UNIVERSIDAD SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS

SEGURIDAD EN TRANSACCIONES COMERCIALES CON TARJETAS DE CRÉDITO

David Haroldo Herrera López

Asesorado por Ing. Carlos Roberto Iraheta Galicia

Guatemala, agosto de 2005

UNIVERSIDAD SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

SEGURIDAD EN TRANSACCIONES COMERCIALES CON TARJETAS DE CRÉDITO

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

DAVID HAROLDO HERRERA LÓPEZ
ASESORADO POR: ING. CARLOS ROBERTO IRAHETA GALICIA

AL CONFERÍRSELE EL TÍTULO DE
INGENIERO EN CIENCIAS Y SISTEMAS

Guatemala, agosto de 2005

UNIVERSIDAD SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	---
VOCAL II	Lic. Amahán Sánches Álvarez
VOCAL III	Ing. Julio David Galicia Celada
VOCAL IV	Br. Kenneth Issur Estrada Ruiz
VOCAL V	Br. Elisa Yazminda Vides Leiva
SECRETARIO	Ing. Marcia Ivonne Véliz Vargas

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Sydney Alexander Samuels Milson
EXAMINADOR	Ing. Edgar Estuardo Santos Sutuj
EXAMINADOR	Ing. Luís Alberto Vettorazzi España
EXAMINADOR	Ing. Ligia Maria Pimentel Castañeda
SECRETARIO	Ing. Pedro Antonio Aguilar Polanco

HONORABLE TRIBUNAL EXAMINADOR

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

SEGURIDAD EN TRANSACCIONES COMERCIALES CON TARJETAS DE CRÉDITO,

tema que me fuera asignado por la Coordinación de la Carrera de Ciencias y Sistemas en febrero de 2004.

David Haroldo Herrera López

Agradecimientos

A:

Dios:

por permitirme cumplir con esta meta e iluminar mi camino día a día.

mis padres:

por su apoyo, amor y orientación.

mi esposa:

por brindarme su amor incondicional y su apoyo para alcanzar esta meta.

mis suegros:

por brindarme su apoyo y confianza.

mi asesor:

Ingeniero Carlos Roberto Iraheta Galicia, por tomarse el tiempo para revisar mi trabajo de graduación y orientarme con las correcciones pertinentes durante el desarrollo del mismo.

mis centros de estudios:

desde mi educación primaria a diversificado, Escuela Nacional Estado Unidos, Instituto Dr. José Matos Pacheco, Instituto Científico Comercial en Computación, los cuales me permitieron prepárame para llegar a la Universidad San Carlos de Guatemala y cumplir esta meta.

ACTO QUE DEDICO A:

mi hija Keyla Nicte Herrera Córdova, por ser la razón que me hace mejor cada día.

mi esposa Maria Virginia Córdova Pérez por su amor, por creer siempre en mí y ser mi apoyo en los momentos más difíciles.

mis padres: Aura Elizabeth López Calderón y William Haroldo Herrera Díaz, por su amor y apoyo a lo largo de mi vida.

mis hermanos: Wilfredo Herrera López y Denis Edu Herrera López, por apoyarme y brindarme su cariño.

mis abuelos: Herminia Díaz de Herrera, Eligia Calderón (q.e.p.d), Luís Herrera (q.e.p.d), y Gilberto López (q.e.p.d) por su cariño.

mis tíos y tías por su confianza y cariño.

mis suegros: Herlinda Pérez de Córdova y José Arturo Córdova Ramírez, por haber creído en mí y haberme brindado su apoyo incondicional.

mí cuñados: Carla Córdova de Sáenz y Juan Carlos Sáenz, por brindarme su cariño y apoyo.

mis sobrinas y sobrino: Virginia, Melany, Nicole, Haydee, Katherine, Emily y José Carlos.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES IX

GLOSARIO XI

RESUMEN XIX

OBJETIVOS XXIII

INTRODUCCIÓN XXV

1. COMERCIO ELECTRÓNICO Y SU SEGURIDAD 1

1.1 Comercio electrónico 1

1.1.1 Historia 1

1.1.2 Definición 2

1.1.3 Categorías 3

1.1.4 Agentes que intervienen 4

1.1.5 Proceso 5

1.2 Seguridad en el comercio electrónico 6

1.2.1 *SSL (Secure Socket Layer)* 6

1.2.2 *SET (Secure Electronic Transaction)* 7

1.2.3 Encriptamiento 7

1.2.4 Certificados digitales 8

1.2.5 Firma digital 8

1.2.6 *PGP (Pretty Good Privacy)* 8

2. AMENAZAS Y ATAQUES AL COMERCIO ELECTRÓNICO 9

2.1 Amenaza 9

2.1.1 Tipos de amenazas 10

2.1.1.1 Interrupción 10

2.1.1.2 Interceptación 11

2.1.1.3 Modificación 13

2.1.1.4 Fabricación 14

2.2 Ataque 15

2.2.1 Ataques pasivos 15

2.2.1.1 Obtención del origen y destino 16

2.2.1.2 Control del volumen de tráfico 17

2.2.1.3 Control de las horas habituales 17

2.2.2 Ataques activos 18

2.2.2.1 Suplantación de identidad 18

2.2.2.2 Reactuación 19

2.2.2.3 Modificación de mensajes 20

2.2.2.4 Degradación fraudulenta del servicio 20

2.3 Métodos para obtener números de tarjeta de crédito ilegalmente 21

2.3.1 Algoritmo para la generación de números de tarjeta de crédito 21

2.3.2 Ingeniería social 23

2.3.3 *Sniffer* 26

2.3.3.1 *Sniffer* en Internet 27

2.3.4 *Keylogger* 29

2.3.5 Sistemas de administración remota 30

2.3.6 *Phishing* 31

2.4 Formas de ingresar a un sistema para robar números de tarjeta de crédito 33

2.4.1 Escaneo de puertos 33

2.4.2 *Back Door* 34

2.4.3 Programas y juegos gratuitos 34

2.4.4 Archivos adjuntos 35

3. FORMAS DE REALIZAR TRANSACCIONES COMERCIALES EN INTERNET CON *WEB SERVICES* 37

3.1 *Web Services* 37

3.1.1 Definición 37

3.1.2 Componentes 40

3.1.2.1 Lógica del negocio 40

3.1.2.2 *XML* 40

3.1.2.3 *SOAP* 41

3.1.2.3.1 Estructura de un mensaje *SOAP* 42

3.1.2.4 *WSDL* 44

3.1.2.4.1 Ejemplo de código *WSDL* 45

3.1.2.5 *UDDI* 47

3.1.2.5.1 Sección blanca 47

3.1.2.5.2 Sección amarilla 48

3.1.2.5.3 Sección verde 48

3.1.3 Como funcionan los *Web Services* 49

3.2 Encaminamiento de mensajes *SOAP* 51

3.2.2 Procesamiento de mensajes *SOAP* 51

3.2.3 *WS-Routing* 53

3.2.3.1 Ejemplo de *WS-Routing* 55

3.3 Seguridad 57

3.3.1 Seguridad a nivel de transporte 58

3.3.2 Autenticación 59

3.3.3 Integridad 61

3.3.4 Confidencialidad 61

3.3.5 No repudio 62

3.3.6 Control de acceso 63

3.4 Esquema de una transacción segura con *Web Services* 64

3.4.1 Esquema de comunicación entre el cliente y la empresa que realiza la venta 64

3.5 Estándares utilizados para garantizar la seguridad en una transacción con *Web Services* 65

3.5.1 *WS-Security* 65

3.5.2 *XML Encrytion* 68

3.5.2.1 Elemento *EncryptedData* 69

3.5.2.2 Ejemplo de *XML Encryption* 70

3.5.3 *XML Digital Signature* 72

3.5.3.1 Ejemplo de un *XML Digital Signature* 74

3.5.4 *WS-ReliableMessaging* 76

3.5.5 *XCAML* 78

3.6 Estándares utilizados para garantizar transacciones con *Web Services* 79

3.6.1 *WS-Cordination* 81

3.6.2 *WS-Transaction* 82

3.6.2.1 Transacciones atómicas 82

4. FORMAS PARA REALIZAR TRANSACCIONES COMERCIALES SEGURAS CON TARJETAS DE CRÉDITO EN INTERNET 85

4.1 Requisitos de seguridad para una transacción comercial 85

4.1.1 Autenticación 86

4.1.2 Integridad 87

4.1.3 Confidencialidad 87

4.1.4 Prueba de la transacción 88

4.1.5 Gestión del riesgo y autorización 89

4.1.6 Disponibilidad y fiabilidad 90

4.2 Aspectos a considerar en una tarjeta de crédito para su uso 90

4.2.1 Seguridad 91

4.2.2 Anonimato 91

4.2.3 Divisibilidad 91

4.2.4 Autonomía 92

4.2.5 Independencia 92

4.3 Aspectos que debe cumplir una empresa que ofrece servicios y productos en Internet 93

4.4 Guía de cómo comprar seguro y de forma inteligente en Internet 94

4.5 Formas para asegurar tu información cuando navegas en Internet 97

4.5.1 Servidor seguro 97

4.5.2 Navegación anónima 98

4.5.3 Prevenir ataques 100

4.5.3.1 Uso de antivirus 100

4.5.3.2 Uso de utilerías de detección 100

4.5.3.3 Estar informado 101

4.5.3.4 Ingeniería social 101

4.5.3.5 Descarga de archivos 102

4.5.3.6 Informar a todos los usuarios del sistema 102

4.5.3.7 Desconfiar 103

5. MEDIOS ALTERNATIVOS DE PAGO 105

5.1 Tarjeta de crédito virtual 106

5.2 Cheques electrónicos 107

5.3 Dinero digital 108

5.4 Ejemplo de mecanismos alternos al pago con tarjeta de crédito 109

5.4.1 *Saf-T-Pay* 109

5.4.2 *Pay-pal* 110

5.4.3 *eCash* 112

5.4.4 *Virtu@lCash* 113

CONCLUSIONES 115

RECOMENDACIONES 119

BIBLIOGRAFÍA 121

ÍNDICE DE ILUSTRACIONES

FIGURAS

1	Transacción normal	10
2	Interrupción	11
3	Intercepción	12
4	Modificación	13
5	Fabricación	15
6	Funcionamiento de un *Web Services*	39
7	Envío de datos desde el cliente al servidor comercial	64
8	Intercambio de mensajes entre el servidor comercial y el servidor de *Web Services*	65

GLOSARIO

***ActiveX*:** es el nombre que Microsoft ha dado a un grupo de tecnologías y herramientas "estratégicas" orientadas a objetos. El principal objeto que uno crea al escribir un programa ejecutable en el entorno *ActiveX* es un componente, un programa autosuficiente que puede ejecutarse en cualquier sitio en la red *ActiveX*.

Algoritmo de cifrado de clave publica: los algoritmos de cifrado de clave pública trabajan con dos tipos de claves: una privada y otra publica, de manera que lo que se cifra con una clave, se descifra con la otra. Los algoritmos de cifrado de clave pública permiten firmar los mensajes, de forma que se pueda verificar fácilmente la autenticidad de origen, integridad y evitar el repudio de origen.

Algoritmos de clave pública: son algoritmos que emplean dos claves en lugar de una. Una de éstas, la llamada ≪pública≫, se emplea para encriptar el mensaje, que se podrá descifrar únicamente con la privada. Este sistema permite difundir sin peligro la clave pública.

Algoritmo de claves simétrica: se basan en la idea de tener una clave secreta que sirve para cifrar y descifrar, de ahí que a estos algoritmos también se les denomine algoritmos de clave simétrica.

***Background*:** significa que un programa se está ejecutando en segundo plano, es decir que se ejecuta sin que el usuario lo ponga a correr.

Clave de sesión: es la que se utiliza para cifrar los datos que vienen del y van al servidor seguro. Se genera una clave de sesión

distinta para cada transacción, lo cual permite que aunque sea reventada por un atacante en una transacción dada, no sirva para descifrar futuras transacciones.

Cookies: es información en forma de texto muy breve que envía el servidor a los usuarios, en otras palabras, es un pequeño archivo que se ubica en la *PC* del usuario para identificarlo. Sirve par que el usuario se registre solo una vez cuando abre el navegador, y, luego el sistema lo reconoce, automáticamente.

e-commerce: es el nombre en inglés que se le da al comercio electrónico.

Ethernet: es el protocolo por el cual se comunican las computadoras en un entorno LOCAL de red.

Factor ciego: consiste en multiplicar un número aleatorio por un

número determinado (conocido como factor ciego) mutando el número real y firmándolas el titular. De esta forma, el banco nunca sabrá el número real pudiendo el titular preservar su anonimato en sus compras electrónicas.

Dirección IP: cuando un navegador *Web* o su aplicación de correo-e pide datos de una página de Internet o el correo-e de otra computadora, automáticamente proporciona la dirección del equipo al que debe enviarse la información solicitada, esto último es lo que se conoce como dirección IP' (IP es protocolo de Internet). IP es un código numérico que identifica a un ordenador específico en Internet.

***IPconfig*:** es una instrucción que muestra una información detallada de datos sobre todos los adaptadores de red. Lo mejor es ejecutar un *IPCONFIG/ALL* y como la salida de datos es muy grande redirigirla a un fichero o

simplemente teclear: *IPCONFIG /ALL*.

Javascript: es un lenguaje que viaja por la red, incrustado ó insertado dentro del código *HTML* de las páginas *Web*, el cual se encarga de realizar acciones para la aplicación cliente, tales como pedir datos, confirmaciones, sacar mensajes, crear animaciones, etc.

Online: es en línea, significa que se está conectado a una red local o de Internet.

Protocolo criptográfico: es una sucesión de mensajes y respuestas que intercambian con la finalidad de establecer una comunicación segura.

Runnig Promisc: Significa que la tarjeta de red está en modo promiscuo y que recibe todos los mensajes aunque no estén

dirigidos a ella.

Spoofing: consiste en suplantar el dominio de una persona o empresa y crear cuentas de correo electrónico supuestamente de esa empresa, para que acto seguido, enviar mensajes haciéndose pasar por esa empresa.

***TCP/IP*:** es el protocolo de red que utiliza Internet, este protocolo de red es una especificación detallada de las reglas que deben seguir los diferentes programas que emplea la red de comunicación de Internet.

Troyanos: son programas que se introducen en un ordenador y los deja a merced de ese usuario que disfruta haciendo daño a los demás.

***URL*:** es la abreviatura de "*Uniform Resource Locator*". Es un modo de dirigirse a información del *Web* de forma

compacta y nada ambiguamente describe exactamente dónde se encuentra la información.

RESUMEN

Cada día se incrementa el número de personas que realiza transacciones comerciales en Internet, ya sea por su facilidad, su diversidad, rapidez y porque cada día es más factible el uso de un computador conectado a Internet, el número sigue creciendo, pero solamente un pequeño grupo de éstas conocen el peligro que pueden correr sus datos y su dinero al realizar este tipo de operaciones.

Las empresas utilizan todo tipo de seguridad, pero como todos saben nada es totalmente seguro, siempre existen fallas humanas, tecnológicas, etc., y los piratas de la red se encargan de localizar estas fallas y aprovecharse de ellas. Muchos de los usuarios del servicio no saben como lo hacen y por eso creen que es poco probable que puedan robarle su información y cometer fraude.

En el primer capítulo se dará una idea de lo que es el comercio electrónico, las categorías que existen, así como los agentes que intervienen en el proceso de una transacción comercial. Además, se dará una breve explicación de los diferentes mecanismos de seguridad que implementan los servidores para contrarrestar los ataques de los piratas y hacer las transacciones más seguras.

En el capítulo siguiente se dará una exposición de las diferentes maneras en que se puede atacar a los servidores o a los usuarios del servicio, así como los diferentes tipos de amenazas, ataques pasivos y ataques activos que se pueden dar en una transacción. También se define una serie de formas de cómo se pueden obtener los datos de las tarjetas de crédito y la tecnología que se utiliza para este propósito.

En el capítulo tres se explica cómo realizar una transacción comercial con *Web Services*, esta tecnología se perfila como una de las líderes en servicios y ya se está usando para transacciones comerciales. Se definirá la forma de localización y de ejecución de un servicio que permita realizar transacciones

comerciales y se definirá la tecnología para el envió, protección y confirmación de la transacción a través de mensajes.

