Ingeniare. Revista chilena de ingeniería, vol. 16 Nº 2, 2008, pp. 282-294

EXTENDIENDO EL MODELO e-SCARF DE DETECCIÓN DE FRAUDE EN SISTEMAS DE COMERCIO ELECTRÓNICO

EXTENDING THE e-SCARF MODEL FOR FRAUD DETECTION ON ELECTRONIC COMMERCE SYSTEMS

Francisco Arias[1] **Narciso Cerpa**[2]

Recibido 29 de octubre de 2007, aceptado 2 de mayo de 2008
Received: October 29, 2007 Accepted: May 2, 2008

RESUMEN

En este trabajo se extiende un modelo existente de detección de fraude, denominado SCARF, el cual está basado en una técnica de auditoría concurrente que consiste en la inserción de rutinas de auditoría dentro de un programa de aplicación, en este caso un sistema de comercio electrónico. Estas rutinas capturan datos de las transacciones electrónicas y son comparados con reglas que un usuario auditor ha definido previamente para detectar posibles transacciones fraudulentas.
Para extender el modelo se incorporan como requerimientos la evaluación y sugerencias que un conjunto de 15 auditores realizaron a una segunda versión del modelo, denominada e-SCARF, así como también se incluyen mejoras propuestas por los autores de este trabajo. Para validar el modelo extendido, éste se ha implementado para que funcione en conjunto con una plataforma de comercio electrónico de pruebas y un conjunto de usuarios de distintos países han realizado la simulación de compras en dicha tienda.
El producto principal de este trabajo es un modelo extendido, más robusto en sus funcionalidades que sus antecesores, con cambios en la estructura de datos, y nuevos operadores de reglas. Otro producto es el prototipo que lo implementa para una plataforma de comercio electrónico actual.

Palabras clave: Fraude en comercio electrónico, venta en línea, detección de fraude, fraude en Internet, técnicas de auditoría.

ABSTRACT

In this work we extend a fraud detection model, called SCARF, which is based on a concurrent auditing technique that consists of inserting auditing procedures within an application program; in this case, an electronic commerce system. These procedures undertake the capture of electronic transactions data, which is compared with rules that are previously defined by an auditor with the purpose of detecting fraudulent transactions.
To extend this model, we have a set of some requirements: 1) suggestions from a group of 15 auditors that evaluated the second version of this model, called e-SCARF; 2) improvements proposed by the authors of this work. To evaluate the extended model, we have implemented it together with a testing electronic commerce platform. A set of clients from different countries has tested the model by simulating purchases from a store in the electronic commerce platform.
The result of this work is a validated extended model, with more functionalities than its previous versions, changes in the data structure and new operating rules. Another effect is the prototype that implements the model for a current electronic commerce platform.

Keywords: E-commerce fraud, on-line sales, fraud detection, Internet fraud, auditing techniques.

INTRODUCCIÓN

Hoy en día, gracias a la masificación de las redes públicas como Internet, las posibilidades del comercio electrónico son reales para virtualmente cualquier organización o individuo con un computador y una conexión.

Existen muchas ventajas de conducir el comercio electrónico a través de Internet. Por ejemplo, el marketing

[1] Facultad de Ingeniería. Universidad de Talca. Merced 437. Curicó, Chile. E-mail: farias@utalca.cl
[2] Facultad de Ingeniería. Universidad de Talca. Merced 437. Curicó, Chile. E-mail: ncerpa@utalca.cl

se ve beneficiado ya que puede proveer una forma rápida y económica de atraer nuevos clientes, incluso de áreas remotas. Además, las transacciones son realizadas con mayor precisión al prescindir de la reescritura de las órdenes de compra. Por otra parte, las tiendas de comercio electrónico están abiertas 24 horas al día, los siete días de la semana [25].

El comercio electrónico es una tecnología relativamente nueva y, como tal, existen pocas metodologías formalizadas para el desarrollo de tales sistemas. Esto no sólo significa que existen características que estos sistemas adolecen, sino que además la presencia de errores es probablemente alta.

Otra debilidad, y es en la que este trabajo se centra, reside en la dificultad para la detección de fraude. Lo que hace al comercio electrónico vulnerable es que las transacciones son realizadas remotamente y la habilidad de los proveedores para legitimar la identidad de sus clientes es limitada [8]. Además, debido a la necesidad de atraer clientes, la mayoría de los negocios en Internet aceptan el uso de tarjetas de crédito como medio de pago. Los consumidores, sin embargo, se encuentran dubitativos de entregar en línea información de sus tarjetas debido a falencias de seguridad y una alta incidencia de fraude [25], [4]; aunque también ocurre una desconfianza de los proveedores en enviar sus productos sin verificar previamente el pago [1]. Estos antecedentes ponen en evidencia la necesidad de desarrollar sistemas de detección de fraude que se caractericen por ser flexibles y fáciles de mantener.

Hoy en día existen diversos modelos de detección de fraude [2]. Los modelos actuales para detectar fraude involucran el análisis de las transacciones con el fin de realizar un seguimiento al comportamiento de los clientes y obtener patrones de consumo [28]. Estos modelos en general se pueden clasificar en tres enfoques principales [3]:

- Basado en reglas que requieren del conocimiento y la experiencia de un auditor para la elaboración de reglas de detección de fraude.
- Métodos supervisados que consisten en la elaboración de listas positivas y negativas a partir de datos ya clasificados como legítimos e ilegítimos [24], así como también incluyen métodos de minería de datos y puntación de crédito centrados en análisis estadísticos para establecer patrones de detección de fraude [15].
- Detección de anomalías para identificar patrones de datos que no son consistentes con el resto de la información en las transacciones [12].

La implementación de modelos de detección de fraude otorga al auditor un mayor control sobre el proceso de auditoría, mejorando su capacidad de análisis de las transacciones de comercio electrónico [26]. Además, proveyendo los medios de recuperación de datos adecuados, el auditor está en condiciones de realizar auditorías cercanas al óptimo [11].

Desde la perspectiva de un auditor, o experto de sistemas de seguridad para una empresa, identificar el fraude tan pronto como es cometido es de vital importancia. Para cumplir con este objetivo el auditor puede considerar la auditoría continua cuando la mayor parte de la información está en formato electrónico. Por ejemplo, los auditores pueden utilizar software para detectar situaciones de excepción, que ellos mismos han definido, entre todas las transacciones procesadas [14].

Basado en una técnica de auditoría concurrente llamada SCARF (sigla en inglés para *System Control Audit Review File*), se desarrollaron dos versiones de un modelo de detección de fraude para comercio electrónico. La primera versión fue desarrollada por Ng y Wong bajo el nombre de SCARF [23], [30] y la segunda por Loh y fue denominada e-SCARF [18].

