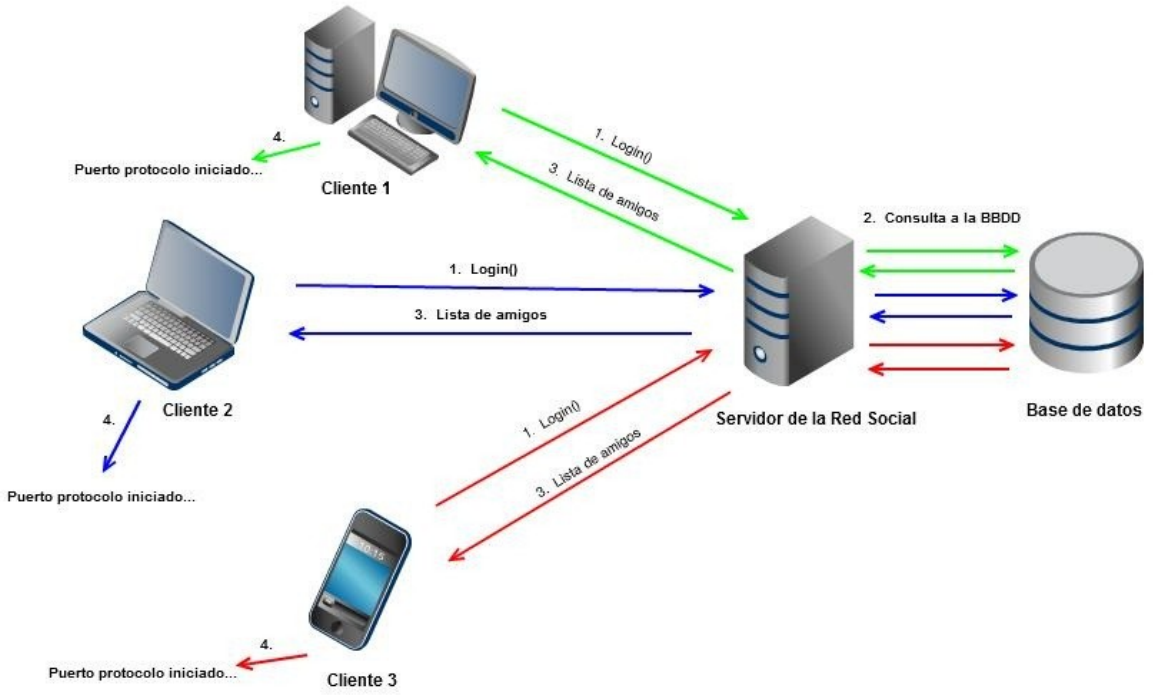
Descripción del esquema: protocolo

- **Usuario que crea consulta:**
  - Ejecuta función de selección de usuario.
  - Si es él el responsable envía la consulta al WSE.
  - En caso contrario firma la consulta con su clave privada y la envía al usuario que ha determinado la función.

- **Usuario que recibe una consulta:**
  - Acepta o rechaza la consulta según la probabilidad de aceptar consultas que tenga asignado al usuario que le envía la consulta.
  - Verifica la firma de la consulta, y si es válida, guarda el certificado de la transacción en disco.
  - Ejecuta función de reenvío de usuario.
  - Si es él el responsable envía la consulta al WSE.
  - Si no, firma la consulta con su clave privada y la envía al usuario que ha indicado la función de reenvío.

Diseño

- **Componentes**:
  - *Servidor*. Simula el servicio de una red social.
  - *Base de Datos*. Contiene información de los usuarios de la red social.
  - *Cliente*. Usuario de la red social y a su vez usuario del protocolo.



Implementación

- **Lenguaje de programación**: JAVA.
- **Base de datos**: MySQL.
- **Comunicaciones**: Sockets.
- **Clases implementadas**:
  - *Server*. Simula el comportamiento de la red social.
  - *ServerThread*. Procesa una petición que llega al servidor.
  - *Client*. Simula un usuario de la red social y del protocolo.
  - *Protocol*. Implementa el funcionamiento del esquema diseñado.
  - *Query*. Describe una consulta procesada por el protocolo.
  - *User*. Representa e identifica un usuario para ejecutar el protocolo apropiadamente.
  - *WSE*. Implementa las funciones para enviar una consulta al WSE.
  - *Response*. Representa la respuesta del WSE.
  - *XML*. Funciones relacionadas con la creación y manipulación XML.
  - *Crypto*. Métodos relacionados con la firma digital de una consulta.