En el cuarto capítulo, se dá una serie de formas para operar las transacciones más seguras para las personas que utiliza este servicio o para las que están pensando en utilizarlo. Estas reglas o métodos servirán para dar a conocer las formas en que se puede proteger la información de los usuarios antes de realizar una transacción como cuando se está realizando.

Por último, se definen tecnologías utilizadas como alternativa al pago con tarjeta de crédito y se dan ejemplos de las empresas que presentan diferentes formas de realizar el pago en Internet a través del uso de dichas tecnologías.

OBJETIVOS

- General

Definir las formas de implementar la seguridad en transacciones comerciales con tarjetas de crédito realizadas a través de Internet y, de esta forma, crear conciencia en los usuarios de este servicio de los peligros que éste implica.

- Específicos

1. Dar una idea de lo que es el comercio electrónico, la forma en que funciona y las técnicas actuales que brindan seguridad a las transacciones comerciales.

2. Mostrar los diferentes tipos de amenazas que hay en el momento de realizar una transacción comercial, así como las diferentes tecnologías que son utilizadas por usuarios malignos para obtener los datos privados de las tarjetas de crédito.

3. Definir los conceptos de las diferentes tecnologías que intervienen en una transacción comercial segura por medio de *Web Services*.

4. Brindar una metodología que consiste en una serie de recomendaciones y pasos para los usuarios del comercio electrónico para hacer, de cierta manera, más seguras sus transacciones.

5. Dar a conocer las tecnologías alternas al pago con tarjeta de crédito más funcionales y utilizadas que se disponen en Internet en estos momentos.

INTRODUCCIÓN

El uso de tecnología de telecomunicaciones para realizar todo tipo de transacción u operaciones se hace cada día más normal, no es raro ver que cada vez que esta tecnología da un paso adelante, también se encuentra una nueva forma de utilizarla para el comercio.

Con el invento de Internet se abrió una puerta para las telecomunicaciones, pero éstas eran más que todo de uso científico o educacional, no fue hasta inicios de los 90 que se pudo utilizar esta tecnología para hacer transacciones financieras o económicas; a este tipo de operaciones se le llamó comercio electrónico, ya que, además de realizar cierto tipo de operaciones permite mantener relaciones comerciales de manera directa con el comprador y el vendedor a través de redes de telecomunicaciones y equipos de cómputo.

Como toda nueva tecnología el comercio electrónico tiene sus desventajas o fallas. En este trabajo de tesis se pretende presentar las vulnerabilidades que presenta realizar transacciones comerciales con tarjetas de crédito en Internet.

Los piratas informáticos pueden penetrar la seguridad de los sitios que ofrecen el servicio de venta en línea o, bien, a las máquinas de los compradores, algunos lo hacen por diversión, otros por desafió y otros por codicia.

Existen usuarios maliciosos en Internet cuyo principal propósito es obtener información de las transacciones y la utilizan para cometer fraude a otras compañías o, bien, para descubrir quienes son personas con altos ingresos y estafarlos.

Así como existen tecnologías para obtener información privada existen tecnologías y técnicas para asegurar no en un 100% las transacciones comerciales sí en un porcentaje alto. Estas tecnologías combinadas con medidas de precaución pueden hacer que el comprar en Internet no sea tan peligroso ya que se pueden evitar los ataques de los piratas informáticos.

1. COMERCIO ELECTRÓNICO Y SU SEGURIDAD

El comercio electrónico es una metodología para hacer negocios entre empresas y consumidores por medio de Internet. En este capitulo se dará una definición de comercio electrónico, las categorías que existen, los agentes que intervienen en el proceso de una transacción comercial y los mecanismos de seguridad que implementan los servidores para contrarrestar los ataques y hacer las transacciones más seguras .

1.1 Comercio Electrónico

1.1.1 Historia

A principios de los años 90 el gobierno de Estados Unidos quito el respaldo económico que sufragaba parte de los gastos que generaba Internet, con esta medida se liberaron las medidas de restricción que concebían el Internet solamente con propósitos científicos y de entretenimiento, prohibiendo que se usara con fines comerciales. Al ser retiradas las medidas restrictivas que

pesaban sobre el comercio algunas empresas comenzaron a anunciarse en la red. En esta primera etapa Internet se utilizaba como medio publicitario no fue hasta 1994 cuando se creo la ISN (*Internet Shoppin Network*), que era una tienda que ofrecía miles de productos que podían ser adquiridos a través de Internet.

1.1.2 Definición

Existen varias definiciones de comercio electrónico o *e-commerce* como se le conoce en Internet, algunas lo ven como "el medio por el cual se puede realizar transacciones comerciales por medio del Internet", otro lo ven en si "como el medio de realizar compra y venta de bienes sin tener contacto físico entre los participantes". En cualquier definición que se de del comercio electrónico existen un punto en que todas coinciden y es en el que es "un medio que permite la compraventa de bienes y servicios por medio de sistemas electrónicos en el que las partes no tienen ningún contacto físico".

1.1.3 Categorías

- ✓ Empresa - Empresa (B2B): Este tipo de comercio electrónico es el que se realiza entre empresas, por ejemplo una empresa que realiza pedidos a un proveedor, esta se desarrolla mas que todo con redes privadas.

- ✓ Empresa - Consumidor (B2C): Este tipo de comercio electrónico es el que se realiza entre una empresa y el consumidor, de este es el que se encuentra una infinidad en Internet, a través del cual es posible comprar desde una camisa hasta una casa.

- ✓ Empresa – gobierno (B2A): Esta categoría esta surgiendo, es la que cubre todas las transacciones que se realizan entre empresas comerciales y entidades gubernativas. Esta categoría puede llegar a crecer rápidamente ya que se puede realizar el pago de impuestos o tasas corporativas a través de Internet.

1.1.4 Agentes que intervienen

En el comercio electrónico intervienen por lo menos 4 partes:

- ✓ El proveedor o vendedor que es el que ofrece los bienes o servicios a través de un sitio *Web* en *Internet*.

- ✓ El cliente o comprador que es la persona o empresa que adquiere los bienes o servicios que son ofrecidos por el vendedor en *Internet*.

- ✓ El gesto de los medios de pago, que es la empresa que establece la forma en que el vendedor reciba el pago por los bienes o servicios que son adquiridos por el cliente.

- ✓ La entidad de verificación que garantiza la autenticidad de los agentes (vendedor y cliente) que intervienen en la transacción a través de un certificado electrónico.

1.1.5 Proceso

El comercio electrónico tiene una serie de pasos que siempre se cumplen para asegurar que la transacción sea correctamente realizada.

1. El cliente se debe conectar a Internet a través de un modem, cable, fibra óptica o microonda.

2. A través de un navegador accesa a la dirección electrónica del vendedor o proveedor.

3. Ve los productos o servicios que ofrece la tienda navegando por las diferentes paginas del sitio, elige el producto o productos que desea adquirir.

4. Llena el formulario necesario para el pago con sus datos.

5. La información que el cliente proporciona es encriptada con uno de los diferentes algoritmos de encriptamiento que existe y es enviada a un servidor de transacciones conectado en línea que asegura su privacidad.

6. La información encriptada es transferida a una red de procesamientos que es verifica la autenticidad del medio de pago y que genera la orden.

7. El vendedor envía los artículos al cliente a una dirección indicada por el cliente.

8. El gestor de medios de pago se encarga de la transacción y envía el pago a al comerciante.

1.2 Seguridad en el comercio electrónico

1.2.1 SSL (*Secure Socket Layer*)

Protocolo que fue creado con el fin de establecer una comunicación segura por medio del cifrado de datos que son intercambiados entre el servidor y el cliente, esto lo hace con un algoritmo de cifrado simétrico, además proporciona el cifrado de la clave de sesión mediante un algoritmo de cifrado de clave publica, esto hace que la clave de sesión sea distinta para cada

transacción, lo cual impide que cualquier persona ajena que pueda descifrar una clave de alguna transacción no pueda utilizarla en transacciones futuras.

1.2.2 SET (*Secure Electronic Transaction*)

Este protocolo ofrece la autenticación de todas las partes involucradas en una transacción comercial por medio de certificados de autenticidad, así como la confidencialidad e integridad de los datos por medio de las técnicas de encriptamiento, evitando así que el vendedor tenga acceso a la información del pago, y que el banco no pueda ver la información de los pedidos para que no pueda formase un perfil de compra.

1.2.3 Encriptamiento

Consiste en una técnica que se utiliza para asegurar la privacidad de la información por medio de algoritmos que logran enmascararla o disfrazarla, esto permite que solamente el usuario y el servidor puedan descifrar el mensaje, así se puede evitar que si el mensaje llega caer en manos ajenas aunque la puedan leer no podrá interpretarlo.

1.2.4 Certificados digitales

Un certificado digital permite identificar la identidad de un servidor e impedir que nadie pueda suplantarlo, esto lo hace por medio de dos claves privada y pública, que no son más que dos archivos que permiten definir un conjunto de claves de encriptación y una identidad certificada.

1.2.5 Firma digital

La firma digital es un mecanismo electrónico que por medio del cifrado de una clave secreta del emisor permite verificar la autenticidad e integridad de un mensaje, además garantiza que el mensaje no va a ser rechazado.

1.2.6 PGP (*Pretty Good Privacy*)

Es un programa que por medio de algoritmo de encriptación permite enviar un mensaje garantizando que solo el destinatario pueda leerlo y brindando la autenticidad de una firma electrónica al garantizar que el mensaje que se recibe es el original, que no ha sido manipulado y que es del remitente original.

2. AMENAZAS Y ATAQUES AL COMERCIO ELECTRÓNICO

Cuando una persona realiza una transacción comercial tanto la persona como la transacción esta expuesto a diferentes amenazas y ataques, en este capítulo se definirá ataques que pueden recibir un servidor o un usuario del comercio electrónico, así como los diferentes tipos de amenazas a las que una transacción queda expuesta. Por ultimo se define una serie de formas de cómo obtener los datos de una tarjeta de crédito y la tecnología que se utiliza para este propósito.

2.1 Amenaza

Una amenaza al comercio electrónico se puede definir como la oportunidad que brinda el entorno de un sistema, ya sea el del comprador o vendedor, para que se de una violación de la seguridad de dicho sistema permitiendo que se desarrolle el fraude en el comercio electrónico.

2.1.1 Tipos de amenazas

Los ataques al comercio electrónico se pueden dividir caracterizando el sistema como un flujo de información que va del comprador al vendedor.

Figura 1 Transacción Normal

Comprador **Vendedor**

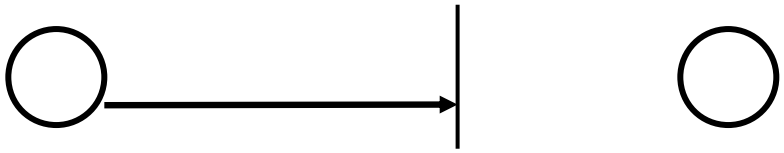
Fuente: http://www.etse.urv.es/~cmolina/XCI/files/trans_xci_v/agresion/Agresión a la Seguridad.htm

2.1.1.1 Interrupción

La interrupción a un sistema es cuando se corta el flujo de información entre las partes (comprador y vendedor) evitando de esta manera que se lleve a cabo la transacción.

Este tipo de ataque puede llegar a destruir un recurso del sistema al intentar interrumpir la comunicación, esto se puede lograr cortando la línea de comunicación (línea telefónica, cable, fibra óptica), destruyendo un elemento del hardware o bien deshabilitando el sistema de una de las partes.

Figura 2 Interrupción



Comprador **Vendedor**

Fuente: http://www.etse.urv.es/~cmolina/XCI/files/trans_xci_v/agresion/Agresión a la Seguridad.htm

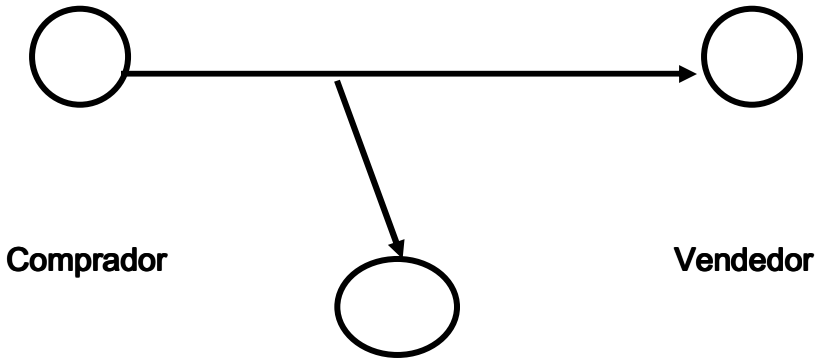
2.1.1.2 Interceptación

Este tipo de ataque se da cuando una persona, programa o computador accesa de forma ilegal a la información (numero de tarjeta de crédito, numero de orden o pedido, dirección del comprador, teléfono del comprador) que es

enviada entre el comprador o vendedor mientras se hace la transacción comercial, este ataque es a la confidencialidad de la información.

La interrupción se puede dar capturando los paquetes de información que circulan por la red con un *sniffer* o por medio de troyanos que copian archivos (intercepción de datos), también se pueden leer las cabeceras de los archivos de paquetes para averiguar la identidad de los usuarios (intercepción de identidad) mediante el uso de *spoofing*.

Figura 3 Interrupción



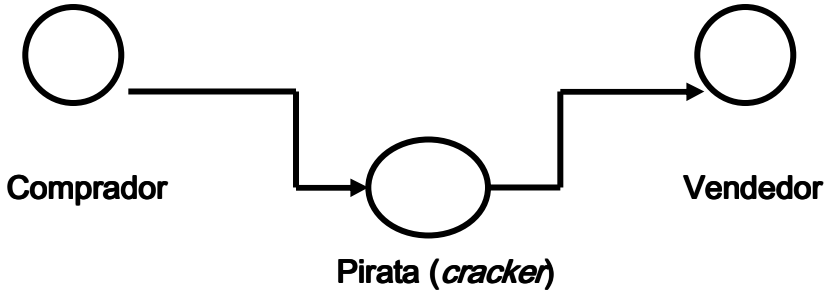
Fuente: http://www.etse.urv.es/~cmolina/XCI/files/trans_xci_v/agresion/Agresión a la Seguridad.htm

2.1.1.3 Modificación

Este tipo de ataque se da cuando una persona o programa o computador intercepta la información que se intercambia durante la transacción y la puede modificar, pudiendo así cambiar los datos del comprador por unos falsos o bien alterar los valores de la transacción, este tipo de ataque esta dirigido hacia la integridad de la información.

Como se definió en el inciso existen varias formas de capturar la información que circula en la red, así también existen varias formas de manipular la información siendo los virus y los troyanos los que cuenta con mas capacidad para hacerlo.

Figura 4 Modificación



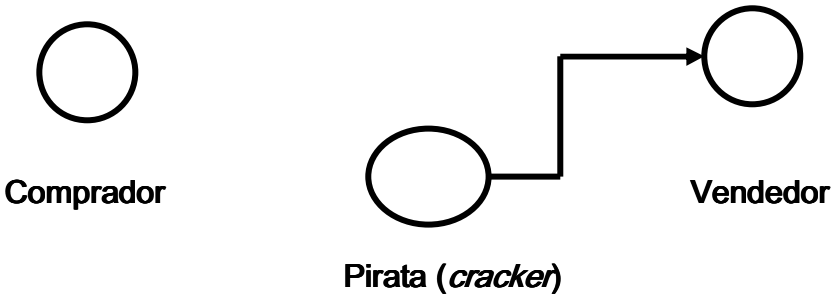
Fuente: http://www.etse.urv.es/~cmolina/XCI/files/trans_xci_v/agresion/Agresión a la Seguridad.htm

2.1.1.4 Fabricación

Una persona o entidad que no cuenta con la autorización del sistema o es bien posee una identidad falsa puede enviar o ingresar datos falsos (numero de tarjeta falso o robado) al sistema para realizar una compra, así como también puede generar formularios falsos para poder obtener información privada de un comprador; este tipo de ataque se da contra la autenticidad de la información.