La primera versión del modelo, denominada SCARF [23], [30], contaba con las principales funcionalidades de este tipo de herramientas: permitía construir reglas, auditar las transacciones en la plataforma de comercio electrónico IBM Net.Commerce y verificar el cumplimiento de las reglas con los datos recibidos.

Luego, la segunda versión del modelo [18], denominada e-SCARF, agregaba algunas mejoras como los niveles de alerta y reportes más acabados. Pero el aporte de Loh [18] fue realizar un importante estudio empírico donde un conjunto de 15 auditores hicieron una evaluación de e-SCARF, además de sugerir un conjunto considerable de características adicionales para ser incluidas en un modelo posterior.

Con esta información, los objetivos generales de este proyecto son principalmente extender y mejorar el modelo de detección de fraude [23], [30]. Para lograr esto, primero es necesario estructurar el modelo para que se adapte a una plataforma de comercio electrónica actual. Una vez estructurado el modelo, se inicia el proceso de extensión. Proceso que incluye un análisis y estudio de las funcionalidades existentes en el modelo para la evaluación de mejoras, además de evaluar la incorporación de nuevos elementos.

Para el mejoramiento del modelo inicial se ha considerado agregar los siguientes aspectos:

- Unificación de todos los componentes del sistema de auditoría.
- Recuperación de los datos y generación de reportes.
- Procesamiento de los datos de facturación durante el proceso de pago.
- Exportación e importación de las reglas que un auditor ha creado.
- Administración de usuarios que pueden utilizar el sistema de auditoría.
- Sesiones con identificación de usuario auditor, para controlar el fraude corporativo [21].
- Exportación e importación de los registros transaccionales capturados.
- Reportes estadísticos de los datos capturados.
- Búsquedas flexibles mediante referencias cruzadas de los datos.
- Soporte para la definición de valores difusos para la confección de reglas.

Para la implementación de la primera versión del modelo SCARF los autores de aquella investigación [23], [30] comenzaron estudiando y eligiendo entre los métodos y técnicas de auditoría existentes hasta ese entonces. Se evaluaron tres técnicas de auditoría que son: alrededor del computador [7], con el computador [13] y a través del computador [7]. Siendo esta última la más apta para el objetivo de auditar las transacciones de una plataforma de comercio electrónico. Un subconjunto de técnicas de auditoría a través del computador son los denominados métodos de auditoría concurrente, donde detectar el fraude tan pronto como es cometido es la premisa más importante. En general los sistemas basados en estos métodos son clasificados en aquellos que analizan todas las transacciones procesadas y aquellos que analizan sólo transacciones seleccionadas bajo algún criterio [22]. Entre estos métodos, los autores del primer modelo evaluaron los siguientes:

- Facilidad de Prueba Integrada (*Integrated Test Facility*): permite probar un programa de aplicación mediante la inserción de entidades ficticias dentro del sistema y posteriormente datos adicionales son procesados vía estas entidades [20].
- Registro Extendido de Captura (*Snapshot/Extended Record*): esta técnica involucra registrar el estado, en una posición relativa del sistema, y todos los otros datos relacionados a la transacción en diferentes puntos del flujo a través del sistema. Es importante registrar la fecha y la hora del registro para evaluar posibles diferencias [9].
- Archivo de Revisión, Auditoría y Control de Sistemas (*System Control Audit Review File, SCARF*): esta técnica consiste en la inserción de rutinas de auditoría dentro del programa de aplicación. Estas rutinas pueden ser puestas en diferentes puntos de una transacción para monitorear el tráfico de datos [29].

Dada la naturaleza de la aplicación a auditar, SCARF fue el método elegido por los autores [23], [30] para auditar la plataforma de comercio electrónico. Una vez que definieron la forma en que se capturan los datos de las transacciones, establecieron un método basado en reglas para determinar qué transacciones son sospechosas, donde el usuario auditor es quien define la conformación de las reglas.

MODELO DE DETECCIÓN DE FRAUDE

Los autores del modelo inicial [23], [30] desarrollaron un modelo de detección de fraude basado en dos técnicas principales: una variante de Reglas Incrementales (RDR del inglés *Ripple Down Rules*) [10] para facilitar al auditor el diseño de las reglas que sirven para clasificar las transacciones; y SCARF para la adquisición y almacenamiento de los datos capturados.

En la figura 1 se muestra el modelo de detección de fraude compuesto por dos capas: en una se encuentra la plataforma de comercio electrónico y es donde se hallan insertos los procedimientos de auditoría. Estos procedimientos son los responsables de capturar datos transaccionales como: nombre de usuario, clave, número de orden, productos en la orden, precios de los productos, etc. Mientras que en la otra capa se encuentran el servidor y el sistema de detección de fraude.

Variante de reglas incrementales

El modelo inicial [23], [30] incorpora una técnica que sirve al usuario auditor como estrategia para evidenciar transacciones sospechosas de fraude. Dada la naturaleza de las transacciones de comercio electrónico y su composición en variables, implementan un modelo basado en reglas. Una versión modificada de Reglas Incrementales es utilizada para reducir la complejidad de la creación de reglas. La estructura es semejante a un árbol binario invertido que se recorre evaluando las condiciones de cada nodo y, dependiendo del cumplimiento o no cumplimiento de la condición, se sigue la arista correspondiente al resultado de la evaluación. La variación radica en que un nodo del árbol puede representar una decisión o prueba atómica que debe ser evaluada y/o una acción o conclusión a ser tomada.

