ABADAM: Un modelo de seguridad para el acceso a bases de datos utilizando agentes móviles

David Antonio Franco Borré[1]
Yasmín Moya Villa[2]
Moises Ramón Quintana Alvarez[3]
Miguel Angel García Bolaños[4]
Julio Cesar Rodríguez Ribón[5]

Resumen

En este artículo se analizan aspectos concernientes a los agentes móviles y los problemas de seguridad que estos conllevan, presentando un modelo o esquema para acceder a una base de datos que permita mejorar los tiempos de respuestas en la red y crear un modelo de seguridad que colabore en el proceso de autenticación de los usuarios. El modelo propuesto se ha denominado ABADAM (Acceso a BAses de Datos utilizando Agentes Móviles), ABADAM se implementó utilizando: el lenguaje de programación Java, la API Aglets y MS-SQL Server.

ABADAM brinda un alto nivel de seguridad para proteger al servidor de base de datos de ataques de agentes maliciosos, permitiendo el acceso solo a los agentes válidos. La seguridad está basada fundamentalmente en el proceso de autenticación o identificación de los agentes que lleguen al servidor, este proceso será capaz de detectar agentes no válidos y los eliminará al considerarlos agentes maliciosos. Además si algún usuario mal intencionado creara un agente móvil con login y password válidos no podrá realizar la autenticación al no portar la credencial que lo acredite como un agente confiable y sería destruido inmediatamente.

Palabras clave: Agentes Móviles, Aglets, Base de Datos Disribuida, Java, MS SQL Server.

ABADAM: A security model to databases access using mobile agents

Abstract

This article analyze aspects related to the mobile agents and security problems these entail, presenting a model access to databases that allow in a high degree to improve the response times in the network and to create a security model that collaborates with user's authentication process. The proposed model has denominated ABADAM (Access to Databases using Mobile Agents), ABADAM was implemented using: the Java programming language, API Aglets and MS-SQL Server.

ABADAM offers a high level of security to protect the database server from malicious agents attacks, allowing access to the valid agents only. The security is based fundamentally on the authentication or identification process of agents that arrive to the server, this process will be able to detect non-valid agents and it will eliminate when considering them malicious agents. In addition if some ill-disposed user created a mobile agent with login and password valid will not be able to make the authentication when it will not be carrying the credential that credits it as a reliable agent and it would be destroyed immediately.

Key words: Aglets, Distributed Databases, Java, Mobile Agents, MS SQL Server.

1. Introducción

En los últimos años se están utilizando los agentes móviles como herramienta de programación en ambientes distribuidos e Internet [1][2][19]. El crecimiento acelerado de las redes ha conducido a un mayor desarrollo de sistemas de información distribuidos donde se puede acceder a bases de datos distribuidas. La necesidad de tener los programas cerca de los datos para aumentar el rendimiento de las aplicaciones origina tecnologías para ambientes distribuidos, con la aparición de los agentes móviles y lenguajes multiplataformas como Java, los ambientes de redes distribuidas han demostrado mejorar el desempeño de las aplicaciones permitiendo una interacción local en el procesamiento de la información, reemplazando la conectividad constante en la red. A pesar del desempeño que se ha logrado con el uso de agentes móviles para acceder a bases de datos distribuidas, el crecimiento de esta tecnología se ha visto retardada por problemas de seguridad.

El aporte de este trabajo consiste en diseñar un modelo base para implementar un sistema

[1] Profesor de la Facultad de Ingeniería de la Universidad de Cartagena.
[2] Profesora de la Facultad de Ingeniería de la Universidad de Cartagena y la Universidad Tecnológica de Bolívar.
[3] Profesor de la Facultad de Ingeniería de la Universidad Tecnológica de Bolívar.
[4] Profesor de la Facultad de Ingeniería de la Universidad de Cartagena.
[5] Profesor de la Facultad de Ingeniería de la Universidad de Cartagena.

Artículo recibido en Enero de 2008, aprobado para publicación en Abril de 2008

de agentes móviles que colaboren en el proceso de autenticación para acceder a un servidor de base de datos, su diseño flexible, permite aplicarse a cualquier arquitectura de bases de datos que soporte instrucciones SQL y se pueda conectar mediante un driver ODBC.

2. Agentes móviles

Un Agente Móvil [3] es un objeto especial que tiene un estado de datos (otros objetos no agentes, estructuras y bases de datos), un estado de código (las clases del agente y otras referencias a objetos) y un estado de ejecución (el control de procesos que se ejecutan en el agente). La característica principal de un agente móvil es que accede a un conjunto de métodos que le permiten moverse de un servidor a otro. Un agente móvil tiene: un programa que define las actividades que ejecutará y un grado de inteligencia que, además de ayudarlo a interactuar con los usuarios, lo ayuda a la resolución de tareas adversas de su entorno. Las características básicas de un agente móvil son: autonomía, movilidad, concurrencia, direccionabilidad, continuidad, reactividad, sociabilidad y adaptabilidad.