Este tipo de ataque se da cuando se obtiene información de forma ilegal capturando datos de un comprador en una transacción comercial, generando datos falsos, por medio de sitios de ventas falsos o bien por medio del envió mensajes de promoción falsos donde piden los datos (nombre, dirección, teléfono, numero de tarjeta de crédito) de un usuario para después realizar transacciones ilegales con la identidad de estos.

Figura 5 Fabricación



Fuente: http://www.etse.urv.es/~cmolina/XCI/files/trans_xci_v/agresion/Agresión a la Seguridad.htm

2.2 Ataque

Un ataque al comercio electrónico se puede definir como todas aquellas acciones o eventos que violen la seguridad del sistema y que permitan poner en riesgo la confidencialidad, integridad, legitimidad y disponibilidad de la información, haciendo uso indebido de esta.

2.2.1 Ataques pasivos

Los ataques pasivos consisten en la obtención de la información por medio del monitoreo de los sistemas de comunicación, estos ataques son los

más difíciles de detectar ya que no causan ninguna modificación a la información, ni alteran el entorno en el que se encuentran.

El objetivo primordial de este tipo de ataque es el análisis de trafico y la intercepción de datos que se intercambia en una transacción comercial y aunque son difíciles de detectar también son fáciles de prevenir por medio de algunas técnicas de encriptación o bien brindando protección de la información privada por medio de *firewall.*

2.2.1.1 Obtención del origen y destino

Este tipo de ataque se lleva a cabo por medio de la intercesión de datos o paquetes que son enviados entre el comprador y vendedor en una transacción comercial, consiste en leer las cabeceras de los mensajes capturados para saber la identidad de cada uno de estos y así poder efectuar cualquiera de los ataques activos que se describen posteriormente sobre uno de los usuarios.

2.2.1.2 Control del volumen de tráfico

Este tipo de ataque se efectúa de la misma manera que el ataque anterior, solamente que en este caso la información que sIngdhl.8231e extrae sirve para llevar el control de las veces que un usuario compra, que compra y que sitios frecuenta más para realizar sus compras. También sirve para llevar el control del volumen de ventas que realiza determinado sitio.

2.2.1.3 Control de las horas habituales

En este tipo de ataque se lleva un control de las horas en que un usuario realiza sus compras y en que sitio las efectúa, esto se hace con el fin de saber a que hora un usuario realizara sus compras para poder capturar datos más importante como numero de tarjeta de crédito, dirección, nombre, teléfono y contraseña.

2.2.2 Ataques activos

Este tipo de ataque se diferencia del ataque pasivo en que en este caso existe modificación de la información que es capturada o bien la creación de información falsa para realizar transacciones fraudulentas.

Este tipo de ataque es más fácil de detectar por cualquiera de los usuarios del comercio electrónico, aunque su prevención es más difícil ya que a medida que la tecnología avanza y crea instrumentos como SSL, SET o certificados y firmas digitales que garantizan la autenticidad del comprador y vendedor así como el establecimiento de comunicaciones seguras, también los *hackers* crean nuevas maneras de falsificar la información y de interceptar los paquetes de datos que viajan en la red.

2.2.2.1 Suplantación de identidad

Consiste en realizar transacciones comerciales con el nombre de otro usuario autorizado para realizarlas, ya sea como comprador efectuando compras en sitio de ventas con tarjetas de crédito robadas, esto permite que

una persona con varios números de tarjetas de crédito pueda hacer varias comprar pequeñas que a la larga pueden llegar a una cantidad significativa.

Como vendedor se puede ofrecer el servicio de venta de un sitio que se cree es verdadero y así se procede a robar o copiar números de tarjetas de crédito y contraseña para utilizarlos posteriormente realizando compras fraudulentas.

Este tipo de ataque puede incluir uno o más de los ataques activos que son descritos posteriormente como la reactuación al capturar una secuencia de autenticación y repetirla dando a una persona o entidad los privilegios que alguien autorizado tiene para acceder a información restringida.

2.2.2.2 Reactuación

Consiste en la captura de datos verídicos, legalmente autenticados y autorizados para después retransmitirlos pudiendo por ejemplo incrementar el limite de compra de una tarjeta en la entidad que verifica los datos de dicha

tarjeta para que se autoricen las compras que hagan aunque el limite real haya sido rebasado.

2.2.2.3 Modificación de mensajes

Este tipo de ataque activo consiste en la modificación de la totalidad o parte de los datos de una transacción como el contenido de la lista de compras, la cantidad a pagar por la compra, también la modificación de los precios del catalogo.

También incluye el retardo de un mensaje para producir un efecto de autorización falso, como por ejemplo la autorización de una compra solamente que cambiando el número de tarjeta de crédito y el destinatario.

2.2.2.4 Degradación fraudulenta del servicio

Este tipo de ataque consiste en impedir el uso normal de un recurso o el servicio de comunicación de una entidad no permitiendo el envió de mensajes a un usuario, como por ejemplo avisos o notificaciones de seguridad donde

indiquen que su tarjeta esta siendo usada sin autorización o que el sitio de venta el falso ya que no esta registrado ni autenticado por ninguna autoridad.

2.3 Métodos para obtener números de tarjeta de crédito ilegalmente

2.3.1 Algoritmo para la generación de números de tarjeta de crédito

Es una forma de generar números de tarjetas validos que pueden ser utilizados para realizar compras en *Internet* ya que algunas empresas solo les basta con se les proporciones un número de tarjeta que considere valido para permitir la transacción, este algoritmo para generar números de tarjeta de crédito validos consta de varios pasos:

1. Se debe escoger un número de identificación de cuatro dígitos de un banco que respalde a una tarjeta de crédito, por ejemplo el de *Citibank* en Europa que es el 4539.

2. Se debe agregar un dígito que identifique a la entidad financiera que emitió la tarjeta, por ejemplo *American Express* = 3 y un numero de identificador

de usuarios por ejemplo 512 0398 7356. Este paso consiste en agregar los últimos 11 dígitos que identifican al usuario de la tarjeta de crédito.

3. Se debe multiplicar por dos todos los dígitos de las posiciones impares y aquellos que al multiplicar den un resultado mayor que 9, se debe de sumar el resultado.

Números impares 4,3,4,1,0,9,7,5

Multiplicación de números impares por dos

4*2=8

3*2=6

4*2=8

1*2=2

0*2=0

9*2 = 18 = 1+8 = 9

7*2 = 14 = 1+4 = 5

5*2 = 10 = 1+0 = 1

4. Después de multiplicar los números impares se debe de sumar todos los números de las tarjetas incluyendo los resultados de la multiplicación que sustituyen a los números impares originales, si el número resultante es un múltiplo de diez entonces se tiene un número de tarjeta de crédito valido, 5+9+5+2+3+8+3+6+8+6+8+2+0+9+5+1=80.

2.3.2 Ingeniería social

La ingeniería social consiste en conseguir información de personas mediante trucos o engaños ganándose la confianza de estos para que así sin estas se den cuenta revelen información sensible acerca de un sistema, de clientes o bien información propia que podría causar un daño tanto material como físico.

La ingeniería social es una técnica muy utilizada por usuarios maliciosos para obtener números de tarjetas de crédito, horarios de compras en Internet, sitios utilizados para realizar las compras, claves de acceso y toda información que podría servir para cometer fraudes a empresas que proporcionan servios o productos en Internet.

La ingeniería social en Internet se puede dar de diferentes maneras, ya que una empresa o persona puede tener su sistema protegido con tecnología de punta para brindar seguridad a los datos de una tarjeta de crédito, pero basta con un empleado o tarjeta habiente inocente para que se pueda obtener una clave de acceso o el número de la tarjeta.

Los mejores sistemas de seguridad caen simplemente por errores humanos, los usuarios maliciosos aprovechan las debilidades del ser humano como la curiosidad, el sexo, la inocencia, la avaricia o el miedo, para obtener información que pueda servir para burlar cualquier seguridad establecida.

Existen muchas formas en Internet donde se puede lograr el objetivo de tomar información a base de engaños, entre los más populares tenemos:

- ✓ Salones de *Chat*: Este es un medio muy utilizado en Internet para comunicarse con otras personas, es en estos salones en que muchos usuarios con malas intenciones se ganan la confianza de las personas y se hacen sus amigos; al ganarse la confianza de las victimas es fácil que estas proporcionen datos de sus compras, como el día, la hora, el sitio *Web* y que

se va a comprar, esta información sirve para formar un perfil de la victima y así saber que día se va a conectar para poder obtener el número de tarjeta de crédito y su contraseña por medio de un *sniffer* o *keylogger*. Existen algunas personas tan inocentes que proporcionan los datos de la tarjeta de crédito como el tipo, la fecha de vencimiento e incluso el número.

- ✓ Correo Electrónico: existen personas que utilizan este medio de comunicación para enviar información importante a otros destinatarios, parte de esta información va desde los hábitos de compra, como el lugar, la hora o el servicio o producto que se compro, hasta el tipo, el número y fecha de vencimiento de la tarjeta con la que compran; estas personas no están consientes de que este tipo de información podría caer en manos usuarios que pueden capturar los correos o bien entrar de manera ilegal a la cuenta de correo del destinatario y leer los correos que haya recibido.

- ✓ Sitios *Web* Falsos: esta forma de obtener información de las tarjetas de crédito esta muy extendida en Internet pero no es muy conocida por muchas personas. La forma más común de obtener datos de tarjetas de crédito por medio de estos sitios Web falsos es con la pornografía, existen sitios que

ofrecen fotos, videos y sexo virtual a un precio muy bajo o incluso gratis, tan solo tienen que llenar un formulario donde se le piden los datos de su tarjeta de crédito y se les proporciona el servicio. Lo que no saben las personas que ingresan a estos sitios y adquieren el servicio es que su información puede ser capturada por usuarios con malas intenciones que pueden utilizarla para cometer fraude a otras empresas y a la victima.

Existen otras formas que utilizan los usuarios maliciosos para conseguir los números de tarjetas de crédito, engañando a usuarios inocentes como cuestionarios y sorteos falsos, todo formularios en donde piden que uno ingrese los datos de la tarjeta de crédito y ofrecen regalar algo o darle mas barato puede implicar que existe alguien detrás de esto que utilizara esa información para su propio beneficio.

2.3.3 *Sniffer*

Es un programa que tiene la capacidad de capturar todos los paquetes de información que circulan en un sistema de red; aparte de capturar los datos típicos que le interesan a los *crackers* como contraseñas y nombres de usuario

también pueden capturar datos financieros como el número de la tarjeta de crédito, así como el tipo y fecha de vencimiento de la misma.

El *sniffer* se aprovecha de un defecto del protocolo *Ethernet* que generalmente envía la información a todos los ordenadores que están conectados en red aunque el paquete que se envía no este dirigido a ellos, estos ordenadores basándose en la cabecera del paquete *Ethernet* acepta la información si va dirigida a ellos si no la dejan pasar.

Lo que hace el *sniffer* para capturar la información es poner una tarjeta de red en modo promiscuo, esto significa que la tarjeta de red capturara todos los paquetes que pasen por allí aunque no vayan dirigidos a dicha tarjeta.

2.3.3.1 *Sniffer* en *Internet*

Un *sniffer* también nos permite capturar datos que circulan en Internet entre un cliente y un servidor en una transacción comercial, pudiendo de esta forma capturar los datos de la tarjeta de crédito.

Por medio del protocolo *TCP/IP* se puede capturar toda la información que circula de cliente a servidor y de servidor a cliente; el *sniffer* tiene un puerto por el que espera que se conecte un cliente haciendo una función de servicio, cuando el cliente se conecta queda establecido el *socket* servicio y el *sniffer* hace la solicitud al servidor fingiendo ser el cliente estableciendo el *socket* cliente.

Después de establecer los *sockets* toda la información que circula entre el cliente y el servidor pasa por el *sniffer*, es decir que este actúa como un puente en el que toda información que cruce debe de pasar primero por el.

Otra forma de capturar la información que viaja en Internet entre dos computadoras en colocar un *sniffer* en una de ellas, por facilidad se ponen en la computadora del cliente, este *sniffer* captura toda la información que se intercambiara entre el cliente y el servidor guardándola así en un archivo Log, también puede tener la capacidad de avisar cuando el cliente este conectado para que así el *cracker* puede leer el archivo Log para que no crezca mucho; la particularidad de este *sniffer* es que no pone la tarjeta en modo promiscuo, así que solo capturara los datos que vallan para esa computadora nada más.

2.3.4 *Keylogger*

Es un programa que tiene la capacidad de capturar las teclas que se presionan en una computadora y de guardarla en un archivo con extensión Log. Este programa es instalado en la maquina de la victima como un virus troyano aunque este no lo sea, puede venir en un programa gratuito que bajemos de Internet o bien en un correo electrónico, al ejecutar el programa o archivo de Internet este automáticamente instala un archivo con extensión INI en un directorio de nuestro disco, este queda configurado para ejecutarse cada vez que la computadora se inicie. Cada vez que se ejecuta este programa será capaz de capturar toda la información que se teclee como las claves de acceso, números, tipo y fecha de vencimiento de una tarjeta de crédito.

Una de las principales características de este programa es que se ejecuta en *background*, aunque se pulse las teclas “ctrl. + Alt + supr” que son las que nos muestra una ventana con las tareas que se ejecutan, este no aparecerá en dicha ventana y no podrá ser detectado; además de esta característica este programa aunque funciona como un virus no puede ser detectado por ningún antivirus.

2.3.5 Sistemas de administración remota

Estos sistemas de administración remota son programas que se ejecutan en una computadora que seria la victima y que le permiten acceso a todos los recursos de dicha computadora a un cliente remoto sin que la victima se de cuenta.

Este sistema consiste en dos programas, un programa que esta en la computadora de la victima que se conoce como el servidor, este programa se encarga de realizar todas las acciones que se le solicitan además es el que abre el puerto por medio del cual entran en la computadora; el otro programa es el cliente, este programa es el que se instala en la maquina del *cracker* y es el cliente, este programa es el que se instala en la maquina del *cracker* y es el que controla o hace las peticiones al programa servidor.

Para que estos programas funcionen el usuario malicioso debe de saber la dirección IP de la victima, normalmente los programas servidor mandan un mail cuando la victima se conecta e indican su IP, después se ejecuta el programa cliente y este hace la conexión con la maquina victima, en otras ocasiones el

programa cliente busca automáticamente en Internet quien tenga un puerto abierto, este puerto es el que deja el programa servidor abre al ejecutarse.

Para que se ejecute el programa primero un usuario sin conocimientos debe bajarlo de Internet, las maneras más comunes de bajar estos por medio de archivos adjuntos en el correo electrónico, por transferencia de archivos en los salones de *chat* o por fallas en la seguridad de los programas.

Este tipo de sistema de administración remota son catalogados como troyanos, los troyanos son programas que se instalan en una computador por la propia victima sin que este lo sepa, además de realizar funciones que no tienen permitidas y se ejecutan de manera invisible.

2.3.6 *Phishing*

Es una forma de estafa diseñada con la finalidad de robar los datos de identidad a las personas. El delito consiste en obtener información tal como privada como números de tarjetas de crédito o contraseñas por medio de engaños.

Este tipo de fraude se recibe habitualmente a través de mensajes de correo electrónico o de ventanas emergentes. El usuario malintencionado envía millones de mensajes falsos que parecen provenir de sitios *Web* reconocidos por las personas o de su confianza, como su banco o la empresa de su tarjeta de crédito.

Dado que los mensajes y los sitios *Web* que envían estos usuarios parecen oficiales, logran engañar a muchas personas haciéndoles creer que son legítimos. La gente confiada normalmente responde a estas solicitudes de correo electrónico proporcionando sus números de tarjeta de crédito, contraseñas, información de cuentas u otros datos personales.

Para que estos mensajes parezcan aun más reales, el estafador suele incluir un vínculo falso que parece dirigir al sitio *Web* legítimo, pero en realidad lleva a un sitio falso o incluso a una ventana emergente que tiene exactamente el mismo aspecto que el sitio *Web* oficial, una vez que el usuario está en uno de estos sitios *Web*, introduce información personal sin saber que se transmitirá directamente al delincuente, que la utilizará para realizar compras, solicitar una nueva tarjeta de crédito o robar su identidad.