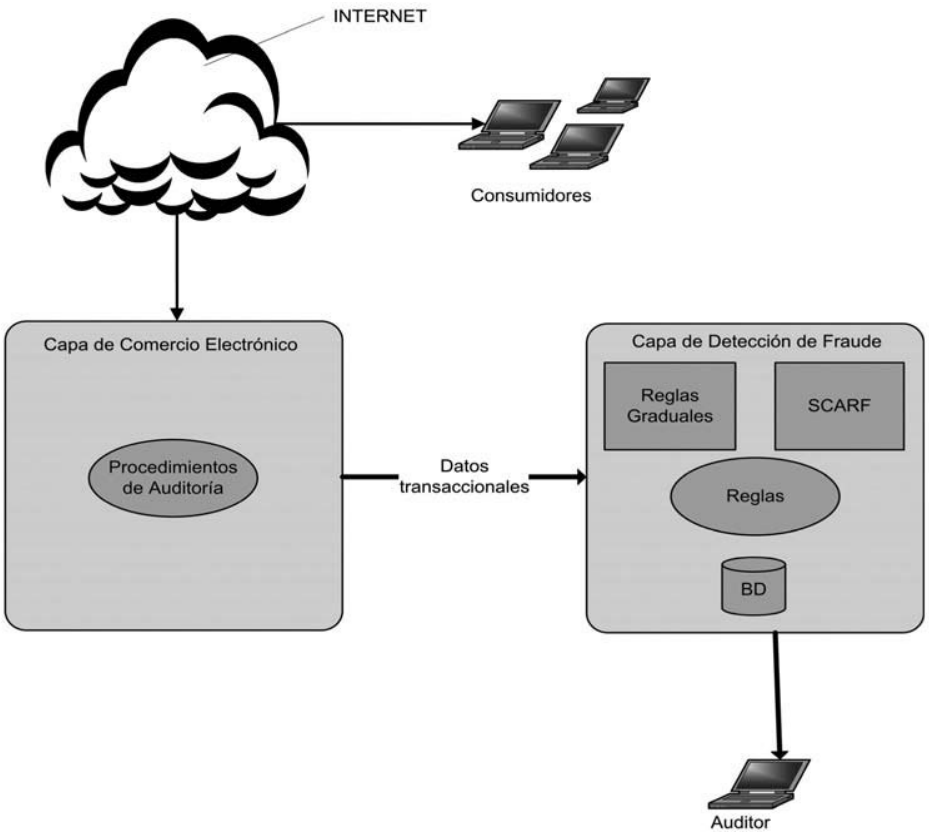

Figura 1. Modelo de detección de fraude.

A partir de esta definición, el proceso de clasificación consiste en comparar los campos de la transacción con cada nodo, seleccionando el nodo siguiente, a partir de la arista verdadera o falsa según dicte la evaluación del nodo actual.

En la figura 2 se muestra un ejemplo de una regla creada utilizando la variante de reglas incrementales.

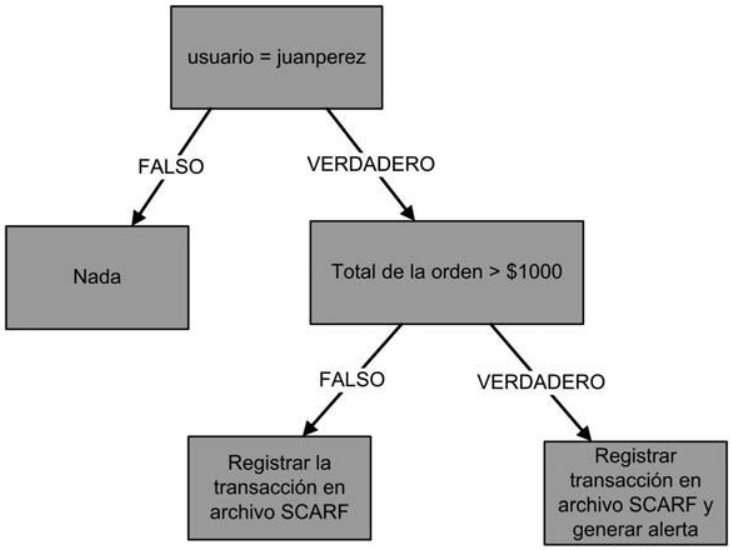

Figura 2. Ejemplo de regla con el método de reglas incrementales modificado.

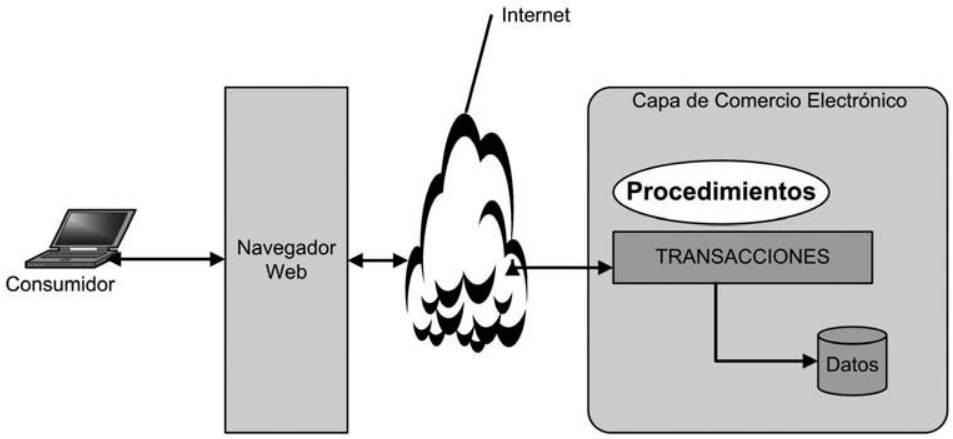

Figura 3. Localización de los procedimientos de auditoría.

Archivo de revisión, auditoría y control de sistemas

Esta es una técnica de auditoría concurrente que utiliza procedimientos de auditoría (en inglés *hooks*) dentro de un programa de aplicación para monitorear continuamente las transacciones [29].

Los procedimientos de auditoría son una técnica de programación utilizada dentro de una cadena de operaciones donde en algún punto específico se requiere de una manipulación especial. Así, después de que el evento de manipulación ocurre, el flujo de control de la cadena sigue su orden original. Esta técnica permite al auditor obtener, por ejemplo, totales de control independiente como resultado del procesamiento normal [13].

Los procedimientos de auditoría son ubicados, como se muestra en la figura 3, de tal forma que sea posible capturar los datos transaccionales necesarios en el momento en que la transacción se produce. De esta forma, la aplicación de detección de fraude puede evaluar los datos tan pronto como éstos son procesados por la plataforma de comercio electrónico. Las plataformas de comercio electrónico usualmente proveen herramientas para extender la lógica del negocio. Haciendo uso de estas herramientas, es posible extender las funcionalidades de los comandos y tareas que controlan toda la lógica del negocio y, en este caso, insertar el código correspondiente para implementar los procedimientos.

Según el modelo de este proyecto, los procedimientos de auditoría no modifican el comportamiento normal de la plataforma de comercio electrónico, es decir, la presencia de los procedimientos es transparente para los usuarios consumidores y administradores.

Esta técnica es utilizada con frecuencia con fines de depuración de código o extensión de las funcionalidades originales de un sistema, pero también es a veces mal utilizada para insertar código (potencialmente malicioso) dentro del evento manipulado.

Los datos transaccionales capturados son almacenados en el archivo SCARF. El análisis de este archivo (y los reportes generados a partir de los datos) es responsabilidad del auditor, quien debe hacer un seguimiento a cualquier irregularidad que pueda aparecer.

Es posible implementar alertas y reportes en tiempo real con la ayuda de SCARF. Esto debería ser implementado para aquellas irregularidades que requieran que algunas acciones sean tomadas tan pronto como sea posible [27].

En SCARF existen tres consideraciones principales: determinar cómo el archivo SCARF será actualizado, definir el formato de los reportes a ser generados y escoger cuándo estos reportes serán generados [18].