Un agente para que sea catalogado como agente móvil debe ser capaz de ejecutarse en diferentes computadoras, es decir, que pueda suspender su ejecución en un servidor y reanudarla en otro servidor una vez que se haya desplazado a este. Un agente móvil es una entidad que tiene cinco atributos: estado, implementación, interfaz, identificador y un ente principal (Figura 1). Cuando el agente se mueve en la red toma estos atributos y viaja con ellos. El agente necesita al estado para reanudar el cómputo después de viajar, la implementación se necesita para la ejecución del agente independiente de su ubicación, la interfaz se necesita para comunicación del agente, el identificador se necesita para reconocer y ubicar al agente mientras viaja, y el ente principal determina la responsabilidad legal.

Figura 1. Agente móvil: atributos básicos de los agentes móviles. Fuente: Programming and Deploying Java Mobile Agents with Aglets [3].

Una de las principales motivaciones para el uso de los agentes móviles es su aplicabilidad en sistemas distribuidos y se tienen siete ventajas al utilizar agentes móviles [4]:

1. Reducen el tráfico de red.
2. Superan el estado latente de las redes.
3. Encapsulan protocolos.
4. Se ejecutan de forma asíncrona y autónoma.
5. Se adaptan dinámicamente.
6. Son heterogéneos por naturaleza.
7. Son robustos y tolerables a fallas.

Entre las aplicaciones de los agentes móviles, se encuentran las siguientes [5]:

- Recolección de datos de distintos sitios.
- Búsqueda de filtrado.
- Monitorización.
- Comercio electrónico.
- Procesamiento paralelo.
- Distribución masiva de información.
- Interacción y negociación.
- Cálculos en multiprocesos.
- Redes parcialmente desconectadas.
- Entretenimiento y entrega de correo inteligentemente.

Una de las principales plataformas para el desarrollo de agentes móviles es Java dado que maneja de manera eficiente la seguridad y portabilidad, por su característica de lenguaje interpretado (su salida no es un ejecutable, sino un código binario ejecutado por una máquina virtual que emula el intérprete de Java). Vale la pena mencionar algunos sistemas que permiten la implementación de agentes móviles, como lo son: Aglets, Java-to-go, ARA, Voyager, Mole, Agent TCL, Odyssey, TACOMA, entre otros.

3. Seguridad de los agentes móviles

Las aplicaciones basadas en agentes móviles no deben implementarse hasta que se tomen medidas mínimas de seguridad en la red, como tener un firewall y brindar protección a los servidores contra código malicioso que viene en un agente móvil, todo esto se puede prevenir con controles y protecciones adecuadas para el acceso en ambientes de ejecución. El tema de la seguridad de agentes móviles ha sido un factor fundamental para que no sea aceptado en su totalidad en Internet, pero hay platafor-

mas para la creación de agentes móviles que permiten alcanzar un nivel de seguridad aceptable, una de ellas son los Aglets que son capaces de restringir el acceso de los agentes generalizándolos en dos grupos: confiables y no confiables.

Los agentes móviles son programas que viajan con su código y datos desde un computador a otro a través de Internet o cualquier red, lo que conlleva a que surjan varios problemas de seguridad porque tanto los agentes móviles como los servidores con que interactúan son vulnerables a ataques y brechas de seguridad. Algunos problemas de seguridad que se pueden presentar son: Autenticación, Autorización y Confidencialidad. La movilidad de los agentes y la interacción con otras máquinas de la red, puede hacer que surjan algunos requerimientos de seguridad en el acceso a bases de datos distribuidas que podrían ser clasificados de dos tipos. El primero de ellos se refiere a los problemas de seguridad inherentes a los sistemas de agentes móviles y en el segundo se refiere a los aspectos que se requieren para garantizar la seguridad en el ámbito de las Bases de Datos. A continuación se expone con mayor detalle cada uno de estos problemas.

3.1. Amenazas de seguridad inherentes al sistema de agentes móviles

La seguridad ha sido tradicionalmente un inconveniente en los sistemas informáticos, pero el sistema de agentes móviles por sus características presenta un nuevo tipo de inconvenientes de seguridad, como lo son el problema del agente malicioso y el problema del servidor malicioso [6] [7].

3.1.1. Problema del agente malicioso

Cuando los agentes móviles llegan al servidor, necesitan algunos permisos de usuario para poder ejecutar su código. Normalmente el control de acceso a los equipos de una red se hace por medio de contraseñas, en donde cada usuario se autentica al inicio con su *login* y *password*; pero este esquema funciona para sistemas estáticos y no para agentes móviles, puesto que el agente debe llevar una contraseña para cada sitio de su itinerario, lo que aumenta su tamaño y así mismo incrementa los requerimientos de seguridad para poder proteger esas contraseñas.

Entre los agentes pueden existir agentes maliciosos que realizan acciones no deseadas o destructivas, como el acceso no autorizado, alteración de la información, sobrecargar o causar incluso daños a la integridad del servidor.

La solución que se le da normalmente al problema del agente malicioso se lleva a cabo por medio de restricción de privilegios a los agentes, pero esto podría ocasionar que los usuarios que traten de acceder a la base de datos mediante un agente no pueda hacerlo, por tal razón es necesario buscar mecanismos alternos de protección que brinden seguridad y permitan la realización de las tareas de los agentes.