2.3.4 Formas de ingresar a un sistema para robar números de tarjeta de crédito

2.3.4.1 Escaneo de puertos

El escaneo de puertos es una técnica que consiste en buscar que puertos o canales de comunicación que están abiertos y que puedan ser receptivos o de utilidad. La forma en que se lleva a cabo es ir de puertos en puerto llamando a ver si alguno contesta, si al hacer la llamada al puerto este contesta significa que el puerto esta abierto o activo, si por el contrario no contesta significa que esta cerrado u oculto.

En el escaneo de puertos se establece una relación cliente / servidor en donde el servidor es la maquina que escucha las llamadas a sus puertos, este se identifica con una dirección IP y el número de un puerto determinado; el cliente es la maquina que hace la llamada al servidor y hace la conexión a través de dicho puerto que debe estar abierto o disponible.

2.3.4.2 *Back Door*

Son trozos de programas escritos por los programadores que permiten el ingreso ilegal a un sistema o programa sin que se les autorice previamente o saltándose los métodos usuales de autenticación. Las puertas traseras como también se les conoce son dejadas a propósito por los programadores para agilizar la tarea de probar el código en la etapa de desarrollo para así no tener que estar pasando por la seguridad que requieren estos sistemas para realizar ciertas tareas.

Al finalizar la etapa de desarrollo y entregar el producto terminado se les olvida corregir el error o bien a propósito lo dejan para entrar posteriormente al sistema saltando la seguridad del mismo.

2.3.4.3 Programas y juegos gratuitos

Esta forma de colocar programas que puedan llegar a robar no solo la información de una tarjeta de crédito si no que también toda información que contenga una maquina es muy utilizada en Internet, existen un sin fin de

programas y juegos gratuitos que ofrecen una serie de opciones y de entretenimiento que a veces es difícil negarse a utilizarlos, pero que al ejecutarse instalan sin consentimiento o conocimiento de los usuarios programas que se ejecutar de manera oculta y que son capaces de robar la información privada y enviárselas a otros usuarios.

2.3.4.4 Archivos adjuntos

Esta es una de las formas comunes de enviar programas que pueden robar información de una maquina dejando abierto algún puerto, una puerta trasera o bien para colocar un *keylogger* o un *sniffer*.

Consiste en enviar en un archivo adjunto en un correo electrónico o en un archivo de transferencia de un *chat*, que al ser descargado y ejecutado instala uno programa automáticamente sin que la persona se de cuenta o autorice dicha instalación, es mas algunos han logrado que al solo abrir el mensaje de correo automáticamente el archivo se descargue y se ejecute.

3. FORMA DE REALIZAR TRANSACCIONES COMERCIALES EN INTERNET CON *WEB SERVICES*

Existen diferentes tecnologías que permiten realizar transacciones comerciales en *Internet* de manera segura, una de las que más esta destacando por su independencia de la plataforma de desarrollo y por su fácil publicación, localización e invocación mediante protocolos Web estándar son los *Web Services*. En este capítulo se detalla el uso de *Web Services*, la forma de localización y de ejecución de un servicio que permita realizar transacciones comerciales y la tecnología para el envió, protección y confirmación de la transacción a través de mensajes.

3.1 *Web Services*

3.1.1 Definición

Un *Web Service* o servicio Web es según la W3C (*World Wide Web Consortium* que es el organismo que se encarga de desarrollar una parte

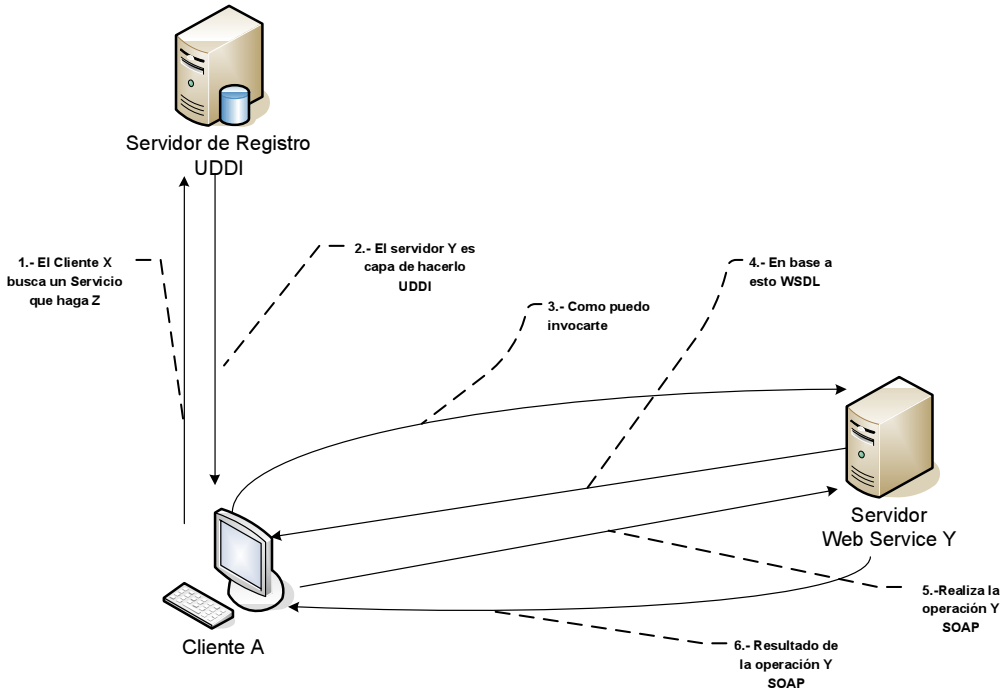
significativa de los estándares en Internet) ". Un servicio Web es una aplicación de software que se identifica mediante un *URL*, cuya interfase y uso son capaces de ser definidos, descritos y descubiertos mediante artefactos *XML* y soporta interacciones directas con otras aplicaciones de software usando mensajes basados en *XML* y protocolos basados en *Internet*".

En base a esta definición se puede definir los *Web Services* como componentes de software que tienen un formato que describe la interfaz de sus métodos y atributos basado en *XML*, un protocolo de aplicación basado en mensajes que permite que una aplicación interaccione con el *Web Service* y un protocolo de trasporte que se encarga de trasportar los mensajes por Internet.

Los *Web Services* no son aplicaciones que posean una interfaz grafica para los usuarios puedan utilizarlos, sino que son aplicaciones que accedidas por Internet por otras aplicaciones y se comunican por medio de mensajes. Son componentes que son publicados por medio de un directorio *UDDI* que contiene información de lo que hace y de su estructura *WSDL*, cuando otra aplicación o servicio *Web* desea accesarlo lo busca en el directorio y en base a sus

especificaciones le pasa los parámetros necesarios para realizar una transacción por medio de un mensaje *SOAP* con formato *XML*.

Figura 6. Esquema del funcionamiento de un *Web Service*



1. El cliente X busca un servicio que haga Z
2. El servidor Y es capas de hacer (*UDDI*)
3. Como puedo invocar el servicio
4. En base a esto *WSDL*
5. Realiza la operación Y
6. Resultado de la operación Y

3.1.2 Componentes

3.1.2.1 Lógica del negocio

Este componente es el que procesa la petición para generar la información solicitada por el cliente, es decir es el que se encarga de resolver el problema y para ello se comunica con otros *Web Services*, accesa a bases de datos o bien solicita información a otras aplicaciones. El resultado de la petición es enviado por medio de un mensaje con formato *XML*.

3.1.2.2 *XML*

XML es el acrónimo de la palabra *eXtensible Markup Language* (Lenguaje de marcado ampliable o extensible) desarrollado por W3C. Se basa en documentos de texto plano en los que se utilizan etiquetas para delimitar los elementos de un documento.

XML define estas etiquetas en función del tipo de datos que esta describiendo y no de la apariencia final que tendrán en pantalla. Una de las

principales características de este lenguaje es que permite crear etiquetas propias o ampliar las existentes. Este leguaje se destaca como estándar para el intercambio de datos entre diversas aplicaciones o software con el caso de *SOAP*.

3.1.2.3 *SOAP*

Es un protocolo simple de acceso a objetos (*Simple Object Access Protocol*), es un protocolo que permite la comunicación entre aplicaciones por medio de mensajes con formato *XML* en *Internet*. Los mensajes *SOAP* son independientes del sistema operativo y de los protocolos y puede ser transportado por medio de varios protocolos incluyendo *http*, *tcp/ip*, etc.

En los *Web Services* se utiliza para definir la comunicación, es decir define la codificación de las llamadas a los métodos de los *Web Services* y como debe se debe de codificar el resultado para que otras aplicaciones lo puedan entender.

El protocolo *SOAP* consta de 3 partes:

1. Descripción del contenido del mensaje

2. Reglas para la codificación de los tipos de datos en *XML*

3. Representación de las llamadas para la invocación y respuesta generar por el *Web Services*.

3.1.2.3.1 Estructura de un mensaje *SOAP*

```
<?xmlversion="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
  Soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
    <soap:Header>
    </soap:Header>
    <soap:Body>
      <soap:Fault>
      </soap:Fault>
```

```
    </soap:Body>
</soap:Envelope>
```

<?xml version="1.0"?>: Un mensaje *SOAP* es un documento *XML* por lo que debe de comenzar con el *tag* de *XML* y la versión.

<soap:Envelope: Indica que comienza el *envelope* del mensaje.

xmlns:soap="http://www.w3c.org/2004/12/soap-envelope": Indica la asociación del elemento *envelope* con el espacio de nombres.

Soap:encodingStyle= "http://www.w3c.org/2004/12/soap-encoding">: Indica donde se encuentran definidos los tipos de datos utilizados en el documento.

<soap:Header>: Esta línea indica el comienzo del encabezado, en esta sección se incluye información especifica del mensaje.

</soap:Header>: Indica el fin del *tag* del encabezado.

<soap:Body>: Esta línea indica que comienza el cuerpo del mensaje.

<soap:Fault>: Esta línea indica el inicio de la notificación de fallos, cualquier tipo de fallo que se produzca esta en definido en esta sección.

</soap:Fault>: Fin del *tag* de fallos.

</soap:Body>: Fin del *tag* que contiene el cuerpo del mensaje.

</soap:Envelope>: fin del mensaje SOAP.

3.1.2.4 WSDL

Web Services Description Language (Lenguaje de descripción de Servicios Web), es un lenguaje que se basa en *XML* para realizar la descripción de los *Web Services*, este lenguaje describe la interfaz de un *Web Service*

como un conjunto de métodos capaces de intercambiar mensajes, esta descripción incluye el numero de argumentos, tipo de cada uno de los parámetros utilizados en cada método, así como la descripción de los elementos que retornar.

WSDL define el lugar en el que esta disponible el servicio y los elementos necesarios para escribir un programa que pueda interactuar con el *Web Service*, es decir que es el manual de operación porque indica cuales son las interfaces y los tipos de datos necesarios para la utilización del *Web Services*.

3.1.2.4.1 Ejemplo de código *WSDL*

```
<?xml version="1.0">
<definitions>
  <types>
  </types>
  <message>
  </message>
  <portType>
```

```
  </portType>
  <binding>
  </binding>
</definitions>
```

<?xml version="1.0"?>: Debe de comenzar con el *tag* de *XML* y la versión.

<definition>: Este *tag* indica el comienzo del documento.

<types>: Aquí se definen los tipos de datos utilizados por el *Web Services*

</types>: Fin de la definición de tipos.

<message>: Aquí se definen los métodos y parámetros necesarios para realizar una operación.

</message>: Fin de la definición de parámetros y métodos.

<porType>: En esta sección se definen las operaciones que pueden ser realizadas y los mensajes que involucran.

</porType>: Fin de la definición de las operaciones y mensajes.

<binfing>: definición del formato del mensaje y detalle del protocolo por cada portType.

</binding>: Fin de la definición del formato del mensaje y detalle de protocolo.

</definition>: Fin del documento.

3.1.2.5 *UDDI*

Integración, Descubrimiento y Descripción Universal (*Universal Description, Discovery and Integration*), es un directorio de servicios *Web* distribuido y basado en *Web* que permite que se listen, busquen y descubran los servicios *Web*, este directorio permite a los posibles usuarios que puedan obtener todas la información necesaria para la invocación y ejecución de un *Web Service*.

Una entra a la lista de directorios *UDDI* es un archivo *XML* que describe un negocio y los servicios que ofrece, esta descripción contiene un archivo *WSDL* que describe la interfaz, además incluye información sobre personas de contacto, enlaces y datos técnicos para que se evalué los servicios y determinar si satisfacen las necesidades de la empresa.

3.1.2.5.1 Sección blanca

Describen los datos de la empresa como nombre, dirección, información de contactos. Es similar a la información que aparece en los directorios telefónicos, permite definir la procedencia del *Web Services* y determinar si su origen es confiable.

3.1.2.5.2 Sección amarilla

Incluye categorías de catalogación industriales tradicionales, así como su ubicación geográfica. Mediante el uso de códigos y claves predeterminadas los negocios se pueden registrar el directorio para facilitar a otros servicios la búsqueda utilizando para eso los índices de clasificación.

3.1.2.5.4 Sección verde

Contiene la información técnica que describe la interfaz del servicio con información suficiente para que se pueda escribir una aplicación que interactué con el *Web Service*.

Además brindan información complementaria para que sean más eficientes los mecanismos de búsqueda y referencia de especificaciones del servicio *Web*.

3.1.3 Como funcionan los *Web Services*

Los *Web Services* están formados por un conjunto de estándares que permiten implementar aplicaciones distribuidas que se comuniquen con otros servicios desarrollados en lenguajes y plataformas diferentes. Un *Web Service* es descrito mediante un lenguaje de descripción de servicio como el lenguaje *WSDL*, esta descripción de su interfase debe ser lo suficientemente detalla para que un usuario pueda diseñar una aplicación cliente que permita comunicase con el *Web Services*.

La descripción y las políticas de uso son publicadas en un registro conocido utilizando el método de registro *UDDI*. Las aplicaciones cliente envían peticiones al registro y reciben detalle del servicio o los servicios que se ajustan a los parámetros de búsqueda.

La llamada, uso, ejecución e instancia de los *Web Services* se hace a través de mensajes con el protocolo *SOAP*, estos mensajes viajan a través de la red por medio del protocolo *http*.

Por ejemplo si queremos crear un *Web Service* debemos en primer lugar de definir y desarrollar la lógica que vamos a ofrecer, después se crea un fichero en *WSDL* que describa la funcionalidad del servicio, el protocolo de trasporte, los parámetros necesario para interactuar con el y la dirección para ser invocado.

Seguidamente procedemos a publicar nuestro servicio en un directorio *UDDI* lo que lo hace publico y disponible para su acceso, a partir de este momento los usuarios puede utilizar el *Web Service* realizando una consulta en el directorio *UDDI*, para esto necesitan enviar al directorio un mensaje *SOAP*

con los parámetros que indiquen que tipo de empresas se busca y que servicio se necesitan. Esta búsqueda devolverá un mensaje con un listado que contiene las especificaciones y dirección de acceso de los servicios que cumplieron con los parámetros.

Con la dirección de acceso y las especificaciones se puede construir una aplicación o *Web Service* que interactué con el otro *Web Services* y que nos permita realizar una transacción comercial o bien cualquier acción que se necesite como obtener información.

3.2 Encaminamiento de mensajes *SOAP*

La mayoría de aplicaciones distribuidas hacen uso del enrutamiento de una manera u otra, esto hace posible la implementación de arquitecturas de red más flexibles que son más fáciles de ampliar y mantener conforme pasa el tiempo.

El enrutamiento se puede implementar a través de software o hardware utilizando una gran variedad de algoritmos y protocolos de red, los algoritmos

de enrutamiento pueden llevar a cabo cualquier tarea, desde la conversión de direcciones hasta un análisis de contenido mas avanzado.

3.2.2 Procesamiento de mensajes *SOAP*

Actualmente en Internet se puede encontrar una amplia lista de tipos o formas de enrutamiento, a pesar de eso los *Web Services* necesitan una solución personalidad diseñada especialmente para el envió y recepción de mensajes *SOAP*. El modelo de procesamiento de mensajes *SOAP* define tres tipos de nodos que componen una canalización de mensajes:

- ✓ Remitente *SOAP*: que es quien envía el mensaje

- ✓ Destinatario *SOAP*: que es quien recibe el mensaje al final

- ✓ Intermediario *SOAP*: es un nodo que actúa como remitente y destinatario al mismo tiempo.