Las versiones anteriores del modelo [18], [23], [30] utilizan una variación de SCARF en la forma de cómo los datos son almacenados. La implementación de estos modelos usa una base de datos en lugar de un archivo, como fue concebido inicialmente el concepto de SCARF. Esto es debido a que esta es una solución más efectiva para manejar los datos transaccionales capturados. El sistema administrador de la base de datos es el encargado de preocuparse por la integridad y seguridad de los datos almacenados [18], asimismo puede ofrecer un rápido acceso a ellos y un más eficiente uso de los recursos de espacio en almacenamiento. Para obtener el modelo entidad-relación que soporta los datos del sistema es necesario aplicar el

procedimiento de proyección del modelo orientado a objetos propuesto por Cerpa [5].

Se necesita acceso a la documentación y un alto conocimiento de la plataforma de comercio electrónico para poder determinar cuáles son los puntos donde es necesario insertar los procedimientos de auditoría. Una vez determinados cuáles son los datos que se desea capturar en los registros SCARF, es importante proveer al auditor un acceso expedito a los datos capturados. Los datos en el registro SCARF deben ser correctamente clasificados y presentados al auditor en forma de reportes, para así facilitar la interpretación de los resultados [18].

CARACTERÍSTICAS DEL MODELO EXTENDIDO

En un marco de trabajo para comercio electrónico existen cuatro acciones principales que pueden llevarse a cabo para mejorar la seguridad y confianza entre los actores involucrados en una transacción de comercio electrónico: prevención, detección, investigación y corrección [5]. Cuando un fraude es detectado, este modelo no contempla la funcionalidad necesaria para detener el flujo electrónico de la transacción sospechosa, pero sí provee la información necesaria para que el usuario auditor detenga el envío de productos de aquella transacción. Es decir, este modelo no tiene un carácter correctivo, sino más bien su fortaleza es proveer las funcionalidades necesarias para la detección y la investigación de acciones fraudulentas.

Arquitectura del modelo extendido

El modelo extendido presenta una arquitectura cliente-servidor que se analiza en dos tramos: el primero, es a través de los datos capturados por los procedimientos en la capa de comercio electrónico. Cada uno de los tres procedimientos (Identificación, Orden y Pago) envía los datos capturados en forma de paquetes UDP (del inglés *User Datagram Protocol*) al cliente. Este protocolo orientado a mensajes (datagramas) no requiere que se establezca una conexión previa al inicio de la comunicación, de esta forma sólo se necesita que el cliente del sistema esté conectado con su respectivo servidor de detección de fraudes para estar operativo y recibir los datos capturados por los procedimientos.

Entre el cliente y el servidor de detección de fraude se establece una conexión a través de TCP/IP (del inglés *Transmission Control Protocol/Internet Protocol*); esta conexión es iniciada por el usuario y permite la transmisión de los datos capturados por el cliente en forma ordenada y libre de errores. Una vez que se ha establecido la conexión entre el cliente y el servidor, el cliente procesa los datos recibidos de cualquiera de los tres procedimientos y los envía como un flujo de datos a través de la red.

La arquitectura de este modelo es muy similar a la versión del modelo inicial [23], [30], donde se implementó el sistema en una distribución 2-capas. En la capa de comercio electrónico se encuentra el sistema mismo de comercio electrónico, su base de datos y su servidor. En esta capa además se sitúan los procedimientos de auditoría implementados para la plataforma de comercio electrónico y el cliente del sistema de detección de fraude.

El cliente, si es necesario, buscará en la base de datos de la plataforma de comercio electrónico más información transaccional y la enviará al servidor de detección de fraude para su procesamiento. El servidor de detección de fraude se encuentra en la otra capa de esta configuración.

El auditor tiene acceso al sistema de detección de fraude en la misma capa donde se encuentra el servidor de detección de fraude. En esta capa se le provee de la interfaz necesaria para realizar todas las operaciones de administración del sistema de detección de fraude.

En la figura 4 se ilustra la arquitectura del modelo, donde el servidor de detección de fraude aparece bautizado como e-Llaitun[3].

Modelo de los procedimientos de auditoría en la capa de comercio electrónico

Este modelo incluye tres procedimientos en la plataforma de comercio electrónico, en distintas secciones del proceso de una transacción. El modelo inicial incorporó dos procedimientos: uno para el proceso de identificación de usuarios y otro para el procesamiento de las órdenes. El modelo extendido considera un tercer procedimiento para el proceso de pago de las órdenes. Los procedimientos capturan algunos datos en el acto mismo de la transacción y, cuando es necesario, el cliente consulta a la base de datos de la plataforma de comercio electrónico para complementar con más datos relacionados con la transacción.

En la tabla 1 se listan todos los datos que son capturados por los tres procedimientos en el momento en que ocurre una transacción.

[3] Proviene del Mapudungún *llaitún*, que significa vigilar o fijarse bien en algo.