3.1.2. Problema del servidor malicioso

Un servidor malicioso es aquel que podría espiar ó alterar el código y/o los datos de un agente, proporcionar llamadas falsas al sistema, retornar al agente datos no deseados, entre otros. Cuando un agente móvil llega al servidor, ya no depende de la máquina que lo envió, por lo tanto, el servidor receptor debe instanciar la clase que dio origen al agente, leer el estado de datos que se encuentra serializado y ejecutarlo, por lo que el código y datos del agente móvil quedan totalmente expuestos.

Los ataques a los que podrían estar sometidos los agentes móviles por parte de un servidor malicioso son la clonación de un agente para robarle o modificarle la información que el agente porte, interrumpir o cambiar el itinerario del agente, o simplemente destruir al agente.

3.2. Problemas en el acceso a bases de datos con agentes móviles

Con el desarrollo que ha tenido Internet y las comunicaciones se ha incrementado exponencialmente el uso de bases datos distribuidas apareciendo los agentes móviles como una alternativa para mejorar el desempeño en este tipo de aplicaciones distribuidas.

El modelo de agentes móviles va teniendo cada vez un mayor auge, tanto que existen en el mercado herramientas para crear agentes móviles, se cuenta con estándares, plataformas y productos que facilitan su desarrollo. Sin embargo, debido a problemas de seguridad el acceso a bases de datos distribuidas mediante agentes móviles no ha tenido el auge esperado. Por tanto se hace necesario buscar soluciones para lograr un nivel de seguridad adecuado para pro-

teger el acceso a las bases de datos y la forma de alcanzarlo es logrando mejorar aspectos de seguridad como son [6]: la autenticación, autorización, confidencialidad e integridad.

4. Aglets

Antes de seleccionarse Aglets para la implementación de los agentes móviles se analizaron otras plataformas de agentes como Concordia, que es un framework implementado en Java creada por Mitsubishi Electric que solo provee un modelo para trabajar con aplicaciones móviles [16]. JATLite es otro framework que provee la funcionalidad básica para construir aplicaciones de sistemas multiagentes, donde tanto el servidor de nombres como la funcionalidad para el intercambio de mensajes son centralizados lo que puede producir un cuello de botella en el sistema cuando se trata de una aplicación real con varios agentes ejecutándose simultáneamente [17]. Y también tenemos a JADE, que es una plataforma implementada completamente en Java, cuya finalidad es simplificar el desarrollo de sistemas multiagentes a través de un conjunto de sistemas, servicios y agentes, pero está más orientado a los agentes inteligentes [18].

Uno de los principales marcos de trabajo para agentes móviles, desarrolladas en Java, son los Aglets, creados por el equipo del laboratorio de desarrollo de IBM en Tokio. Un Aglet tiene una serie de estados que definen su funcionamiento. Los principales eventos en la vida de un Aglet [3] son: Creación o Clonación; Liberación de recursos; Movilidad; y Persistencia (permite la desactivación y activación del Aglet).

El servidor crea un contexto para los Aglets. Allí serán insertados todos los Aglets que llegan al servidor, además tiene un monitor de red que permite monitorear la red buscando por otros Aglets, y un administrador de seguridad que protege al sitio (Figura 2).

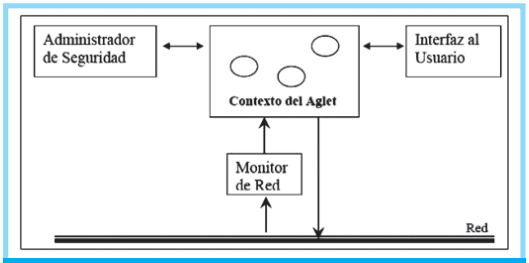

Figura 2. Contexto de los Aglets, Interfaz al usuario y Monitor de red. Fuente: Programming and Deploying Java Mobile Agents with Aglets [3].

La estructura definida en los Aglets permite que estos actúen autónomamente colaborando con otros Aglets. La colaboración es una necesidad en los agentes porque en general ninguno está aislado ni puede realizar tareas complejas solo. La movilidad y el paso de mensajes son características importantes para la colaboración. Los Aglets no pueden migrar con su estado: memoria, pila de ejecución y registros, ya que están implementados en Java donde un programa no tiene acceso directo para manipular su estado (definido en el modelo de seguridad de la máquina virtual Java - JVM). Por lo tanto, cuando el Aglet va a ser transferido, debe almacenar todo lo necesario para luego retomar su estado cuando es reactivado, esto es llamado movilidad débil.

Al utilizar los Aglets, se tiene un modelo de fácil comprensión para la programación de agentes móviles sin tener que hacer modificaciones en la máquina virtual de Java o código nativo. También se tiene el soporte de una comunicación dinámica que permite a los agentes comunicarse con otros agentes tanto conocidos como desconocidos. Además, se proveen mecanismos de seguridad que son de fácil comprensión y lo suficientemente simples para permitir que los usuarios finales puedan especificar una serie de restricciones de acceso a los agentes móviles.

5. Modelo Propuesto: ABADAM

Después de haberse analizado los aspectos concernientes a los agentes móviles y los problemas de seguridad que estos conllevan se presentará en esta sección un modelo de seguridad para el sistema de agentes móviles que acceden a una base de datos distribuida, además de brindar un proceso que colabore en la autenticación de los usuarios que quieren acceder al servidor de base de datos mediante este sistema.