Durante el procesamiento del mensaje un nodo asume una o mas funciones que determinan como se procesaran los encabezados, dichas

funciones reciben nombres únicos, cuando un mensaje llega a un nodo *SOAP*, se determina la función que asume el nodo y se realiza una acción dependiendo de la función que asume el nodo. Las funciones que pueden asumir los nodos son de ahora en adelante simplemente "*none*" (ninguno), de ahora en adelante simplemente "next" (siguiente) y de ahora en adelante simplemente "*ultimateReceiver*" (destinatario final).

3.2.3 *WS-Routing*

Es una especificación que define un nuevo elemento de encabezado para la información del enrutamiento, hace posible la definición de una ruta de acceso de reenvió para un mensaje *SOAP*, una ruta que es inversa opcional para un mensaje *SOAP* de respuesta, así como una forma de correlacionar ambas. Esto permite admitir mensajería de solicitud, o de solicitud y respuesta, de igual a igual y diálogos de larga duración. *WS-Routing* define un único

elemento de encabezado denominado *path* para la especificación de detalles de enrutamiento

```
<wsrp:path xmlns:wsrp="http://schemas.xmlsoap.org/rp">
  <wsrp:action />
  <wsrp:to />
      <wsrp:fwd>
        <wsrp:via />
  </wsrp:fwd>
      <wsrp:rev>
    <wsrp:via />
  </wsrp:rev>
      <wsrp:from />
      <wsrp:id />
      <wsrp:relatesTo />
      <wsrp:fault />
</wsrp:path>
```

<wsrp:path xmlns:wsrp="http://schemas.xmlsoap.org/rp">: encabezado del *WS-Routing*.

<wsrp:action />: indica la intención del mensaje.

<wsrp:to />: identifica al receptor final.

<wsrp:fwd>: identifica a los intermediarios del reenvío.

<wsrp:via />: identifica un nodo de intermediario.

</wsrp:fwd>: fin *tag* identificados de intermediarios.

<wsrp:rev>: identifica intermediarios inversos.

<wsrp:via />: identifica un nodo de intermediario.

</wsrp:rev>: fin *tag* intermediario inverso.

<wsrp:from />: identifica al remitente.

<wsrp:id /> : identifica de forma única este mensaje.

<wsrp:relatesTo />: correlaciona este mensaje con otro.

<wsrp:fault />: proporciona detalles adicionales sobre el error específico del enrutamiento, y se utiliza junto con el error de *SOAP* estándar.

</wsrp:path>: fin del *tag path*

Todos estos elementos son opcionales a excepción del *tag* de *action* y normalmente no se utilizan todos al mismo tiempo.

3.2.3.1 Ejemplo de *WS-Routing*

Un mensaje que va a pasar por 3 intermediarios, llega a un nodo determinado y si este pertenece a uno de los nodos definidos en el encabezado, este nodo tiene la responsabilidad de quitar el elemento vía que

lo representa y reenviar el mensaje al siguiente nodo, este procedimiento continua hasta que se llega al nodo destinatario identificado.

```
<wsrp:path soap:mustUnderstand="1"
  soap:actor="http://schemas.xmlsoap.org/soap/actor/next"
  xmlns:wsrp="http://schemas.xmlsoap.org/rp">
<wsrp:action>http://example.org/ws-routing/Echo</wsrp:action>
<wsrp:to>http://localhost/endpoints/endpoint1.asmx</wsrp:to>
<wsrp:fwd>
   <wsrp:via>http://localhost/RouterA/echo.rp</wsrp:via>
   <wsrp:via>http://localhost/RouterB/echo.rp</wsrp:via>
   <wsrp:via>http://localhost/RouterC/echo.rp</wsrp:via>
</wsrp:fwd>
<wsrp:id>uuid:fa858133-3835-4fd9-8063-c9e2ab93be73</wsrp:id>
</wsrp:path>
```

Mientras el nodo este procesando la ruta de acceso de reenvió puede de manera opcional crear una ruta de acceso inversa con el elemento **vía** que se

quita del *tag* de **fwd** y lo coloca en el *tag* **rev**, creando así la vía de regreso del mensaje.

```
<wsrp:path soap:mustUnderstand="1"
  soap:actor="http://schemas.xmlsoap.org/soap/actor/next"
  xmlns:wsrp="http://schemas.xmlsoap.org/rp">
<wsrp:action>http://example.org/ws-routing/Echo</wsrp:action>
<wsrp:to>http://localhost/endpoints/endpoint1.asmx</wsrp:to>
<wsrp:fwd/>
<wsrp:rev>
   <wsrp:via>http://localhost/RouterC/echo.rp</wsrp:via>
   <wsrp:via>http://localhost/RouterB/echo.rp</wsrp:via>
   <wsrp:via>http://localhost/RouterA/echo.rp</wsrp:via>
</wsrp:rev>
<wsrp:id>uuid:fa858133-3835-4fd9-8063-c9e2ab93be73</wsrp:id>
</wsrp:path>
```

La ruta de acceso inversa se procesa utilizando el mismo modelo que la de envió, solo es una ruta de acceso del destinatario de vuelta al remitente.

3.3 Seguridad

Los *Web Services* están orientados a la computación distribuida cuya aceptación los convierten en una implementación de gran relevancia de arquitectura orientada a servicios. Las principales características de los *Web Services* es que son piezas de software auto-contenidas, auto-descritas, modulares, estructuradas a partir del ensamblaje de componentes, pueden ser publicados, localizados e invocados a través de la *Web*, además son independientes del lenguaje de implementación, independientes de la plataforma e inherentemente basados en estándares.

La seguridad es un concepto muy importante a tomar en cuenta en la implementación de esta tecnología debido a las características mencionadas, hay que mantener los servicios de seguridad básicos como confidencialidad, integridad, autenticidad de origen, no repudio y control de acceso. También hay que tomar en cuenta la arquitectura de referencia planteada por el W3C para los servicios *Web* que hace mención a que para garantizar la seguridad en los servicios *Web* es necesario un amplio espectro de mecanismos que solventen

problemas como la autenticación, el control de acceso basado en roles, la aplicación efectiva de políticas de seguridad distribuidas o la seguridad a nivel de los mensajes.

3.3.1 Seguridad a nivel de transporte

La primera medida de seguridad que se debe de tomar cuando se va a intercambiar mensajes *SOAP* que contienen información que debe ser privada y que se transportara por medio de http es utilizar *SSL/TLS*.

SSL/TLS se compone de un conjunto de bibliotecas criptográficas que los *Web Services* pueden utilizar con el fin de proporcionar sistemas sólidos de encriptación y autenticación para transmitir datos por Internet, también nos servirá para comprobar que envía los datos al servidor correcto por medio del certificado digital.

Debido a que los mensajes viajan a través de varios nodos intermediarios es necesario asegurar la comunicación extremo a extremo, esto se logra

utilizando el protocolo *WS-ReliableMessaging*, este mecanismo la confiabilidad en la entrega de mensajes intercambiados.

Además para que el mensaje no se pierda en el camino utilizaremos el protocolo de *WS-Routing* que nos permite implementar técnicas de ruteo en el mensaje y así definir una vía para el mensaje de ida y de vuelta.

3.3.2 Autenticación

La autenticación entre el un servicio *Web* cliente y cualquier otro de los servicios con los que interactuara en una transacción comercial se hará teniendo una clase secreta con cada uno de los servicios *Web* y enviársela de manera segura.

Para el envió de la clave secreta se utilizara el mecanismo descrito por la especificación *WS-Security* que indica el empleo de elementos de seguridad de tipo *UsernameToken* definido en el perfil *Username Token Profile*. Esta especificación permite asegurar los mensajes SOAP indicando una manera de aplicar las primitivas de seguridad *XML* de firma digital (*XML Digital Signature*) y cifrado (*XML Encryption*).

Para implementar este marco de trabajo sobre mensajería distribuido *SOAP* se utiliza un módulo *SOAP* que incluye una cabecera de seguridad con el nombre de usuario y clave secreta del servicio *Web*. En cada mensaje enviado desde el servicio *Web* hacia algunos de los servicios *Web* deberá enviar su clave/contraseña de forma que demuestre su identidad. Igualmente, los servicios *Web* deberán utilizar algún mecanismo para demostrar la identidad del origen de las respuestas. Si el servicio *Web* firma digitalmente cierta parte del mensaje, los servicios *Web* receptores podrán validar la firma y comprobar así su identidad.

3.3.3 Integridad

Para garantizar a un servicio *Web* que la información que recibe es la misma que la información que fue enviada desde un sistema cliente se aplicaran las primitivas definidas en *XML Digital Signature* según lo indica el

estándar *WS-Security* que consisten en cifrar con la clave privada el resultado de aplicar una función resumen sobre el contenido del que queremos garantizar la integridad. Es decir que se va a aplicar una función de resumen sobre el contenido del mensaje y esta va a ser cifrada, si alguna parte del mensaje ha sido modificado al aplicar la misma función de resumen sobre le mensaje recibido se podrá determinar este hecho.

3.3.4 Confidencialidad

La confidencialidad de los mensajes se alcanza aplicando técnicas de cifrado sobre aquellas partes que deseamos mantener confidenciales frente a los posibles atacantes. La especificación *XML Encryption* definida por el W3C, define un modelo de procesamiento para cifrar, descifrar y formatear en *XML* datos cifrados. La manera de aplicar esta especificación sobre mensajes *SOAP* viene determinada por la especificación *SOAP Message Security 1.0* que forma parte del estándar *WS-Security*.

En el caso de que un *Web Services* se comunique con una aplicación y desee implementar el cifrado del mensaje debe de hacerlo mediante una clave

simétrica conocida por ambas partes y distribuida inicialmente por procedimientos seguros.

3.3.5 No repudio

Cuando se realizan transacciones comerciales es un requisito ser capaz de probar que una acción tuvo lugar y que fue realizada por cierta entidad. En el caso de los servicios *Web*, es necesario ser capaz de demostrar que un cliente utilizó un servicio pese a que éste lo niegue así como demostrar que un servicio fue ejecutado.

La aplicación de *XML Digital Signature* y su correspondiente soporte legal permiten garantizar el no repudio en los mensajes intercambiados entre los servicios *Web*. Para ello se utilizan los mecanismos descritos en *WS-Security* para la aplicación de las firmas digitales mediante *XML Digital Signature* ofreciendo así una solución estándar para el control del no repudio.

3.3.6 Control de acceso

Los servicios *Web* deben disponer de mecanismos que les permitan controlar el acceso a sus servicios. Se debe de poder determinar quién y cómo puede hacer a qué y cómo sobre sus recursos. La autorización concede permisos de ejecución de ciertos tipos de operaciones sobre ciertos recursos a ciertas identidades autenticadas.

Una vez se ha autenticado al solicitante y se conoce su identidad, se utilizarán mecanismos de autorización para controlar el acceso apropiado a los recursos del sistema. El control de acceso se llevara acabo por medio del estándar *XACML*, este estándar permite definir políticas de control de acceso a los servicios *Web*, haciendo posible crear políticas de seguridad complejas basadas en reglas.

3.4 Esquema de una transacción segura con *Web Services*

Después de definir las especificaciones y estándar de seguridad que se utilizan en una transacción comercial entre un *Web Services* y otras

aplicaciones que pueden ser otros *Web Services* u otra aplicación en Internet, definimos el esquema grafico de la transacción comercial.

3.4.1 Esquema de comunicación entre el cliente y la empresa que realiza la venta

Figura 7 Envió de datos desde el cliente al servidor comercial

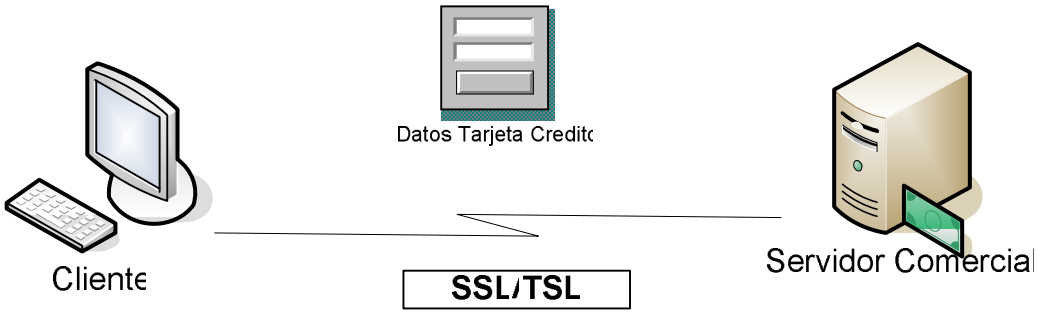
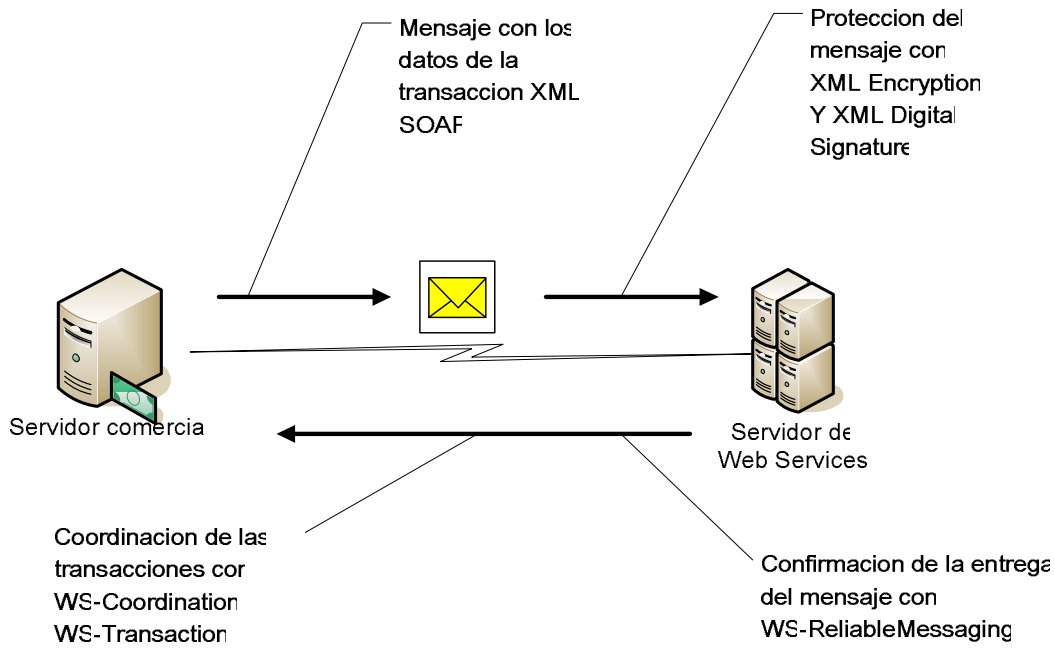


Figura 8 Intercambio de mensajes entre el Servidor Comercial y el Servidor de *Web Services*



3.5 Estándares utilizados para garantizar la seguridad en una transacción con Web Services

3.5.1 WS-Security

Esta especificación es un estándar que proporciona integridad, confidencialidad y opcionalmente no repudio a los mensajes *SOAP* intercambiados entre servicios Web.

WS-Security define los mecanismos básicos para proporcionar un marco de trabajo seguro en el intercambio de mensajes, a partir de estos mecanismos

básicos, define primitivas adicionales junto con extensiones para el intercambio de *tokens* de seguridad que permitan la emisión y propagación de las credenciales dentro de diferentes dominios de confianza.

Con el fin de garantizar la comunicación segura entre las partes, ambas deben intercambiar credenciales de seguridad (directa o indirectamente). Sin embargo, cada interlocutor necesita determinar si puede confiar en las credenciales presentadas por el otro. Las extensiones que proporciona sobre *WS-Security* son:

- ✓ Métodos para emitir, renovar y cambiar (un tipo por otro) *tokens* de seguridad.

- ✓ Métodos para establecer y acceder a las relaciones de confianza presentes.