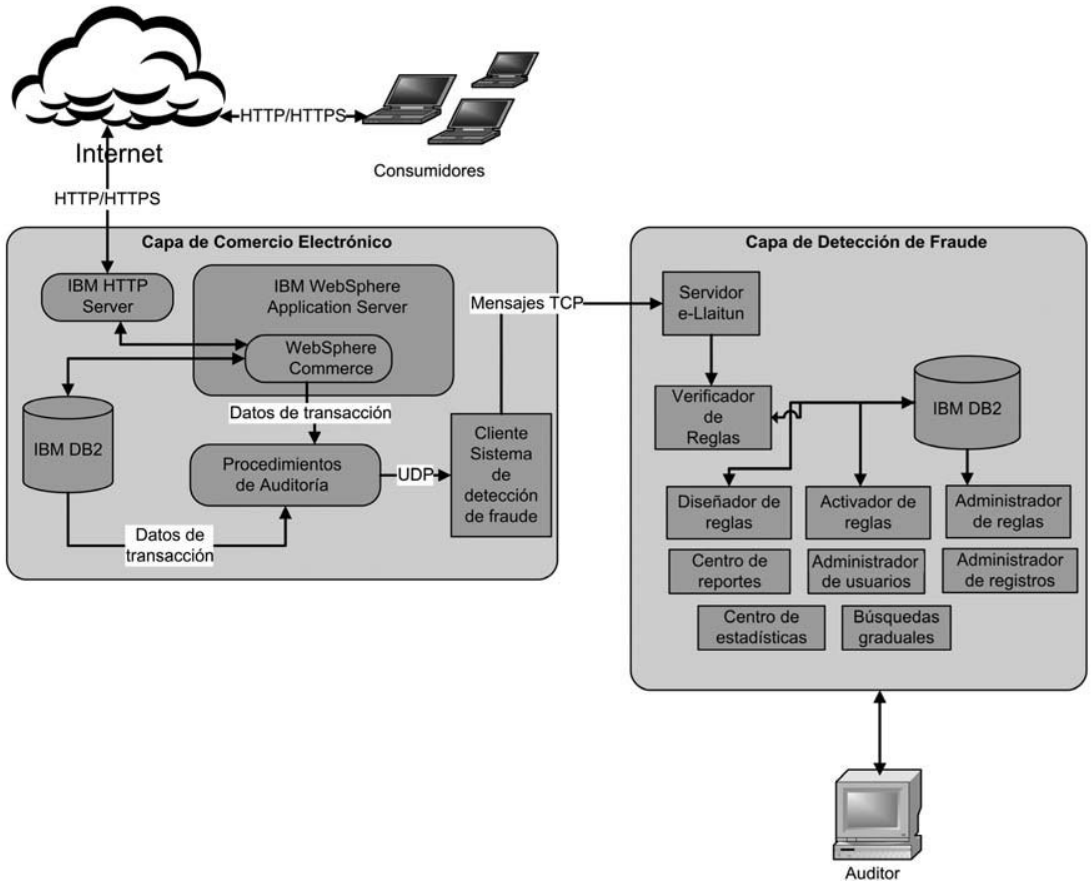

Figura 4. Arquitectura del modelo extendido.

Tabla 1. Datos capturados durante la transacción.

Procedimiento	Dato	Descripción
Identificación	Identificador de usuario	Cadena que contiene el nombre de usuario.
Identificación	Clave	Cadena que contiene la clave del usuario.
Identificación	Identificador de tienda	El identificador de la tienda a la cual pertenece el usuario.
Identificación	Resultado de identificación	Éxito o fracaso en el inicio de sesión.
Orden	Identificador de orden	Identificador de la orden.
Pago	Identificador de usuario	Cadena que contiene el nombre de usuario.
Pago	Número de tarjeta de crédito	Número de tarjeta de crédito utilizada en la orden.
Pago	Identificador de tienda	El identificador de la tienda a la cual pertenece el usuario.
Pago	Resultado del pago	Éxito o fracaso en el proceso de pago.

Modelo del cliente en la capa de comercio electrónico

El cliente, que está situado en la capa de comercio electrónico, tiene la misión de ordenar los datos recibidos desde los procedimientos y enviarlos en un flujo de datos al servidor de detección de fraude, el que puede estar en la misma capa de comercio electrónico o en una capa distinta como se presenta en su arquitectura original. Cuando son requeridos datos adicionales, el cliente consulta a la base de datos de la plataforma de comercio electrónico por ellos, los ordena y los envía al servidor. En la tabla 2 se listan todos los datos que envía el cliente, finalmente después de consultar la base de datos. La tabla 2 además muestra los datos que son capturados por el modelo inicial [23], [30] y los que se han agregado en el modelo extendido. En esta versión del modelo se han agregado nuevos datos que no se capturaban en las versiones anteriores de éste: la dirección IP de consumidor que se identifica en el portal de comercio electrónico, la fecha y hora en que la orden fue ingresada al sistema, la dirección de facturación para la orden, la dirección de envío de cada producto que se

incluye en la orden y la fecha y hora en que cada producto fue puesto en la orden. Además, esta versión incluye un nuevo procedimiento para el proceso de pago. Éste permite, entre otras cosas, llevar un registro de la cantidad de tarjetas de crédito utilizadas por un consumidor.

Tabla 2. Datos enviados por el cliente.

Procedimiento	Datos
Identificación	**Modelo inicial:** usuario, clave, identificador de tienda, resultado de identificación. **Modelo extendido:** dirección IP.
Orden	**Modelo inicial:** identificador de orden, identificador de tienda, tipo de pago, número de tarjeta de crédito, resultado de la orden, usuario, identificador de productoX[4], cantidad de productoX, precio de productoX. **Modelo extendido:** hora y fecha de ingreso de orden, dirección de facturación1, dirección de facturación 2, dirección de facturación 3, ciudad de facturación, estado de facturación, país de facturación, dirección1 de envío de productoX, dirección2 de envío de productoX, dirección3 de envío de productoX, ciudad de envío de productoX, estado de envío de productoX, país de envío de productoX, fecha y hora de ingreso del productoX a la orden.
Pago	**Modelo extendido:** usuario, número de tarjeta de crédito, identificador de tienda, resultado de la validación de la tarjeta de crédito.

Características de la capa de detección de fraude

La capa de detección de fraude está compuesta por varios módulos de administración, que en su conjunto proveen las principales funciones que componen el núcleo de este modelo. A continuación se presenta la descripción de los módulos del modelo inicial [23], [30] que se mantienen en el modelo extendido:

- **Diseñador de reglas:** permite la creación, modificación y eliminación de reglas para la detección de fraude. Este módulo permite al auditor diseñar reglas de manera gráfica, como se muestra en la figura 5.

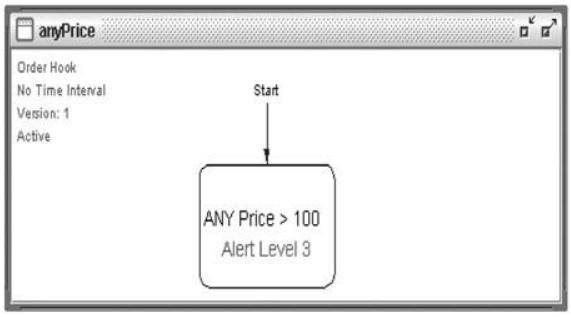

Figura 5. Interfaz para el diseño de reglas.

- **Activador de reglas:** ofrece la funcionalidad de definir qué reglas se encuentran activas para auditar los datos capturados.
- **Servidor:** permite iniciar y detener el servidor de detección de fraude.

El modelo extendido propuesto en este trabajo agrega además los siguientes módulos:

- **Administrador de reglas:** a través de este módulo es posible exportar e importar reglas en dos formatos de archivo.
- **Centro de reportes:** permite al usuario generar reportes para intervalos de tiempo definidos y realizar referencias cruzadas con los datos. Desde aquí, también es posible ver reportes de las actividades realizadas por los mismos usuarios de la aplicación de detección de fraude. Existe además la opción de imprimir los reportes generados.
- **Administrador de usuarios:** ofrece las herramientas necesarias para la gestión de usuarios y sesiones.
- **Administrador de registros:** permite importar y exportar los registros capturados en base a criterios definidos por el usuario.
- **Centro de estadísticas:** este módulo permite al usuario generar reportes estadísticos de variables, reglas o distribución de los datos. En el caso de las variables se obtiene el mínimo, el máximo, el promedio y la desviación estándar de los datos capturados para esa variable. Para las reglas se incluye un reporte con la frecuencia de ocurrencia para cada regla. Y para la distribución, el auditor define el número de intervalos para una variable seleccionada y el reporte muestra cuántos valores de esa variable pertenecen a cada intervalo.
- **Búsquedas flexibles:** a través de este módulo es posible realizar búsquedas flexibles en los datos capturados y ver los resultados en forma de reportes. Para ello, el auditor dispone de un compositor de consultas que lo ayudará a elaborar las consultas que desee realizar.
- **Valores difusos:** este módulo permite calcular o definir valores difusos para las variables numéricas del sistema.