El modelo propuesto se denomina ABADAM (Acceso a Bases de Datos utilizando Agentes Móviles), ABADAM es un modelo implementado utilizando tres grandes componentes: el lenguaje de programación JAVA, el API AGLETS y SQL. El sistema propuesto está conformado por un sistema de agentes móviles, un modelo de seguridad y la base de datos (Figura 3).

Uno de los principales marcos de trabajo para agentes móviles, desarrolladas en Java, son los Aglets, creados por el equipo del laboratorio de desarrollo de IBM en Tokio.

ABADAM brinda un componente adicional de seguridad convirtiéndose también en un proceso colaborador en la autenticación de los usuarios.

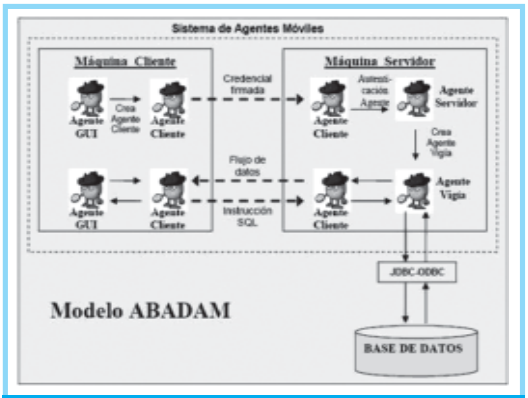

Figura 3. Modelo propuesto: ABADAM.

5.1. Sistema de agentes móviles

La implementación de los agentes que intervienen en el modelo se realizó con Aglets y en las diferentes máquinas que se utilizaron para probar el modelo se instaló Tahiti (Entorno de ejecución de los agentes). El sistema de agentes móviles propuesto en este proyecto está clasificado de la siguiente forma:

- *Agente GUI:* La interfaz gráfica del usuario de ABADAM es invocada por el agente GUI que consiste en un agente estacionario, es decir, que su entorno de ejecución es la máquina desde la cual se conecta el usuario, permitiendo al usuario digitar su usuario, contraseña y dirección IP del servidor. Estos datos son entregados al agente móvil cliente para que se conecte a la base de datos, si el agente cliente regresa con éxito de la conexión se le solicitará que proceda a la captura de la instrucción SQL para realizar la operación deseada en la base de datos y desplegar los resultados de la transacción.
- *Agente Servidor:* El agente servidor es un agente estacionario ubicado en la máquina donde se encuentra la base de datos, este agente tiene la función de crear un agente vigía por cada agente cliente válido que llegue al servidor a establecer una sesión con la base de datos.
- *Agentes clientes:* El agente cliente es un agente móvil creado por la interfaz del usuario y se comunica con el agente servidor, el cual le asigna un agente vigía con quien tratará el cliente mientras dure la sesión.
- *Agentes vigías:* El agente vigía es un agente estacionario creado por el agente servidor por cada agente cliente valido que llegue al servidor. Este agente vigía tiene la función de comunicarse con el agente cliente mientras dure la sesión del usuario para que se de el intercambio de mensajes entre el cliente y el servidor.

Por cada usuario conectado a la base de datos del servidor, existirá un agente móvil cliente y por consiguiente existirá un agente vigía por cada agente cliente. De la misma manera si un usuario cierra su sesión o interfaz se destruirá el agente cliente y vigía correspondiente a ese usuario.

5.2. Modelo de Seguridad

El modelo de seguridad propuesto en ABADAM brinda un alto nivel de seguridad para proteger al servidor de base de datos de agentes maliciosos dando solución al problema del agente malicioso, expuesto en la sección 3.1.1, debido a que solo tendrán acceso aquellos agentes válidos o confiables. El modelo de seguridad de ABADAM está basado fundamentalmente en el proceso de autenticación o identificación de un agente válido que llega al servidor, este proceso será capaz de detectar agentes no válidos y los considerará como agentes maliciosos, los cuales eliminará en el mismo instante que lleguen al servidor. Además ABADAM brinda un componente adicional de seguridad convirtiéndose también en un proceso colaborador en la autenticación de los usuarios, debido a que si algún usuario mal intencionado creara un agente móvil cliente portando un login y password válido no podría realizar la autenticación al no portar la credencial que lo acredite como un agente cliente confiable y será destruido inmediatamente.

La plataforma que se utilizó para crear el sistema de agentes móviles de ABADAM fue Aglets 2.0.2, entre los servicios de seguridad que esta plataforma cuenta no presenta la criptografía [9], por lo que el modelo de seguridad se diseño basado en el esquema de firmas digitales y listas de control de acceso, para evitar que agentes maliciosos no accedan la base de datos localizada en el servidor.

ABADAM implementa un sistema de firma digital utilizando un algoritmo de la familia SHA (Secure Hash Algorithm) [10], para establecer una comunicación segura entre el cliente y el servidor se genera una credencial firmada digitalmente. Esta credencial debe contener el nombre del host de donde proviene el agente móvil, la dirección IP, nombre de la clase u objeto del agente, y la identificación interna del

agente. ABADAM cuenta con varios niveles o momentos de seguridad para detectar si un agente es malicioso (ver figura 4).