El objetivo principal de esta especificación es, por lo tanto, habilitar a los sistemas para que puedan crear patrones de intercambio confiados de mensajes. Esta confianza se representa mediante el intercambio e intermediación de los *tokens* de seguridad. La especificación define un

protocolo agnóstico que permite emitir, renovar e intercambiar *tokens* de seguridad. La sintaxis de los *token* es muy sencilla:

```
<UsernameToken wsu:Id="...">
   <Username>...</Username>
   <Password>...</Password>
</UsernameToken>
```

El elemento *UsernameToken* se utiliza para representar una declaración de una identidad. Se puede utilizar este atributo para poder identificar local y unívocamente este *token* de seguridad. El sub-elemento *Username* contiene una cadena con la identidad declarada. Se puede agregar cualquier atributo, definido en cualquier otro esquema externo al elemento *Username*, siendo éste uno de los dos puntos posibles de extensibilidad de este elemento. El otro punto de extensibilidad es aquel que permite incorporar cualquier tipo de elemento *XML* definido en un esquema propio como elemento hijo de *UsernameToken*.

3.5.2 *XML Encryption*

Es un protocolo que especifica que se puede cifrar parte o la totalidad del cuerpo de un mensaje *SOAP*. Cuando se utiliza el cifrado *XML* se ejecuta un algoritmo en la parte del documento *XML* que se esta cifrando y estos datos *XML* se sustituyen con la información cifrada resultante dentro de un elemento *EncrypData* el cual contiene o identifica mediante una *URL* los datos cifrados.

Un documento *XML*, o el contenido de un elemento *XML* que es cifrado, es reemplazado por su correspondiente elemento *EncryptedData* en el documento *XML* cifrado de salida. Como se cifran datos arbitrarios, el elemento *EncryptedData* podría convertirse en un elemento raíz del nuevo documento *XML* de salida cuyas partes han sido cifradas.

Cuando se realiza el cifrado de datos de *XML*, se sustituye la parte cifrada por un elemento *EncryptedData* definido por la especificación en cuestión. Este elemento permite:

- Definir el algoritmo de cifrado utilizado.
- Especificar la clave de cifrado utilizada.

- Contener o referenciar los datos cifrados.

- Contemplar atributos específicos de los datos cifrados (meta-información).

3.5.2.1 Elemento EncryptedData

El elemento *EncryptedData* presenta la siguiente estructura

✓ “?” denota 0 o una ocurrencia

✓ “+” denota una o más ocurrencias

✓ “*” denota cero o más ocurrencias

✓ Una marca de elemento vacío significa que el elemento debe estar vacío

```
<EncryptedData Id? Type? MimeType? Encoding?>
    <EncryptionMethod/>?
    <ds:KeyInfo>
        <EncryptedKey>?
        <AgreementMethod>?
        <ds:KeyName>?
        <ds:RetrievalMethod>?
        <ds:*>?
    </ds:KeyInfo>?
    <CipherData>
        <CipherValue>?
        <CipherReference URI?>?
    </CipherData>
    <EncryptionProperties>?
</EncryptedData>
```

El elemento *CipherData* puede referenciar los datos cifrados. Si referencia directamente los datos cifrados éstos se representan como el contenido del elemento *CipherValue*; por otro lado, si utiliza el atributo URL del elemento *CipherReference* entonces apunta a la ubicación de los datos cifrados.

3.5.2.2 Ejemplo de *XML Encryption*

Tenemos un documento que tiene los datos de una tarjeta de crédito de la señorita Keyla Nicte Herrera Córdova quien tiene un limite de Q7000.00.

Mensaje *XML* sin encriptar

```
<?xml version="1.0" ?>
   <PaymentInfo xmlns="http://ejemplo.org/pagoTarjeta">
      <Nombre>Keyla Nicte Herrera Cordova</Nombre>
      <Limite="7,000" Moneda="QUET">
         <Numero>1345 5678 9012 3456</Numero>
         <Financiera>Credomatic</Financiera>
         <Expira>06/06</Expira>
      </CreditCard>
```

```
</PaymentInfo>
```

La información del número de tarjeta debe ser debidamente protegida contra posibles ataques, para que la aplicación mantenga esta información de manera confidencial, se puede cifrarla en un elemento llamado *CreditCard*.

Mensaje *XML* encriptado

```
<?xml version="1.0" ?>
   <PaymentInfo xmlns="http://ejemplo.org/pagoTarjeta">
      <Nombre>Keyla Nicte Herrera Cordova</Nombre>
<EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Element">
         <CipherData>
            <CipherValue>A23B45C56</CipherValue>
         </CipherData>
      </EncryptedData>
   </PaymentInfo>
```

Cifrando el elemento *CreditCard* desde su marca de comienzo hasta su marca final, la propiedad identidad del elemento queda escondida. De esta

manera alguien que pueda ver el mensaje no sabe que tipo de operación se esta llevando. El elemento *CipherData* contiene la serialización cifrada del elemento *CreditCard*. Como también puede observarse, el elemento *EncryptedData* ha sustituido el elemento *CreditCard* ya que es éste el que ha sido cifrado.

3.5.3 *XML Digital Signature*

Una firma digital es creada después de pasar el texto de un mensaje *SOAP* a través de un algoritmo de *hash*, esto genera un mensaje comprimido, este mensaje comprimido es luego encriptado empleando la clave privada del individuo que está generando el mensaje, transformándolo en una firma digital.

La firma digital sólo puede ser desencriptada empleando la clave pública de ese mismo individuo. El receptor del mensaje desencripta la firma digital y recalcula entonces el mensaje comprimido. El valor calculado de este nuevo mensaje comprimido se compara con el valor del mensaje comprimido hallado en la firma. Si los dos cálculos son iguales, significa que el mensaje no ha sido alterado.

Las firmas digitales *XML* se aplican sobre contenido digital arbitrario mediante una dirección o referencia. Los objetos de datos son resumidos, es decir se aplica una función de '*hashing*' sobre ellos, y el valor resultante es ubicado en un elemento del documento *XML* (junto con otra información) y ese elemento es a su vez resumido de nuevo y criptográficamente firmado. Estas firmas se representan mediante el elemento *Signature* que posee la estructura, '¿' significa 1 ó mas ocurrencias y '* ' significa 0 ó más ocurrencias

```
<Signature ID?>
   <SignedInfo>
      <CanonicalizationMethod/>
      <SignatureMethod/>
      (<Reference URI? >
         (<Transforms>)?
         <DigestMethod>
         <DigestValue>
      </Reference>)+
   </SignedInfo>
```

<SignatureValue/>

(<KeyInfo>)?

(<Object ID?>)*

</Signature>

Las firmas digitales están asociadas a los objetos de datos vía *URL* incluidas como atributos en los elementos *Referente*, dentro de un documento *XML*, las firmas están relacionadas a los objetos de datos locales mediante identificadores de fragmentos.

3.5.3.1 Ejemplo de un *XML Digital Signature*

<Signature Id="**FirmaDigital**" mlns="**http://www.w3.org/2005/xmldigsig#**">

<SignedInfo>

<CanonicalizationMethod Algorithm="**http://www.w3.org/RECxml-c14n-050530**" />

<SignatureMethod Algorithm="**http://www.w3.org/xmldigsig#dsa-sha**" />

<Reference URI="**http://www.w3.org/REC-xhtml-050530/**">

```
    <Transforms>
        <Transform Algorithm="http://www.w3.org/REC-xm050530" />
    </Transforms>
    <DigestMethod
        Algorithm="http://www.w3.org/xmldigsig#sha" />
        <DigestValue>j6lwx3rvEPO0vKtMup4NbeVu8nk=</DigestValue>
</Reference>
</SignedInfo>
<SignatureValue>KNHC023456KTF=...</SignatureValue>
    <KeyInfo>
        <KeyValue>
            <DSAKeyValue>
            </DSAKeyValue>
        </KeyValue>
    </KeyInfo>
</Signature>
```

El elemento requerido *SignedInfo* representa la información que se está firmando realmente. El elemento que es normalizado, según el algoritmo definido en el elemento *CanonicalizationMethod* y que es firmado es el

elemento *SignedInfo*. La validación fundamental de la firma consiste en dos pasos obligatorios:

- Validación de la firma: realizada sobre el resultado de firmar el elemento *SignedInfo* y que es transportado como valor textual del elemento *SignatureValue*.

- Validación de la referencia: validación de cada elemento '*digest*' incluido en cada elemento *Reference*, incluido a su vez dentro del elemento *SignedInfo*. Esta validación comprueba la integridad de los objetos de datos que han sido firmados.

El elemento *CanonicalizationMethod* refleja el algoritmo utilizado para normalizar o normalizar el elemento *SignedInfo* antes de calcular su valor de digestión como parte de la operación de cálculo de la firma. Por su parte, el elemento *SignatureMethod* indica el algoritmo utilizado para generar la firma, es decir, indica la operación utilizada para convertir el '*digest*' de la forma normalizada del elemento *SignedInfo* en el valor expuesto por el elemento *SignatureValue*.

Cada elemento *Reference* incluye el algoritmo '*digest*' y el valor resultante de aplicarlo sobre el dato referenciado. Un objeto de datos es firmado calculando el valor de su '*digest*' y firmando ese valor. La firma es más tarde verificada mediante la referencia a los objetos de datos y el proceso de validación de firma.

El elemento *KeyInfo* indica la clave que debe ser utilizada para validar la firma. Posibles identificaciones incluyen certificados, nombres de claves y algoritmos de acuerdo de clave, este elemento es opcional.

3.5.4 *WS-ReliableMessaging*

Es un protocolo que permite la entrega confiable de mensajes entre aplicaciones distribuidas, definiendo las funciones que garantizan una entrega de mensajes eficaces, asíncronos y confiables. La arquitectura de *WS-ReliableMessaging* admite interacción con otras especificaciones y estándares de mensajería y servicios Web permitiendo no volver a diseñar los esquemas de mensajes en el nivel de aplicación.

Este protocolo no esta enlazado a protocolos o sesiones de transporte subyacentes por lo que la duración de una conversación puede abarcar largos periodos de tiempo, de este modo las conversaciones se pueden suspender durante la transmisión y reanudarse sin necesidad de retrasmitir toda la conversación.

Además este protocolo cuenta con varias funciones que facilitan una transferencia bidireccional de gran tamaño, además no impone restricción arbitraria sobre el numero de mensajes pendientes en transito, permitiendo que dos extremos pueden tener una serie de mensajes confiables en transito en cualquier dirección a través de una única conexión de transporte.

3.5.5 XCAML

Es un lenguaje basado en *XML*, cuya finalidad es definir políticas de control de acceso. Es decir que proporciona una sintaxis para gestionar el acceso a los recursos por parte de ciertos usuarios que desean realizar operaciones sobre dichos recursos.

XACML es un estándar que describe tanto un lenguaje de políticas como un protocolo petición/respuesta para el control de las decisiones de autorización. Tanto el lenguaje como el protocolo, están definidos en *XML*. El lenguaje de políticas *XACML* se utiliza para describir requisitos de control de acceso generales, y tiene unos puntos de extensión estándar que permiten la definición de nuevas funciones, tipos de datos, combinaciones lógicas, etc.

El protocolo petición/respuesta nos permite componer una petición para solicitar si cierta acción debería, o no, ser permitida, además de ofrecernos formas de interpretar las decisiones de control. La respuesta siempre incluye una contestación que refleja si la petición debería ser o no permitida utilizando uno de los siguientes cuatro valores:

- *Permit* (permitir)

- *Deny* (denegar)

- *Indeterminate* (no se pudo tomar una decisión porque ocurrió un error o fue omitido algún valor requerido)

- *NotApplicable* (la petición no puede ser contestada por este servicio).

3.6 Estándares utilizados para garantizar transacciones con *Web Services*

Mientras los modelos de transacciones planas se adaptan bien a un modelo de negocios tradicional, las transacciones comerciales con *Web Services* pueden crear nuevos e interesantes desafíos para este tipo de transacciones.

El conjunto actual de especificaciones para servicio *Web* define protocolos para la interoperabilidad entre *Web Services*, esto permite la participación de un número grande de participantes formando grandes aplicaciones distribuidas. El

resultado de estas actividades que resultan pueden ser estructuras complejas, con relaciones complejas entre sus participantes.

WS-Coordination y *WS-Transaction* define el mecanismo para mantener la coordinación de contenidos a través de diferentes mensajes *SOAP*, permitiendo de esta manera un escenario de transacción entre los *Web Services*. Los beneficios de tener estos protocolos estándar para transacciones entre servicios *Web* incluyen lo siguiente:

- ✓ Una lógica transaccional que es inherentemente a lo complejo: si existe un protocolo estándar disponible, esto permite crear soluciones propias. Los desarrolladores pueden utilizar estas soluciones para el desarrollo rápido de aplicaciones.

- ✓ Adhesión a los estándares de la industria ayuda a construir soluciones ínter operables. Definir protocolos estándar para transacciones de *Web Services* ayudara a alcanzar una mejor interoperabilidad.

3.6.1 *WS-Cordination*

WS-Coordination define un marco extensible de coordinación. *WS-Coordination* no es por si mismo un protocolo, es únicamente un marco de trabajo con el cual se puede coordinar las actividades de las aplicaciones distribuidas de los *Web Services*.

WS-Coordination define una entidad coordinadora, por ejemplo un servidor coordinador con una dirección *URL* el cual proveer servicios de coordinación a aplicaciones que quieren participar en una transacción. El coordinador provee los servicios de activación y registro.

Cualquier aplicación que quiera ser parte de la transacción primero debe contactar al coordinador y solicitar el servicio de activación, el servicio de activación creara una nueva instancia y se encargara de entregara el contexto a los participante que lo requieran. El contexto de coordinación es actualmente una estructura que contiene un identificador de la actividad y la información necesaria para identifica al participante y la actividad que realiza.

El participante que esta solicitando también solicitara el servicio de registro para registrar su rol en la transacción, el rol depende del tipo de actividad que va a realizar el participante. Los roles dependen del tipo de actividad y de como el participante esta implicado en la transacción.

3.6.2 *WS-Transaction*

WS-Transaction es un protocolo que define dos tipos de actividades que puede usar el *WS-Coordination*. Para cada actividad *WS-Transaction* define la secuencia actual de operación, los variados protocolos transaccionales y los diferentes roles de las aplicaciones participantes.

3.6.2.1 Transacciones atómicas

Las especificaciones de *WS-Transaction* definen un conjunto de protocolos que soportan transacciones atómicas, el término de transacciones atómicas no es específico de los *Web Services*, es un concepto que se conocen en aplicaciones de bases de datos. Este protocolo define el concepto de

transacciones atómicas en base al concepto de transacciones atómicas de bases de datos.

Una transacción atómica en *Web Services* tiene las siguientes características:

- ✓ El resultado de la transacción es de todo o nada, es decir que o se cumple la transacción para todos los participantes o no se cumple para ninguno, las actividades envueltas en una transacción atómica son indivisibles, el conjunto completo de actividades o se completa o falla.

- ✓ Una transacción debe tomar el menor tiempo posible para ejecutarse y liberar los recursos lo más pronto posible.

- ✓ únicamente aplicaciones confiables deben de tener acceso al proceso de transacción, debido a que usuarios maliciosos pueden bloquear los recursos indefinidamente y denegar el servicio.

- ✓ Cuando son varios los participantes en una transacción el coordinador pregunta a todos los participantes se completo la transacción, si todos votan que si la transacción se de cómo cometida, si uno vota que no o no vota la transacción se da como fallida.

4. FORMAS PARA REALIZAR TRANSACCIONES COMERCIALES SEGURAS CON TARJETAS DE CRÉDITO EN INTERNET

En este capítulo se definen una serie de formas para operar transacciones más seguras por parte de los usuarios que utilizan el comercio electrónico y para las empresas que prestan ese servicio. Estas reglas o métodos servirán para dar a conocer como se puede proteger la información de los usuarios antes de realizar una transacción, durante la transacción y cuando esta ya fue realizada.

4.1 Requisitos de seguridad para una transacción comercial

Para asegurar la seguridad en una transacción de tipo comercial es necesario que se cumplan ciertos requisitos, estos son mayores cuando la transacción se realiza por medio de un sistema electrónico y en el cual los participantes no tienen contacto físico.