[4] Se reemplaza X por un número que identifica a cada producto en la orden.

Esta función es de utilidad para el auditor al momento de diseñar reglas en las cuales desea comparar valores difusos. Por ejemplo, si una orden incluye un producto cuyo valor es mayor al máximo de todos los productos antes registrados por el sistema.

VALIDACIÓN DEL MODELO EXTENDIDO

Para validar el modelo extendido se ha implementado un prototipo y ha sido puesto en marcha en conjunto con un sitio de comercio electrónico para pruebas que ofrece todas las funcionalidades de una tienda de comercio electrónico estándar: cuentas, carro de compras, despachos, etc. Se pidió a algunas personas que se registraran en la tienda y realizaran algunas compras. Las reglas diseñadas para probar el sistema se describen en la tabla 3.

Tabla 3. Descripción de las reglas de pruebas.

Procedimiento	Regla	Descripción
Identificación	"IP ∈ ES", "IP ∈ SE", "IP ∈ AR" Ventana de Tiempo: No. Nivel de Alerta: 3.	Captura los datos de los usuarios cuya dirección IP pertenece a España, Suecia o Argentina.
Identificación	"Contador Fallos de Identificación = 3" Ventana de Tiempo: 1 Hora. Nivel de Alerta: 3.	Captura los datos de los usuarios que hayan fallado en su identificación en tres oportunidades dentro de un período de tiempo de una hora.
Orden	"Cualquier Precio > 100" Ventana de Tiempo: No. Nivel de Alerta: 3.	Captura los datos de una orden que incluya al menos un producto cuyo precio sea mayor que 100 (dólares en la tienda de ejemplo).
Pago	"Contador de Fallos del Número de Tarjeta de Crédito = 1" Ventana de Tiempo: No. Nivel de Alerta: 3.	Captura la información de aquellas transacciones que hayan sido rechazadas por la validación del número de tarjeta de crédito en una oportunidad.
Pago	"Contador de Tarjetas >= 3" Ventana de Tiempo: No. Nivel de Alerta: 3.	Captura los datos transaccionales de aquellos usuarios que han utilizado al menos 3 tarjetas de crédito distintas.

Los datos capturados por el procedimiento de identificación y enviados por el cliente se muestran en la tabla 4. Todos estos datos son capturados en el preciso instante en que el usuario intenta iniciar sesión en la tienda de comercio electrónico. Se observa que el usuario "alex" equivoca tres veces su clave poniendo a prueba la regla "Contador de Fallos de Identificación = 3" (resultado de identificación con valor "0" en la tabla 4).

Para los resultados de la tabla 4, la aplicación de detección de fraude ha generado alertas para el usuario "ricardo" ya que la dirección IP del computador desde el cual inició sesión pertenece a España y esto ha accionado la regla "IP ∈ España".

En la tabla 5 se presenta una muestra de los datos capturados por el procedimiento de orden.

Los tres resultados presentados en la tabla 5 han generado alertas en el sistema de detección de fraude, al tener al menos un producto dentro de sus órdenes cuyo valor monetario es mayor que 100, accionando así la regla "Cualquier Precio > 100".

A continuación, en la tabla 6, se presenta una muestra de los datos capturados por el procedimiento de pago con tarjeta de crédito.

El usuario "ricardo" ha introducido cuatro números de tarjetas de crédito distintos. Por tanto ha accionado la regla "Contador de Tarjetas >= 3" en dos oportunidades. Mientras que el número de la tarjeta de crédito ingresada por el usuario "alex" ha sido rechazado por el sitio de comercio electrónico accionando la regla "Contador de Fallos de la Tarjeta de Crédito = 1".

Los resultados obtenidos en estas pruebas han sido satisfactorios, sin embargo, es necesario realizar pruebas en un ambiente real de negocios. Empezando por el hecho de que las reglas deben ser diseñadas por un usuario auditor o experto de seguridad de una empresa; para medir así el grado de dificultad que puede presentarse a un usuario del sistema. Lo mismo aplica para todas las demás funciones. Por otra parte, pese a que las personas que visitaron la tienda desconocían el funcionamiento de la aplicación de detección de fraude, recibieron instrucciones generales y acotadas de cómo realizar las compras en la tienda.

Por lo tanto, la evaluación realizada dentro de este entorno de pruebas se considera válida, teniendo en cuenta que dicho entorno además se utilizó para evaluar aspectos técnicos del software tales como tiempos de respuestas y corrección de errores de implementación.

Desde el punto de vista de diseño, se mejoró notablemente la interfaz con respecto a las versiones anteriores del prototipo, ofreciendo ahora una interfaz amigable con todas sus funcionalidades unificadas en ella. El sistema e-Llaitun, como producto final de este trabajo, es robusto en sus funcionalidades, eficiente en la captura de datos transaccionales y está unos pasos más cerca de profesionalizarse.

Tabla 4. Datos capturados por el procedimiento de identificación.

Tipo Procedimiento	**Usuario**	**Clave**	**Identificador de tienda**	**Resultado de identificación**	**IP**	**País**
0	aaguiluz	aaguiluz007	10001	1	200.112.239.11	Chile
0	mpineda	mpineda123	10001	1	200.27.241.178	Chile
0	ricardo	ricardo001	10001	1	88.8.118.113	España
0	cdiaz	cdiaz4talca	10001	1	164.77.100.78	Chile
0	alex	xtreme3	10001	0	200.50.58.87	Chile
0	alex	xtreme5	10001	0	200.50.58.87	Chile
0	alex	xtreme9	10001	0	200.50.58.87	Chile
0	alex	xtreme7	10001	1	200.50.58.87	Chile
0	Lute	ellute1	10001	1	88.3.110.28	España
0	ElDioni	Eldioni1	10001	1	138.100.49.89	España
0	Dahlborn	osito61	10001	1	130.243.24.210	Suecia
0	kanguro	kanguro1	10001	1	138.100.49.162	España

Tabla 5. Muestra de datos capturados por el procedimiento de orden.