En la figura 4 están enumerados 13 pasos que se dan, en su orden, al momento en que un cliente desea conectarse a la base de datos, si el cliente no resulta confiable, estos pasos no se darán en su totalidad y el agente cliente será eliminado dependiendo del caso.

El primer nivel de seguridad de ABADAM se da en el momento que el agente móvil, despachado por el cliente, llega al servidor (esto se evidencia en la segunda línea de logs de la Figura 5) y consiste en validar la credencial por su firma digital, si el agente servidor no puede validar la firma se asumirá que está tratando con un agente malicioso y lo eliminará instantáneamente, de lo contrario el proceso seguirá su curso.

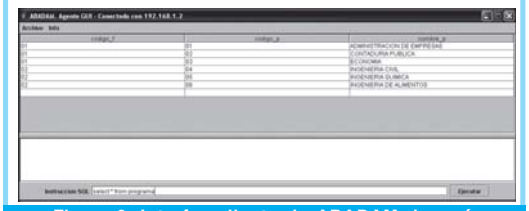

Figura 6. Interfaz cliente de ABADAM después de haber realizado una consulta.

El segundo nivel de seguridad ocurre una vez se ha comprobado la fidelidad de la firma digital, en ese momento se realiza una validación adicional de seguridad que consiste en verificar mediante el nombre del host y la dirección IP del cliente coincida con algunas de las entradas de la lista de control de acceso ubicada en el servidor, si el agente cliente no cumple con esta validación será destruido por considerarse un agente malicioso. En este nivel se controla que agentes maliciosos o no válidos se apoderen de un agente válido y traten de acceder al servidor.

El tercer nivel de seguridad consiste en crear una lista dinámica con la identificación interna de cada agente después de haber superado los niveles explicados en los dos párrafos anteriores, la identificación interna de cada agente corresponde a un único número de identificación asignado al agente, de tal manera que no pueden existir agentes válidos que no estén en esa lista, de lo contrario serían considerados agentes maliciosos y serían eliminados del servidor.

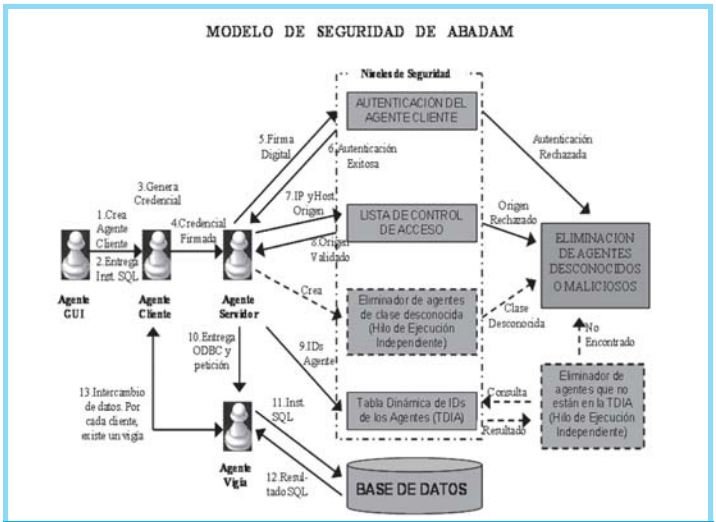

Figura 4. Modelo de seguridad.

Un cuarto nivel de seguridad independiente a los mencionados en los tres párrafos anteriores consiste en un hilo de ejecución del agente servidor que se encarga de destruir cualquier agente que no pertenezca a un nombre de objeto válido (entiéndase como objeto válido los tipos de agente manejados por ABADAM), este control de seguridad surge para eliminar cualquier clase de agente malicioso que quiera atacar el servidor.

El Sistema de agentes propuesto en este modelo evita que agentes no deseados o malintencionados lleguen al servidor, con lo cual elimina cualquier posibilidad que el servidor colapse por sobrecarga de agentes no válidos. Los aspectos descritos resaltan claramente el nivel de seguridad brindado por el Sistema de Agentes, además de mejorar el rendimiento de la red al no mantener canales constantes entre cliente y servidor, aspecto que no es fácilmente manejado en la forma tradicional de comunicación cliente-servidor.

5.3. Acceso a la Base de Datos

En la Figura 6 se muestra la interfaz cliente de ABADAM, la cual captura la instrucción SQL y el agente cliente es el encargado transportarla hasta el servidor, que después de verificar la autenticidad del cliente, el servidor crea el agente (como se evidencia en la tercera línea de logs de la Figura 5) vigía para siga atendiendo al cliente.

El agente vigía es el que se conecta a la base de datos mediante el ODBC creado en el servidor de base de datos. A pesar que en este

El Sistema de agentes propuesto en este modelo evita que agentes no deseados o malintencionados lleguen al servidor, con lo cual elimina cualquier posibilidad que el servidor colapse por sobrecarga de agentes no válidos.