Pruebas

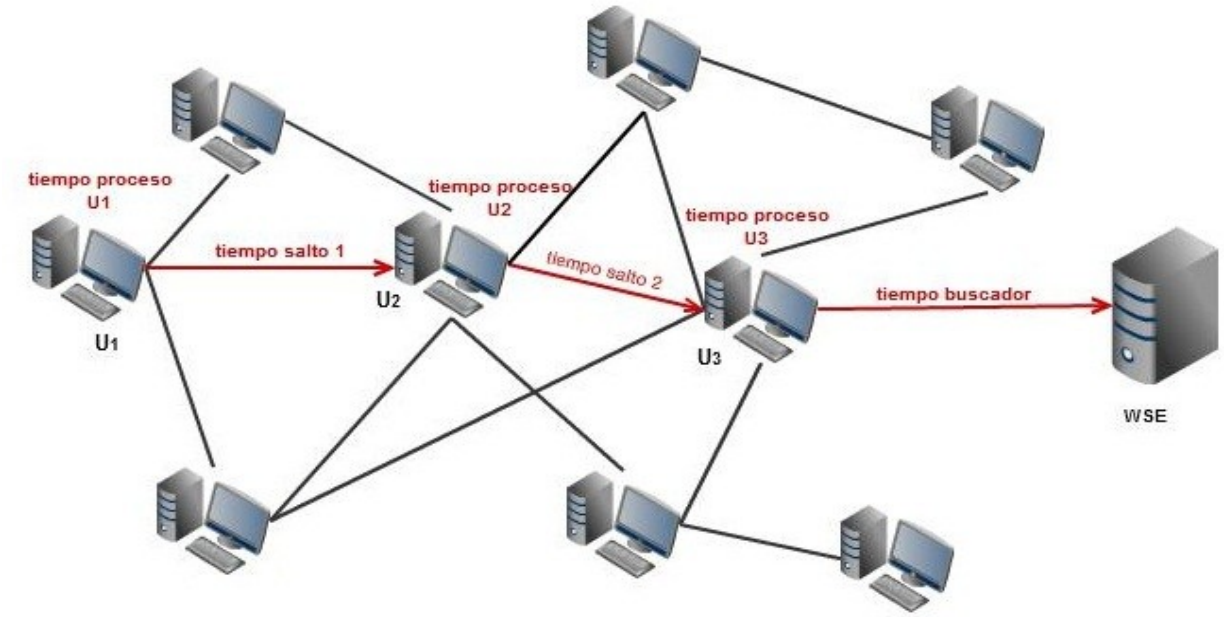
- Red social de 20 y 100 usuarios.
- Uso de la aplicación ***ReadGraph*** para importar los datos de un fichero que representa una red social a la base de datos.
- Pruebas realizadas en una sola máquina:
  - *Procesador: Intel Core 2 Duo T7500 2'2 Ghz*
  - *Memoria RAM: 2Gb*
  - *Arquitectura: 64 bits*
  - *Sistema Operativo: Windows 7 Ultimate 64 bits*

- No se realiza el envío real al WSE.
- Datos interesantes: número de saltos de cada consulta y tiempo de proceso de cada usuario.
- Clase ***Statistics***. Utilizada para guardar información de un usuario del protocolo, útil para efectuar análisis.
- Clase ***Test***. Script que permite configurar los parámetros de una prueba e iniciar el proceso.

Pruebas

- Estimación del tiempo de una consulta en un entorno real:
  - Tiempo medio de comunicación entre dos usuarios en una red P2P: 530 ms.
  - Tiempo medio de una búsqueda directa al WSE: 400 ms.

$$t_{consulta} = t_{procesoUsuario} * (n^{o}\ medio\ saltos + 1) + 2 * (t_{salto}) * (n^{o}\ medio\ saltos) + t_{respuestaWSE}$$



Resultados

- Se distribuyen las consultas de forma equitativa por la red.
- Siguiendo un comportamiento normal todos los usuarios mantienen una alta predisposición a aceptar consultas de sus vecinos.
- El sistema aísla correctamente a los usuarios que se comportan de forma egoísta → El protocolo no protege su identidad frente al WSE.
- Usuarios con menos vecinos → más probabilidad de enviar consultas directamente al WSE.
- Red de 20 usuarios → 2,11 saltos de media por consulta.
- Red de 100 usuarios → 3,09 saltos de media por consulta.
- Estimación del tiempo medio de búsqueda en un entorno real:
  - Número medio de saltos por consulta: 2,11525
  - Tiempo medio de proceso de usuario: 0,105488 segundos
  - Total búsqueda = 2,9708 segundos

Conclusiones

- Se han logrado con éxito **los objetivos del PFC** siguiendo la planificación inicial.
- Se ha obtenido un mejor tiempo de búsqueda en comparación con otras implementaciones.
- Datos satisfactorios en cuanto al número promedio de saltos.
- El aumento en el número de usuarios de la red social implica un aumento en el número medio de saltos por consulta.
- Implementación **totalmente operativa** en un entorno real:
  - Es necesario configurar las tablas NAT de los routers.
  - Se debe disponer de una máquina que simula el servidor de una red social.
- La existencia de una CA complicaría la puesta en marcha en un escenario real.

Fin de la presentación