4.1.1 Autenticación

Es una transacción comercial que se realiza en línea es indispensable saber que la persona o empresa que esta del otro lado es quien dice ser; por la empresa o banco que la respalda para que este compruebe la veracidad de los datos y si tiene capacidad de realizar dicha compra; también se a implemento un PIN que no es más que una serie de número de identificación personal que están asociados con el número de la tarjeta, es decir que antes de dar la autorización se tiene que comprobar la correcta asociación entre el número de tarjeta y el *pin* identificador.

La autenticación del vendedor es a través de certificados digitales que garantizan la identidad del vendedor y que impiden que otra persona o empresa se puedan hacer pasar por dicho vendedor. La autenticidad del resto de los agentes que intervienen en la transacción que garantiza mediante protocolos criptográficos de autenticación basados los algoritmos de clave simétrica; aunque muy pronto serán remplazados por los algoritmos de clave pública para aprovechar la ínter polaridad de los sistemas.

4.1.2 Integridad

La integridad de los datos en otro de los aspectos importantes en una transacción comercial en donde esos datos se refieren a importes de pago, número de tarjetas de crédito y número de identificación personal.

La integridad en los protocolos de comunicación debe ser garantizada por medio de códigos de autenticación de mensajes, *SSL* y firma digitales que por medio del cifrado de datos se encargan de asegurar la autenticidad e integridad de los mensajes y de la comunicación.

4.1.3 Confidencialidad

Los datos privados como el número de autorización de una transacción, el número, la fecha de vencimiento y el número de identificación personal de una tarjeta de crédito que se intercambia en una operación comercial necesitan ser vistos solamente por los interesados o participes de dicha operación.

La confidencialidad de los datos normalmente es garantizada por medio de la encriptación de los datos a través de algoritmos de encriptación, en otros cosas para garantizar la confidencialidad de los datos se requiere que el pago se haga anónimo para que no se pueda seguir su rastro, este tipo de pago requiere de algoritmos más complejos ya que el pago se hace por medio de cuentas bancarias en donde los datos del titular son guardados hasta el momento de la autorización.

4.1.4 Prueba de la transacción

Cuando se realiza una transacción comercial en las que se mueve una cantidad de dinero para asegurar el envió de algún producto o la prestación de un servicio es necesario que exista una prueba que demuestre que la transacción fue realizada y que ha sido pagada en conformidad con las partes para que permita al comprador reclamar el producto o servicio y al vendedor que no se le niegue el pago.

Generalmente intervienen dos instituciones para garantizar la transacción, el banco emisor que el depositario de la cuenta del cliente que debe pagar el

monto de la compra al banco adquiriendo que es el depositario de la cuenta el comercio; cuando se realiza el pago el banco emisor demanda una prueba de que el pago fue realizado y la transacción fue realizada.

4.1.5 Gestión del riesgo y autorización

Otro de los aspectos importantes a la hora de dar la autorización o no de un pago en una transacción comercial es la estimación del riesgo que supone autorizar un pago que puede ser fraudulento. Para autorizar un pago tratando de evitar cualquier tipo de fraude por parte del cliente el comercio comprueba la veracidad de los datos del cliente por medio del número de tarjeta e identificación personal enviándolos a una entidad que se encarga de la verificación de los datos.

Además de verificar la autenticidad de los datos existen otros requisitos que se deben cumplir antes de dar la autorización como saldo disponible, límite de compra establecido y que no aparezca en la lista negra que identifica a los compradores que han tratado o cometido algún fraude.

4.1.6 Disponibilidad y fiabilidad

La disponibilidad y fiabilidad de una transacción comercial depende de la disponibilidad y fiabilidad de los dispositivos y sistema electrónicos de comunicación sobre los cuales se sustenta.

Una transacción debe ser atómica, es decir que o se produce de manera satisfactoria o no se produce, pero no puede quedar es un estado desconocido o pendiente, es por eso que los sistemas deben de poseer partes y protocolos específicos de sincronización.

4.2 Aspectos a considerar en una tarjeta de crédito para su uso

En el momento de escoger una tarjeta de crédito como medio de pago en Internet se deben de considerar varios aspectos que la ideal para este tipo de servicio.

4.2.1 Seguridad

Debido a que todo lo digital es sumamente fácil de copiar, reutilizar, duplicar y falsificar es necesario que los datos de una tarjeta de crédito estén protegidos tanto durante la transacción como en la base de datos que guardan los comerciantes.

4.2.2 Anonimato

Al realizar el pago de alguna transacción en Internet es posible que dejemos un rastro fácil de seguir por personas que aprovechándose de los fallos con los que cuenta la digitalización, a medida que crece esta tecnología también crece el riego de que existan mas intromisiones a la privacidad.

4.2.3 Divisibilidad

El pago de servicios o productos que permiten el pago solamente de cantidades pequeñas como el *Millicent* o bien sistemas que diferencia los pagos como *CyberCash* y *CyberCoin* que permiten pagos de grandes y pequeñas

cantidades respectivamente; inclusive existen sistemas que permiten el pago indistintamente de la cantidad como las tarjetas inteligentes.

4.2.4 Autonomía

Para que una transacción se lleve a cabo existen entidades que autorizan los pagos y que realizan las transacciones, este tipo de esquema implica fuertes gastos de infraestructura, limitaciones y dependencia para los comerciantes y los clientes.

4.2.5 Independencia

Existen restricciones a ciertos sistemas, redes, organizaciones o países que limitan de una tarjeta de crédito como medio de pago que limitan así el número de productos o servicios que se pueden adquirir en Internet; lo ideal sería que una tarjeta de crédito sea independiente de esas limitaciones y que se pueda utilizar en cualquier parte del mundo.

4.3 Aspectos que debe cumplir una empresa que ofrece servicios y productos en Internet

Debido a los casos de transacciones fraudulentas que se ven a diario en Internet debido a la facilidad de suplantar a una empresa, de duplicar transacciones y robar bases enteras de los comercios con números de tarjetas de miles de clientes es necesario cumplir con ciertas reglas para verificar que una tienda virtual realmente es segura y que se puede comprar en ella sin ningún temor.

- ✓ Se debe de buscar empresas responsables que tengan políticas de devolución del dinero si no se esta satisfecho con la compra.

- ✓ La empresa debe de contar con una dirección física en donde se pueda contactar por teléfono, fax o código postal.

- ✓ Si una empresa ofrece productos o servicios con precios extremadamente bajos que no se ven en ningún lado hay que tener precaución ya que algunas empresas que comienzan ofrecen estos descuentos pero muchas

veces desaparecen y dejan a muchos clientes sin productos o con productos de mala calidad.

- ✓ La empresa en donde se vaya a realizar compra debe de poseer un certificado digital que asegure su identidad, además de poseer una conexión segura para manejar datos de la tarjeta de crédito.

- ✓ Se debe de verificar las políticas de seguridad de la empresa o comercio que maneja las tarjetas de crédito ya que después de tantos fraudes podrían asesorar antes de realizar la compra.

- ✓ Se debe de revisar la política de entrega y cobro de las empresas para no ser engañado después con incrementos en el monto de transacción.

4.4 Guía de cómo comprar seguro y de forma inteligente en Internet

- ✓ Antes de suministrar información confidencial por Internet especialmente en lo que a datos financieros se refiere se debe de seguir una serie de consejos

para que así la información que damos tenga menos riesgos de caer en manos ajenas.

- ✓ No se debe entregar más información que la que se considera estrictamente necesaria para completar la transacción; no se debe dar información acerca de nuestro ingreso anual, religión, familiares o trabajo.

- ✓ No se debe entregar datos si no sé esta seguro de que se trasladaran en un servidor seguro.

- ✓ No se debe enviar el número de tarjeta de crédito o número de identificación personal por correo.

- ✓ Exigir imágenes del producto que se esta comprando para que así se tenga una idea de lo que se va a adquirir.

- ✓ Exigir información detallada del producto como el precio, costo de envió, garantía y devolución.

- ✓ Visitar y leer la pagina de política de privacidad del comercio, para así poder saber que pasa con la información privada que se entrega al llenar formularios y la que se entrega indirectamente por la navegación; si no existiera se debe exigir.

- ✓ Procurar el uso para navegar de las últimas versiones de los navegadores y mantenerse al tanto de las actualizaciones.

- ✓ Instalar los parches que reparen las vulnerabilidades de los exploradores que usamos para navegar.

- ✓ Saber que existen varios navegadores que pueden ser más seguros que los más usados como *IExplorer* y el *Netscape*.

- ✓ Mantener la actitud preventiva cuando se navegue en Internet.

- ✓ No brindar fácilmente nuestros datos, es importante saber que los datos que brindemos al llenar cualquier formulario podrían llegar a manos de terceras personas.

- ✓ Mantener un criterio propio respecto a los sitios que visitamos, el hecho de que exista un certificado no quiere decir que nuestros datos están 100% garantizados.

4.5 Formas para asegurar tu información cuando navegas en Internet

Existen muchas formas de robar la información de una persona por medio de programas que se ejecutan en la maquina de la victima y que permiten a los *cracker* revisar todos los datos y tomar aquellos que puedan causarnos algún tipo de daño; pero también existen muchas maneras de prevenir estos ataques, de evitar y de eliminarlos.

4.5.1 Servidor Seguros

Cuando se este comprando en Internet o llenado algún formulario donde se deba ingresar datos en los cuales se requiere que estén protegidos es necesario comprobar que se esta haciendo en una pagina con servidor seguro.

La manera más sencilla de determinar si es un servidor seguro es fijando en la dirección *URL* de la pagina, normalmente comienzan con http://, pero cuando se trata de un servidor seguro esta dirección *URL* debe comenzar de la siguiente manera https:// en donde se agrega la letra s para indicar que los datos estarán protegidos.

En navegadores como *Netscape* aparece además de aparecer la letra s en el URL aparece una llave en la parte inferior izquierda que normalmente aparece partida en sitios que no cuentan con un servidor seguro; con el *IExplorer* aparece en la parte inferior izquierda un candado cerrado para indicar que es una pagina con servidor seguro.

4.5.2 Navegación Anónima

Para poder navegar a través de Internet de una manera anónima y que nuestra dirección *IP* quede oculta existen tres formas de hacerlo:

- ✓ Configurar las opciones del navegador de una manera correcta nos ayuda a mantener de cierta forma de anonimato en Internet, dependiendo de

preferencias de seguridad que establezcamos y si deshabilitar las *cookies*, *ActiveX* y *Javascript* en nuestro navegador se puede mantener la privacidad mas no ocultar nuestra dirección *IP*, también hay que actualizar los navegadores con los parches que se descargan en Internet.

- ✓ Se puede navegar a través de servicios de *Web* de navegación anónima, que son páginas que están en medio de usuario y el sitio visitado manteniendo así nuestra dirección IP anónima. Cada vez que visitemos un sitio a través de este servicio este sitio es el que hace la petición y es el que recibe la respuesta, por lo tanto podemos navegar sin que se conozca nuestra dirección *IP*.

- ✓ Navegación por medio del servidor *Proxy* anónimo, este tipo de servicio es similar al anterior ya que también se interpone entre el usuario y el sitio visitado, es decir que nuestra dirección *IP* no será conocida, la desventaja de este tipo de servicio es que no establece ningún tipo de filtro sobre el contenido que se recibe y el que sale y por consiguiente acepta cookies, *ActiveX* y *Javascript*.

4.5.4 Prevenir ataques

Una de las maneras mas efectivas de evitar un ataque de un *cracker* es previniéndolo, como sabemos podemos ser atacados de muchas maneras diferentes por eso es necesario estar preparado y prevenir estos ataques.

4.5.3.1 Uso de antivirus

Mantener vacunada la computadora siempre es una buena forma de alejar a los *hacker* de la información, debido a que muchos de los programas que sirven para robar la información son virus troyanos que pueden llegar a enviar la información a un computar remoto.

4.5.3.2 Uso de utilerías de detección

El uso de programas de detección es muy importante ya que algunos programas aunque funcionen y se instalen como virus no lo son y es por eso que ningún antivirus los puede encontrar; existen programas que son capaces de encontrar estos espías y de eliminarlos.

4.5.3.3 Estar informado

Una de las mejores maneras de prevenir un ataque es estar bien informado de lo que tiene que ver con seguridad, virus y troyanos; existen paginas que se dedican a difundir estos temas y como se pueden encontrar y quitar.

4.5.3.4 Ingeniería social

Siempre hay que tener cuidado con lo que se diga y a quien se diga, como se explicó la ingeniería social es una técnica que se utiliza para sacar información a base de engaño, por eso es importante no revelar información importante ni financiera en *chat* y correo electrónico; además cuando una persona que dice representar a una empresa pida información es mejor no dársela en el momento y verificar si es quien dice que es.

4.5.3.5 Descarga de archivos

Antes de descarga un archivo, programa o juego es importante ponerse a pensar que muchos de estos archivos pueden contener virus y otros programas que son capaces de leer nuestra información del disco y del teclado y enviársela a otra computadora.

Si el programa parece que hace maravillas o el juegos es muy entretenido es raro que alguien lo regale, por eso se debe pensar dos veces el descargar e instalar estos archivos ya que puede que venga otro programa oculto y que el usuario mismo lo instale y abre un hueco en la seguridad del sistema.

4.5.3.6 Informar a todos los usuarios del sistema

De nada sirve tener el mejor antivirus, utilizar navegación anónima o bien ejecutar programas de detección de espías si va a llegar otro usuario y se pone a navegar en cualquier tipo de paginas, a descargar archivos, a leer mensajes o a chatear con alguien desconocido y dejar el sistema a merced de cualquier ataque.

Es importante que todos los usuarios del sistema sigan las mismas normas de seguridad ya que solo con bajar un archivo infectado o de ejecutar algún programa gratuito y todos los esfuerzos por proteger la información serán inútiles.

4.5.3.7 Desconfiar

Talvez la mejor forma de prevenir una ataque es la desconfianza, siempre dudar el contenido de un archivo, de un amigo que pregunta por nuestro número de tarjeta de crédito, de un mensaje de amor de alguien que no conocemos o de un sorteo que solo con ingresar datos importante ofrece premios que siempre se han soñado. La desconfianza puede mantener el sistema seguro, pero no se debe confundir con la paranoia, tampoco dejar que los temores impidan disfrutar todos los recursos que ofrece Internet, esos recursos que pueden facilitar que existencia o de la información que se encuentre y que ayude al crecimiento intelectual.

5. MEDIOS ALTERNATIVOS DE PAGO

Uno de los mayores obstáculos técnicos y psicológicos que impiden que el comercio electrónico tenga el auge que se espera es la forma de pago que este sistema implica.

La tarjeta de crédito que es el medio mas utilizado para realizar los pagos en el comercio electrónico padece de muchas fallas que no la hacen el medio de pago más ideal para este tipo de sistema; aunque brinda comodidad y es aceptada a nivel mundial, es necesario que exista una institución que garanticé la autenticidad de la información y que aprueba las transacciones, provocando un incremento en el tiempo de la transacción que a veces puede llegar a ser molesto, además con esa forma de pago no existe el anonimato del comprador, quedando expuesto los datos personales para que puedan ser usados ilegalmente.

Por esta razón desde hace un tiempo se han estado desarrollando nuevas formas de pago e implementando tecnología que garanticen la seguridad de los

datos, brinde la comodidad de una tarjeta de crédito, que mantenga el anonimato del comprador y que reduzcan el tiempo de cada transacción.

5.1 Tarjeta de crédito virtual

Esta tipo de tarjeta funciona exactamente como funcionan las tarjetas de crédito normal con la salvedad que no existen físicamente. El usuario debe de solicitar la tarjeta al banco o entidad financiera que ofrece el servicio indicando el número de tarjeta de crédito o numero de cuenta al cual debitaran los pagos efectuados, la entidad emisora le proporcionara un numero de tarjeta y un pin que deberá proporcionar cada vez que realiza una compra junto con el número de tarjeta, este tipo de tarjeta funciona de dos maneras.

La primera es que su limite es de cero, cuando el comprador desea realizar una compra en Internet, debe de solicitar una cantidad igual a la que va a gastar en la compra, la entidad en ese momento le amplia el limite y este puede realizar la compra dejando su limite a cero nuevamente.