Lo novedoso de ABADAM es que un modelo de seguridad para acceder a bases de datos mediante agentes móviles.

proyecto se trabajó con el motor de MS-SQL Server, ABADAM está diseñado para adaptarse a cualquier motor de base de datos, basta con configurar el ODBC apropiado, ofreciendo un alto grado de flexibilidad al poder conectarse con una gran variedad de motores de bases de datos.

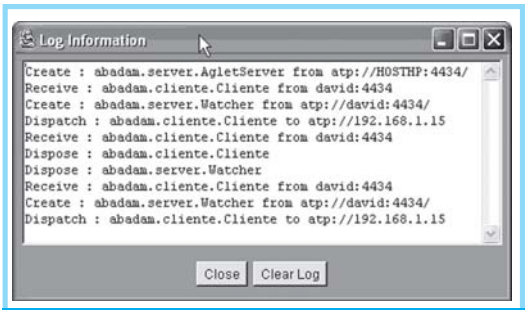

Figura 5. Logs generados por el servidor Tahiti.

Los servicios de ABADAM no actúan en el marco del proceso de autenticación de usuarios del servidor de base de datos, sino que implementa un sistema de seguridad previo que autentica los agentes clientes que portan el login y password para acceder a la base de datos, mejorando los niveles de seguridad en ambientes tan hostiles como son los canales de comunicación compartidos que se manejan en sistemas distribuidos.

Con su sistema de agentes, ABADAM evita que el servidor de base de datos esté expuesto a ataques de otros agentes o de usuarios no válidos, ya que a dicho servidor sólo llegan peticiones de usuarios válidos y no directamente, sino mediante el mismo sistema de seguridad de agentes implementado, luego entonces, el sistema de seguridad de ABADAM no afecta ni riñe con la seguridad que maneja el servidor de base de datos, sino que le colabora para que permanezca descargado de la tarea de verificar la validez de usuarios, evitándole así el consumo innecesario de recursos y la exposición a ataques.

6. ABADAM Respecto a otros Modelos

Después de realizar una revisión exhaustiva del estado del arte, no se encontraron trabajos que propongan modelos de seguridad para acceder a bases de datos mediante agentes móviles, sin embargo, hay publicaciones respecto a accesos a bases de datos con agentes móviles, pero sin tener en cuenta un modelo de seguridad.

Uno de estos modelos es MAMDAS (Application of Mobile Agents in Mobile Data Access Systems) [11], consistente en el diseño, implementación y prototipo de un motor de búsqueda, el cual soporta accesos globales a recursos mientras preserva la autonomía local usando la tecnología de agentes móviles para mejorar el rendimiento, la escalabilidad y la portabilidad. A pesar que MAMDAS trabaja los accesos a las bases de datos con agentes móviles, no incluye ningún modelo de seguridad como el propuesto en ABADAM.

Otro modelo relacionado es el titulado Acceso a bases de datos distribuidas mediante el uso de agentes móviles [12], este proyecto trabaja con la tecnología de agentes móviles aplicado a sistemas de información geográfica (SIG).

Recientemente, la tecnología de agentes móviles se ha ampliado a otros campos, por ejemplo a la minería de datos: Exploring the Capabilities of Mobile Agents in Distributed Data Mining[13] y en la optimización de consultas: Mobile Agent Cooperation Methods in Hybrid Query Optimization[14]. El resultado de estas investigaciones reafirma la optimización obtenida al usar la tecnología de agentes móviles en los respectivos procesos, lo que hace ABADAM es usar también las bondades de la tecnología de agentes móviles pero para mejorar la seguridad en el proceso de autenticación de los usuarios.

En estudios posteriores, se podría utilizar ABADAM incluso para complementar los estudios aquí mencionados, ya que la seguridad debe hacer parte integral de todo proyecto informático, es allí precisamente donde se encuentra lo novedoso de ABADAM, pues brinda un modelo de seguridad para acceder a bases de datos mediante agentes móviles y de la misma manera se podría estudiar la posibilidad de adaptar este modelo para acceder a otros recursos mediante agentes móviles.

En la quinceava Conferencia Internacional de aprendizaje de Máquinas y Cibernética, se presentó el tema : A Platform of Mobile Agent Based on Distributed Web Database[15], en cuya investigación los autores plantean el problema de seguridad al combinar la tecnología de agentes móviles con las bases de datos distribuidas en la web, pero a diferencia de

ABADAM, los autores no muestran un modelo de seguridad al problema planteado, solo describen el proceso de generación y ejecución de los agentes móviles, también introducen mecanismos de cooperación de los agentes móviles con las consultas y procesamiento de transacciones, donde demuestran una vez mas los mejores resultados en cuanto a rendimiento al usar agentes móviles esos procesos.