Tipo Procedimiento	1	1	1
Número de orden	11506	11507	11503
Identificador de tienda	10001	10001	10001
Tipo de pago	Mastercard	Mastercard	Mastercard
Número de tarjeta de crédito[5]	510121535304XXXX	558933833401XXXX	544983084171XXXX
Resultado de la orden	1	1	1
Usuario	ricardo	cdiaz	mpineda
Fecha y hora de ingreso de orden	2007-01-31 16:27:58.297000	2007-01-31 16:50:18.734000	2007-01-31 16:57:20.187000
Dirección de facturación 1	Lancia #6	Dos Norte 1320	Peña 850
Dirección de facturación 2	Restaurant Honoris Causa	Centro	Oriente
Dirección de facturación 3	null	Null	null
Ciudad de facturación	León	Talca	Curicó
Estado de facturación	León	Maule	Curicó
País de facturación	España	Chile	Chile
Producto1	10001	10010	10065
Cantidad1	1	2	1
Precio1	44999	159999	64999
Dirección de envío 1 producto 1	Lancia no 6	Uno Sur 120	Membrillar 0338
Dirección de envío 2 producto 1	Restaurant Honoris Causa	Villa Galilea	Conavicoop
Dirección de envío 3 de producto 1	null	Null	Null
Ciudad de envío producto 1	León	Talca	Curicó
Estado de envío producto 1	León	Maule	Curicó
País de envío producto 1	España	Chile	Chile
Fecha y hora de ingreso de producto 1	2007-01-31 16:20:24.718000	2007-01-31 16:47:16.906000	2007-01-31 11:24:24.562000

[5] Los cuatro últimos dígitos han sido omitidos por razones de seguridad.

Tabla 6. Muestra de los datos capturados por el procedimiento de pago.

Tipo Procedimiento	Usuario	Número de tarjeta de crédito	Identificador de tienda	Resultado del pago
2	ricardo	510121535304XXXX	10001	1
2	ricardo	538841457275XXXX	10001	1
2	ricardo	536429326917XXXX	10001	1
2	ricardo	542062878308XXXX	10001	1
2	alex	550003751044XXXX	10001	0

CONCLUSIONES

El producto final de este proyecto es un nuevo modelo extendido y un prototipo que lo implementa. El modelo extendido posee una serie de mejoras respecto a sus predecesores. Los siguientes son los aspectos más relevantes que fueron incorporados:

- Unificación de todos los módulos de la capa de detección de fraude.
- Nuevo procedimiento de auditoría para el proceso de pago.
- Sesiones con identificación de usuario auditor, para controlar el fraude corporativo [Michaud06].
- Exportación e importación de reglas.
- Reportes actualizados en tiempo real.
- Gestión de usuarios del sistema de detección de fraude.
- Exportación e importación de registros.
- Reportes estadísticos para datos capturados.
- Búsquedas flexibles mediante referencias cruzadas en los datos.
- Cálculo de valores difusos y personalizados.

Y estos son los nuevos datos que se han agregado a los capturados inicialmente:

- Dirección IP del consumidor.
- Cantidad de tarjetas de crédito [19] que registra un consumidor.
- Dirección de facturación de una orden.
- Dirección de despacho de una orden.
- Fecha y hora de ingreso de la orden al sistema.
- Fecha y hora de ingreso de cada ítem a una orden.

Este modelo presenta en su implementación el uso de técnicas que pueden ser reemplazadas por otras más recientes o aplicadas desde otro enfoque. Como es el caso de la variante de Reglas Incrementales para la composición de reglas, que puede ser reemplazada por la implementación de una técnica que, en base a los datos capturados por las mismas reglas, sea capaz de ajustar las reglas o de sugerir nuevas reglas, por dar un ejemplo.

Para los profesionales en comercio electrónico, este modelo actualmente está en condiciones de ser implementado sin dificultad sobre arquitecturas que utilicen la plataforma de comercio electrónico de IBM, así como también puede servir de guía para implementaciones en otras plataformas, realizando los ajustes necesarios para el proceso de captura de datos. Por ejemplo, la plataforma Microsoft Commerce Server posee el "Commerce Pipeline Editor", donde los desarrolladores podrán realizar la implementación de los procedimientos con un mínimo de dificultad.

En una empresa, la implementación de este modelo se traduce en una herramienta que puede proveer una línea de defensa para la detección oportuna de fraude interno o externo en una plataforma de comercio electrónico. Además provee las funcionalidades necesarias para ayudar a los auditores de comercio electrónico a realizar completos análisis de las transacciones capturadas, en búsqueda de posibles fraudes perpetrados en el tiempo.

El comercio electrónico es una actividad creciente en los países de Latinoamérica, pero ya instalada en los países desarrollados, por lo que la demanda por software de detección de fraude va en aumento. La velocidad con que los negocios son realizados hoy en día demanda el uso de herramientas de detección, prevención y/o corrección de fraude en tiempo real. Para un auditor o experto en seguridad, descubrir el fraude tan pronto como es cometido es de vital importancia, además de recuperar la información de manera eficiente, que ayude a mejorar su entendimiento de los patrones de fraude [16].

Alternativas de implementación del modelo

El modelo extendido tiene un potencial interesante para ser mejorado en el futuro. A continuación se enuncian algunas posibles incorporaciones y/o evoluciones:

- **Arquitectura basada en controladores:** en el mercado de software actual existen variadas plataformas de comercio electrónico. Este software está diseñado para funcionar sólo con la arquitectura del software de IBM, WebSphere Commerce. Una mejora notable sería implementar una arquitectura genérica que sea basada en controladores (en inglés *drivers*). De esta forma, sólo sería necesario programar un controlador para cada plataforma de comercio electrónico y el sistema debería seguir funcionando de forma transparente.

- **Incorporación de métodos correctivos:** actualmente el software no posee un enfoque correctivo. Es decir, cada vez que una regla es accionada el sistema cumple con iniciar las alertas seleccionadas por el usuario auditor y no provee las herramientas necesarias para intervenir la transacción, ya que normalmente una regla es accionada cuando una transacción es considerada sospechosa. Pero, por ejemplo, podría suceder que, en una nueva versión, se mantenga un registro de los números de tarjetas de crédito que ya tienen un prontuario de fraude demostrado y sea necesario detener aquellas transacciones que contengan dichos números.
- **Minería de datos:** hoy en día, los datos transaccionales acumulados en una empresa se han convertido en un activo más de ella. Así, si un usuario auditor puede identificar aquellas transacciones que fueron fraudulentas, es posible que a partir de estos datos almacenados se obtengan patrones de fraude electrónico. Esto puede ser posible con la implementación de un módulo que permita realizar minería de datos con la información capturada. De esta forma, es posible también que el mismo sistema genere nuevas reglas, basado en el conocimiento acumulado.
- **Detección basada en lógica inductiva:** una de las técnicas que está obteniendo popularidad consiste en la aplicación de programación de lógica inductiva (ILP sigla en inglés para *Inductive Logic Programming*) para la detección de fraude. Este método consiste en detectar fraude y anomalías por medio de métodos inteligentes artificialmente, que, a diferencia de otros métodos de detección de patrones de fraude, esta técnica no sólo permitiría obtener las reglas, sino además mejorarlas continuamente.
- **Procedimientos personalizados:** una mejora interesante puede ser la incorporación de una interfaz gráfica que permita al usuario auditor definir todo acerca de un procedimiento de auditoría. El usuario podría especificar gráficamente dónde requiere insertar el procedimiento y qué datos le interesa capturar. Esto puede ser logrado con el uso de tecnologías como AspectJ, que permitiría realizar este tipo de acciones de una forma transparente y sistematizada.
- **Distribución en 3-capas**: actualmente el sistema es distribuido en una arquitectura de dos capas. En la capa de comercio electrónico se encuentra el cliente de este software y en la otra capa está el servidor y la interfaz gráfica que permite al usuario auditor realizar las operaciones administrativas de detección de fraude. Una posible mejora podría ser separar la interfaz de administración del servidor de detección de fraude, para que este último pueda funcionar como un servicio del sistema operativo y sea totalmente independiente de la ejecución de las labores administrativas.
- **Sistemas inmunes artificiales:** los métodos de detección de fraude aplicados en este modelo dependen directamente de la experiencia del usuario auditor. Una extensión interesante es la incorporación de un sistema inmune artificial que utilice algoritmos de selección negativa [17] para la detección de fraude y anomalías en el sistema.