La investigación más reciente sobre la seguridad en los sistemas de agentes móviles titulada Trust Enhanced Security for Mobile Agents[20], donde prueban experimentalmente con un prototipo, el aumento en la seguridad al trabajar con la tecnología de agentes móviles, para ello, presentan una arquitectura confiable: MobileTrust (implementado en Java bajo la plataforma Aglets), el cual incorpora un modelo de confianza que capta lo relacionado con la seguridad en un sistema de agentes y proporciona mecanismos seguros de evaluación y actualización que ayudan a la toma de decisiones seguras y precisas para mejorar la seguridad. El problema del servidor malicioso inherente a los agentes móviles, es abordado por MobileTrust asi: Construye un itinerario de confianza, e inicia la ejecución del algoritmo: *recommendation* basado en una evaluación confiable. Para la protección del servidor del agente malicioso, el servidor que se está ejecutando puede tomar decisiones de evaluación de confianza, utilizando la evaluación de relaciones confiables en conjunto con mecanismos de autenticación tradicional, con ello se producen decisiones de autorización dinámicas basadas en la confianza. En términos generales, para mostrar la viabilidad de este modelo, utilizan los siguientes algoritmos: Trust Based Itinerary Composition, Trust Recommendation Algorithm and Trust Update, and Trust Enhanced Authorization [20].

Como se puede observar en otra investigación reciente titulada: "Comparing the Trust and Security Models of Mobile Agents" [21], la mayoría de las plataformas de desarrollo de agentes móviles, minimizan el problema del agente malicioso utilizando firmas digitales y listas de control de acceso. En el modelo de seguridad propuesto en ABADAM se implementan adicionalmente dos niveles más de seguridad, cuya función principal es eliminar los agentes maliciosos que intentan alojarse en el servidor.

7. Conclusiones

En ABADAM los usuarios que desean conectarse a la Base de Datos lo hacen mediante el sistema de agentes móviles que brinda un alto nivel de seguridad contra ataques de otros agentes porque aunque un agente malicioso obtuviese un login y password válidos para conectarse a la base datos, no lo lograría al no tener la credencial que lo amerite como agente válido.

El alto nivel de seguridad se extiende también al no permitir el alojamiento de agentes no deseados o maliciosos en el servidor porque serían destruidos en el mismo instante que lleguen al servidor, esto también evita que el sevidor sea colapsado por la llegada de agentes no deseados. Este aspecto soluciona uno de los problemas por los cuales los agentes móviles no han tenido mucha acogida, como es el problema de seguridad en cuanto al manejo de ataques de agentes maliciosos expuesto en la sección 3.1.1.

Uno de los aportes mas importantes de ABADAM al campo de acción de los agentes móviles, además del alto nivel de seguridad que maneja, es que brinda un mecanismo colaborativo en el proceso de autenticación de usuarios, ya que autentica previamente el agente cliente que porta el login y password como agente válido, aspecto que se refleja de manera positiva en el rendimiento del servidor de base de datos, pues, a dicho servidor sólo llegarán peticiones válidas de usuarios. Esto no sucede en algunos sistemas tradicionales en los cuales muchos usuarios no autorizados establecen contacto con el servidor de base de datos produciendole a éste consumo de recursos y riesgos innecesarios.

ABADAM es una herramienta multiplataforma porque se puede ejecutar sobre cualquier sistema operativo que tenga instalada la máquina virtual de Java y, además, permite al usuario conectarse no solo a bases de datos distribuidas, como se planteó inicialmente en este proyecto, sino que también está diseñado para conectarse a cualquier base de datos mediante un driver ODBC, sea distribuida o no, y ejecutar cualquier instrucción SQL sobre la Base de Datos a la que se logre conectar.

ABADAM es una herramienta multiplata-forma porque se puede ejecutar sobre cualquier sistema operativo que tenga instalada la máquina virtual de Java.

Las aplicaciones donde se puede implementar este modelo no están limitadas al aspecto de seguridad, sino que además,

ABADAM al no tener ocupado el canal de comunicaciones permanentemente mientras se está conectado a la base de datos ofrece un mayor rendimiento en la red institucional, sobre todo en redes donde hay demasiado tráfico, como es el caso de instituciones que además de soportar el tráfico de datos de las aplicaciones empresariales también tienen que soportar el tráfico exagerado que producen los usuarios conectados a la red navegando y descargando archivos de Internet.

Referencias bibliográficas

[1] *Wang Yan y Law, K.C.K.* A mobile agent based system for distributed database access on the Internet. Communication Technology Proceedings, 2000. WCC - ICCT 2000. International Conference on Volume 2, 21-25 Aug. 2000 Page(s):1587 - 1590 vol.2

[2] Haq, M.A. y Matsumoto, M. MAMI: mobile agent based system for mobile internet. Advanced Communication Technology, 2004. The 6th International Conference on Volume 2, 2004 Page(s): 567 – 572.

[3] Danny Lange y Mitsuru Oshima, *Programming and Deploying Java Mobile Agents with Aglets*. Addison Wesley Professional. 1998.

[4] Danny Lange y Mitsuru Oshima, "Mobile Agentes: Enviroments, Technologies and Applications". *Proceedings of the Practical Applications of intelligent Agents and Multi-Agent Technology Conference*, 1998.

[5] Venners Bill, "Solve real problems with aglets, type of mobile agent". *JavaWorld – Under the hood Magazine*, Mayo 1997, pp. 2-4.

[6] M.C. Man y V.K. Wei, "A Taxonomy for Attacks on Mobile Agent", *IEEE EUROCON'2001 Trends in Communications, Vol.2*. 2001, pp. 385-388.

[7] G. Vigna, "Mobile Agents and Security", *Lectura notes in Computer Science*, 1998.

[8] Niklas Borselius, "Mobile agente security", *Electronics & Communication Engineering Journal. Vol. 14*. 2002, pp. 211-218. 2002.

[9] Danny Lange y Mitsuru Oshima, *Programming and Deploying Java Mobile Agents with Aglets*. Addison Wesley Professional. 1998, pp. 176-178.

[10] Scott Oaks, *Java Security*. O'reilly & Associates, Inc. 2001, pp 618.

[11] Yu Jiao, Ali R Hurson. Journal of Database Management. Hershey: Oct-Dec 2004. Vol. 15, Iss. 4; p. 1 (24 pages)

[12] Papastavrou, S.; Samaras, G.; Pitoura, E. Mobile agents for WWW distributed database access. Data Engineering, 1999. Proceedings., 15th International Conference on

23-26 Mar 1999.Page(s): 228-237

[13] U.P. Kulkarni; K.K. Tangod; S. R. Mangalwede; A.R. Yardi. Exploring the Capabilities of Mobile Agents in Distributed Data Mining. Database Engineering and Applications Symposium, 2006. IDEAS '06. 10th International. Dec. 2006. Page(s): 277-280

[14] Win, T.; Khin Mar Lar Tun. Mobile Agent Cooperation Methods in Hybrid Query Optimization. Information and Telecommunication Technologies, 2005. APSITT 2005 Proceedings. 6th Asia-Pacific Symposium on 09-10 Nov. 2005. Page(s): 71- 76

[15] Zhen-Peng Liu; Li-Li Zhu; Jian-Min Xu; Wan-Sheng Tang. A Platform of Mobile Agent Based on Distributed WEB Database. Machine Learning and Cybernetics, 2006 International Conference on Aug. 2006. Page(s): 192-197

[16] Walsh, T.; Paciorek N. y Wong, D. Security and Reliability in Concordia. Mitsubishi Electric ITA, Horizon System Laboratory. USA, 1998.

[17] O´Hare, G. y Jennings, N. Foundations of Distributed Artificial Intelligence. John Wiley & Sons. 1996.

[18] Piccolo, F.L.; Bianchi, G.; Salsano, S. A measurement study of the mobile agent JADE platform. World of Wireless, Mobile and Multimedia Networks, 2006. WoWMoM 2006. International Symposium on a 26-29 June 2006 Page(s):6 pp.

[19] Rana, T.A. Mobile Agents Host Security with Access Control. Computational Intelligence for Modelling, Control and Automation, 2006 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on Nov. 2006 Page(s): 232-232.

[20] Ching Lin; Vijay Varadharajan; Yan Wang; Vineet Pruthi. Trust enhanced security for mobile agents.

E-Commerce Technology, 2005. CEC 2005. Seventh IEEE International Conference on 19-22 July 2005 . Page(s): 231- 238.

[21] Fragkakis, Michail; Alexandris, Nikolaos. Comparing the Trust and Security Models of Mobile Agents. Information Assurance and Security, 2007. IAS 2007. Third International Symposium on 29-31 Aug. 2007 Page(s): 363-368

David Antonio Franco Borré

Ingeniero de Sistemas, Universidad Politécnico Grancolombiano. Magíster en Ciencias Computacionales, Universidad Autónoma de Bucaramanga en convenio con ITESM, Bogotá, Colombia. Docente de la Facultad de Ingeniería, Universidad de Cartagena. Pertenece como investigador al grupo GIMATICA. **dfrancob@unicartagena.edu.co**

Yasmín Moya Villa

Ingeniera de Sistemas, Universidad Tecnológica de Bolívar, de Cartagena, Colombia. Magíster en Ciencias Computacionales, Universidad Autónoma de Bucaramanga en convenio con ITESM, Bogotá, Colombia. Docente de la Facultad de Ingeniería, Universidad de Cartagena y en la Universidad Tecnológica de Bolívar **yasminmoyavilla@yahoo.com**

Moisés Quintana ´Álvarez

Matemático, Universidad Pedagógica de Villaclara, de Santa Clara, Cuba. Magíster en Informática, Universidad Politécnica José A. Echeverría de La Habana, Cuba. Docente en el área de Ingeniería de Sistemas en la Universidad Tecnológica de Bolívar de Cartagena y pertenece como investigador al grupo GRITAS donde realiza estudios sobre Informática Educativa e Inteligencia Artificial. **mquintan@unitecnologica.edu.co**

Miguel Angel García Bolaños

Ingeniero de Sistemas, Universidad Politécnico Grancolombiano. Especialista en Gerencia de Informática, U. Rémington de Medellín, Colombia. Docente en el área de Informática en la Universidad de Cartagena y pertenece como investigador al grupo GIMATICA. **migarbo759@yahoo.com.mx**

Julio Cesar Rodríguez Ribón

Ingeniero de Sistemas, Universidad Industrial de Santander. Especialista en Gestión para el desarrollo empresarial, Universidad Santo Tomás de Aquino. Actualmente cursa Estudios Doctorales en la Universidad Politécnica de Madrid, España. Docente Universidad de Cartagena y pertenece como investigador al grupo GIMATICA donde realiza estudios en el campo del e-learning. **jrodriguezr@unicartagena.edu.co**