Trabajo futuro

El proyecto a futuro contempla la realización de una completa validación del modelo de detección de fraude en un ambiente de negocios real (que involucre empresas, bancos, instituciones financieras, clientes, etc.) a partir de la implementación aquí presentada. De esta forma será posible evidenciar la eficiencia del modelo para su continuo perfeccionamiento y posiblemente evaluar la incorporación de métodos, técnicas y/o implementaciones alternativas, como las previamente mencionadas, que contribuyan al mejoramiento del proceso de detección de fraude.

REFERENCIAS

[1] B. Anderson, J. Hansen, P. Lowry and S. Summers. "Model checking for e-commerce control and assurance". IEEE Transactions on Systems. Man and Cybernetics. Vol. 35 Nº 3. 2005.

[2] T. P. Bhatla, V. Prabhu and A. Dua. "Understanding credit card frauds". Cards Business Review 1. Tata Consultancy Services. June 2003.

[3] R. Bolton and D. Hand. "Statistical Fraud Detection: A Review". Statistical Science, Vol. 17 Nº 3, pp. 235-255. August 2002.

[4] P. Burns and A. Stanley. "Fraud management in the credit card industry". SSRN eLibrary. 2002.

[5] N. Cerpa and R. Jamieson. "A security, trust and assurance research framework for electronic commerce". IFIP TC8 Working Conference on E-Commerce\E-Business. September 2001.

[6] N. Cerpa. "Mapping object-oriented model into a relational model". Encyclopedia of Library and Information Science, pp. 1770-1777. 2003.

[7] M. V. Cerullo and M. J. Cerullo. "Impact of sas no. 94 on computer audit techniques". Information Systems Control Journal. Vol. 1. 2003.

[8] R. Clarke. "Promises and Threats in Electronic Commerce". 1997. Fecha de consulta: 30 de enero de 2007. URLs: http://www.anu.edu.au/people/Roger.Clarke/EC/Quantum.html

[9] G.B. Davis, D.L. Adamas and C.A. Schaller. "Auditing and EDP". 2nd edition. American Institute of Certified Public Accountants, pp. 253-265. 1983.

[10] G.A. Delzoppo, M. Mulholland and D.B. Hibbert. "A Novel Application of Ripple Down Rules to Selecting a Method of Chemical Analysis for a Variety of Chemicals and their Sample Matrices", pp. 2-6. 1993.

[11] G. Dionne, F. Giuliano and P. Picard. "Optimal auditing for insurance fraud". SSRN eLibrary. 2003.

[12] Z. Ferdousi and A. Maeda. "Anomaly Detection Using Unsupervised Profiling Method in Time Series Data". ADBIS Research Communications. 2006.

[13] G. Gay and R. Simmett. "Auditing and Assurance Services in Australia". McGraw-Hill. 2000.

[14] G. Helms and J. Mancino. "The electronic auditor". Journal of Accountancy. Vol. 185 Nº 4, pp. 45-48. April 1998.

[15] C.L. Huang, M.C. Chen and C.J. Wang. "Credit scoring with a data mining approach based on support vector machines". Expert Systems with Applications. August 2007.

[16] K. Jamal, S. Grazioli and P. Johnson. "A cognitive approach to fraud detection". SSRN eLibrary. 2006.

[17] J. Kim, A. Ong and R. Overill. "Design of an artificial immune system as a novel anomaly detector for combating financial fraud in the retail sector". The 2003 Congress on Evolutionary Computation. 2003.

[18] S. Loh. "Using continuous assurance to detect fraud in e-commerce transactions". Thesis (hons). The University of New South Wales. School of Information Systems. Technology and Management. Sydney. 2002.

[19] B. Macklin. "E-commerce at what price?". Privacy protection in the information economy. SSRN eLibrary. 1999.

[20] W.C. Mair, D.R.M. Wood and K.W. Davis. "Computer Control and Audit". 2nd edition. The Institute of Internal Auditors, pp. 143-146, 419-420. 1978.

[21] D. Michaud, C. Dutton and K. Magaram. "Empowering board audit committees: Electronic discovery to facilitate corporate fraud detection". SSRN eLibrary. 2006.

[22] L.C. Mohrweis. "Usage of concurrent EDP audit tools". The EDP Auditor Journal. Vol. 3, pp. 49-54. 1988.

[23] B. Ng and K. Wong. "An audit review system for electronic commerce". Thesis (hons). The University of New South Wales. Schools of Electrical Engineering and Computer Science and Engineering, Sydney. 1999.

[24] C. Phua, V. Lee, K. Smith-Miles and R. Gayler. "A Comprehensive Survey of Data Mining-based Fraud Detection Research". Clayton School of Information Technology. Monash University. 2005.

[25] M. Plonien. "Electronic commerce on the internet". The CPA Journal. Vol. 68 Nº 5, pp. 82-84. May 1998.

[26] A. Reinstein and M. Bayou. "A comprehensive structure to help analyze, detect and prevent fraud". SSRN eLibrary. 1999.

[27] R.S. Sriram and G.E. Sumners. "Understanding Concurrent Auditing Techniques". EDPACS, pp. 1-8. 1992.

[28] Y.H. Tan and W. Thoen. "An Outline of a trust model for electronic commerce". Applied Artificial Intelligence. Vol. 14, pp. 849-862. 2000.

[29] R. Weber. "EDP Auditing: Conceptual Foundation and Practice". 2nd edition. McGraw Hill, pp. 751-785. 1998.

[30] K. Wong, B. Ng, N. Cerpa and R. Jamieson. "An Online Audit Review System for Electronic Commerce". Proceedings of the 13th Bled Electronic Commerce Conference 2000. Bled. Slovenia. June 20-23. 2000.