La principal ventaja de este tipo de tarjeta es que si la información de la tarjeta caerá en manos de usuarios malintencionados estos no podrían realizar ninguna compra ya que el limite es de cero hasta que el dueño de la tarjeta solicite que se le argue dinero a la tarjeta virtual.

La segunda forma en la que actúa este tipo de tarjetas es que se les asigna un numero de tarjeta diferente cada vez que va a realizar su compra, el usuario debe de solicitar el nuevo numero a la entidad financiera para realizar una nueva compra, después de realizar la comprar el numero de tarjeta caduca y ya no es valido para hacer otra compra.

Otra ventaja de este tipo de tarjeta radica que aunque el número caiga en manos de usuarios inescrupulosos ya no es valido y por lo tanto no pueden realizar ninguna transacción con dicho número.

5.2 Cheques electrónicos

Esta forma de pago funciona igual que los cheques de papel, cuando el comprador desea comprar un producto en Internet envía un E-mail con un

cheque electrónico adjunto. El cheque está firmado digitalmente y encriptado. El vendedor lo recibe, lo endosa con su firma digital, luego lo envía por E-mail junto con la boleta de depósito a su banco. El banco verifica la autenticidad, tanto del cheque como del endoso, luego acredita la cuenta del vendedor del cheque.

5.3 Dinero digital

El dinero virtual se adquiere previamente en con una entidad financiera con la que se mantiene una cuenta, esta entidad envía la cantidad de dinero que se establezca al ordenador del comprador, que a partir de ese momento puede utilizarlo. Esto funciona de la siguiente manera:

1 El consumidor solicita una cantidad que se carga a su tarjeta de crédito

2 El dinero es transmitido electrónicamente a la computadora del consumidor junto con el software para manejarlo.

3 Cuando esa persona compra algo en la *Web* de una compañía o comercio que acepta dinero electrónico, llena un formulario con los datos del pago.

4. En la computadora del consumidor, el software debita el monto de la compra.

El empleo del dinero digital en las transacciones, requiere que el servidor del vendedor y comprador, tengan instalados los respectivos programas de cobro y pago provistos por la empresa que brinda el servicio.

5.4 Ejemplo de mecanismos alternos al pago con tarjeta de crédito

5.4.1 *Saf-T-Pay*

Es un mecanismo en el que el comprador puede pagar sus adquisiciones en línea y cargar el monto a una cuenta bancaria.

Este sistema trabaja directamente con el banco, son los bancos los que realizan los pagos a los vendedores, funciona de la siguiente manera los

bancos ofrecerán directamente el sistema a sus clientes sin necesidad de un nuevo instrumento o una clave. Una vez que el cliente elige una compra, *Saf-T-Pay* le facilita la cantidad en moneda local para eliminar riesgos de tipo de cambio. Después, el cliente recibe la mercancía y la paga a la tienda electrónicamente por medio de su banco.

Los compradores podrán acceder a tiendas en todo el mundo debitadas en la moneda de la cuenta, y las transacciones totalmente confidenciales sin revelar en ningún momento los datos financieros del comprador.

5.4.2 *Pay-pal*

Es un sistema que permite a cualquier persona que tenga una dirección de e-mail enviar o recibir dinero en línea utilizando su tarjeta de crédito de manera totalmente segura.

La información confidencial es cifrada mientras se mueve de la computadora del comprador a las computadoras de *PayPal*. El protocolo de cifrado que *PayPal* usa es *SSL*. Cuando el comprador envía o solicita dinero a

través de *PayPal*, la única información que ve el destinatario son su dirección de e-mail, la fecha de registro ("*sign-up*") y si se ha confirmado alguna cuenta en otra institución financiera. Los destinatarios nunca verán su información financiera, como por ejemplo números de cuentas bancarias o de tarjetas de crédito.

Los compradores pueden enviar un pago a cualquier persona o comercio registrada en *PayPal*, con sólo escribir una cantidad en dólares en un formulario en línea. Cuando se envía el pago, *PayPal* cobra a la tarjeta de crédito del comprador la cantidad especificada, si la transacción es autorizada le envía el dinero al comercio registrado. El pago no toma más de cinco minutos, No es necesario que la persona que envía el dinero se inscriba en *PayPal*. El comprador llena un formulario de pago electrónico para enviar el dinero al comercio registrado donde realiza la adquisición de algún servicio o producto. El comprador y el comercio reciben un correo electrónico con los detalles del pago realizado.

5.4.3 *eCash*

Es un sistema de pago desarrollado por la compañía *DigiCash* que permite la transferencia de dinero electrónico de un comprador a un vendedor por medio de un sistema de red online, para poder realizar pagos por medio de este sistema tanto el comprador como el vendedor deben poseer una cuenta en cualquier de los bancos que emiten el dinero electrónico *eCash*.

Con este sistema el comprador debe solicitar al banco un cupón criptográficos que representan el dinero, el banco debe verificar la identidad del cliente, debita la cantidad de su cuenta y envía un cupón que contiene un número de serie que no es mas que el valor de la moneda multiplicado por un factor ciego.

El vendedor recibe el dinero *eCash* y envía los datos de este al banco para que verifique la autenticidad de este y que no haya sido gastado con anterioridad o sea dinero duplicado.

Un rasgo fundamental de *eCash* es que preserva el anonimato de la persona que pago con él, gracias a una técnica criptográfica conocida como firma digital ciega. Sin embargo este sistema presenta algunas desventajas, como el crecimiento del volumen de la base de datos que maneja el banco donde se guardan los números de serie correspondiente a cada moneda.

Otra desventaja que presenta este sistema de pago es que el comprador solamente puede realizar pagos con *eCash* en comercios que tengan cuenta en los bancos que prestan este servicio, ya que las monedas emitidas por el banco no son aceptadas en otros bancos y el número de comercios afiliados al sistema es muy poco comparado con el mercado global del comercio electrónico.

5.4.4 *Virtu@lCash*

El sistema *Virtu@lCash* consiste en una tarjeta que le permite al comprador adquirir bienes o servicios ofrecidos en las tiendas virtuales y que estén afiliadas a este servicio.

El comprador debe solicitar la tarjeta al banco que ofrece el servicio indicando el número de cuenta al cual debitaran los pagos efectuados, el banco le proporcionara además de la tarjeta un *pin* (clave secreta) que deberá proporcionar cada vez que realiza una compra junto con el número de tarjeta, estos datos son enviados a través de un servidor seguro del propio banco donde se verificara su autenticidad y debitar de la cuenta que posee el comprador en dicho banco, realizado este procedimiento el banco notifica al vendedor para que autorice la venta y envié el producto.

Una de las ventajas que ofrece este sistema es que conserva de cierta manera el anonimato del comprador protegiendo la información personal de este, ya que estos datos nunca son accesibles al vendedor, la desventaja principal de este sistema es que se debe acceder a un servidor central para verificar los datos, como lo hemos visto anteriormente, este procedimiento aumenta el costo por transacción y la hace mas lenta.

CONCLUSIONES

1. Existen varias formas para robar números de tarjetas de crédito a usuarios del comercio electrónico que van desde las más simples e ingenuas como preguntarle directamente a alguien la información hasta las más creativas y de última tecnología como los sistemas de administración remota que ponen a disposición de los ladrones la computadora en su totalidad; la mayoría de estos métodos poseen la característica que es el mismo usuario el que permite la obtención de la información sin darse cuenta de ello.

2. El comercio electrónico es una tecnología que facilita la realización de transacciones comerciales sin que las partes involucradas tengan ningún contacto físico sustituyendo así la forma tradicional, aunque existen muchas maneras de proteger la información que se intercambia no deja de tener fallos que no lo hacen al 100% seguro, esto hace que un usuario esté expuesto a una serie de ataques que pueden dejar vulnerable el sistema, si éste no está protegido.

3. El pago con tarjeta de crédito en el comercio electrónico es fácil, el costo es bajo y el tiempo de ejecución es relativamente rápido, pero tiene el inconveniente de que no es 100% seguro, sin embargo, existen alternativas para este tipo de pago, algunas de ellas son más seguras pero tienen un costo más alto y el tiempo de ejecución es mayor, otras son mas baratas y rápidas pero la seguridad que brindan es muy baja.

4. La informática es una ciencia de constantes cambios, aunque día a día existen nuevas formas de asegurar la información de un sistema, también se desarrollan nuevas formas de penetrar esa seguridad.

5. El comercio electrónico puede ser una herramienta muy eficaz tanto para el vendedor como para el comprador, el elemento que no lo ha dejado despegar en nuestro país es la falta de conocimiento de esta herramienta, la inseguridad y el desconocimiento de los usuarios a las técnicas más básicas de seguridad, es necesario que no sólo para el comercio electrónico sino para cualquier tipo de operación que se desee realizar a través de Internet, que los usuarios sepan cómo proteger sus máquinas y su información.

6. Una empresa puede poseer un sistema de seguridad que es impenetrable tecnológicamente, que ha tomado todas las consideraciones tecnológicas para mantener la seguridad e integridad de la información pero por un descuido humano o con mala intención de un empleado puede ver violada la seguridad, esto hace pensar que uno de los factores más importantes a tomar en cuenta, en lo que a seguridad respecta, es el factor humano.

7. A pesar de los ataques a que se puede llegar en una transacción comercial si se siguen las reglas para mantener una máquina actualizada, en cuanto a corrección de errores, a vacunas, liberando la máquina de programas espías y constatando que el vendedor cuenta con los certificados de autenticidad necesaria, la experiencia de comprar a través de Internet puede ser muy útil y beneficiosa.

RECOMENDACIONES

1. Debido a la inseguridad que brinda el comercio electrónico es necesario estar actualizado respecto de los nuevos tipos de ataques a los que un usuario puede estar expuesto, teniendo siempre en cuenta que hoy se creó una nueva forma de proteger la información del sistema y mañana se crea una nueva forma de penetrar dicho sistema y robar la información.

2. El constante ataque al que un usuario está expuesto hace necesario vigilar siempre el sistema, actualizando las vacunas, utilizando detectores de programas que pueden llegar a tomar información, se puede incrementar el nivel de seguridad en un sistema.

3. De nada sirve poseer el mejor sistema de seguridad, poseer la tecnología de comunicación más segura si un empleado u otro usuario puede revelar la clave de acceso al sistema en una plática con un supuesto amigo, visitar páginas sin servidores seguros o bajar programas gratuitos que pueden revelar la información, es necesario informar a los usuarios del sistema

sobre los tipos de ataques que pueden sufrir y las políticas de seguridad que se están llevando a cabo para evitar un ataque por error humano.

4. Desconfiar de todo y de todos puede ser una forma de asegurar la información en un sistema, tener en cuenta para qué va a querer un amigo el número de tarjeta de crédito o quién puede dar acceso gratis a un programa que hace maravillas sin ningún interés, no revelar información importante en ningún lugar o de ninguna forma puede lograr evitar ser víctima de un ataque en Internet.

BIBLIOGRAFIA

1. McClure Stuart, Scambray Joel y Kurst George. Hacking Exposed Network Security Secrets and Solutions. Estados Unidos: Osborne / McGraw-Hill. 1999

2. Cole, Eric. Hackers Beware: The Ultimate Guide to Network Security. 1st edition. Estados Unidos: New Riders Publishing . 2001

3. Northcutt, Stephen; McLachlan, Donald; Novak, Judy. Network Intrusion Detection: An Analyst's Handbook. 2nd Edition. Estados Unidos: New Riders Publishing.

Páginas de Internet visitadas

4. http://webservices.xml.com/pub/a/ws/2003/04/29/transactions.html: 26/04/2005

5. http://www-128.ibm.com/developerworks/library/specification/ws-tx: 26/04/2005

6. http://www.oracle.com/technology/oramag/oracle/03-nov/o63dev_web.html: 26/04/2005

7. http://www-106.ibm.com/developerworks/webservices/library/ws-tranart/ 26/04/2005

8. http://dev2dev.bea.com/pub/a/2004/01/ws-transaction.html: 26/04/2005

9. http://msdn.microsoft.com/library/default.asp?url=/library/enus/dnglobspec/html/ws-transaction.asp: 26/04/2005

10. http://www.oasisopen.org/committees/download.php/2077/BTP_Primer_v1.0.20020605.pdf: 03/05/2005

11. http://msdn.microsoft.com/library/default.asp?url=/library/enus/dnglobspec/html/wsspecsover.asp: 03/05/2005

12. http://www.microsoft.com/spanish/msdn/articulos/archivo/141103/voices/wssecdrill.asp: 25/04/2005

13. http://glud.udistrital.edu.co/glud/areas/doc/articulos/1_articulo_ws/serviciosweb.html: 18/04/2005

14. http://www.gsi.dit.upm.es/~cif/cursos/ssii/agmov4.pdf: 19/04/2005

15. www.inf.utfsm.cl/~rmonge/vespertino/apunte05-1.pdf : 19/04/2005

16. http://www.malditainternet.com/index.php/section=article/sid=58: 19/04/2005

17. http://www.moisesdaniel.com/es/wri/wsepsu.htm#PlataformasEAI: 19/04/2005

18. www.fing.edu.uy/inco/pedeciba/bibliote/reptec/TR0208.pdf: 19/04/2005

19. www.sgp.gov.ar/sitio/foros/rt/docs/interoperabilidad.PDF: 19/04/2005

20. www.urudata.com/Fichas/Esp/Q-flow22vEsp.pdf: 19/04/2005

21. http://www-128.ibm.com/developerworks/webservices/library/ws-secure/: 19/04/2005

22. http://www-128.ibm.com/developerworks/library/ws-secmap/: 19/04/2005

23. http://www-128.ibm.com/developerworks/library/specification/ws-tx/: 19/04/2005

24. http://msdn.microsoft.com/webservices: 02/02/2005

25. http://www.verticalia.com/b2bnoticias/nb2b_verticalia1.htm: 21/04/2005

26. http://revista.robotiker.com/articulos/articulo62/pagina1.jsp: 21/04/2005

27. http://www.w3c.es/divulgacion/guiasbreves/Seguridad: 21/04/2005

28. http://www.w3c.es/divulgacion/guiasbreves/ServiciosWeb: 21/04/2005

29. http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/Overview: 21/04/2005

30. http://www.w3.org/TR/xkms2/: 21/04/2005

31. http://www.instisec.com/publico/verarticulo.asp?id=70: 21/04/2005

32. http://www.eumed.net/cursecon/ecoinet/seguridad/Seguridad1.ppt: 02/03/2005

33. http://195.235.232.62/empresas/casosywhite/documentacion/whiteWEBSERVICES.pdf: 11/03/2005

34. http://www.sgi.es/prensa/articulos_interes/sic54_sgi.pdf: 11/03/2005

35. http://www.willydev.net/descargas/prev/AseguraWebservices.pdf:11/03/2005

36. http://www.sabretravelnetwork.com/supporting_files/sp2_007395.pdf: 11/03/2005

37. http://www.bijonline.com/Article.asp?ArticleID=1005&DepartmentId=9:
11/03/2005

38. http://www.xwss.org/articlesThread.jsp?forum=34&thread=1599: 11/03/2005

39. www.moisesdaniel.com/es/wri/wsepsu.pdf: 11/03/2005

40. http://www.desarrolloweb.com/articulos/1857.php?manual=61:
13/04/2005

41. http://www.desarrolloweb.com/articulos/1853.php?manual=61:
13/04/2005

42. http://www.desarrolloweb.com/articulos/1883.php?manual=61:
13/04/2005

43. http://www.willydev.net/descargas/prev/AseguraWebservices.pdf:
13/04/2005

44. http://es.wikipedia.org/wiki/Servicio_Web: 13/04/2005

45. http://www.fisica.uson.mx/carlos/WebServices/WSRevolution.htm: 13/04/2005

46. http://msdn.microsoft.com/library/spa/default.asp?url=/library/SPA/cpguide/html/cpcontransactionsupportinaspnetwebservices.asp: 13/04/2005

47. http://revista.robotiker.com/articulos/articulo62/pagina1.jsp: 13/04/2005

48. http://www.sgi.es/prensa/articulos_interes/sic54_sgi.pdf: 13/04/2005

49. http://arcos.inf.uc3m.es/~fgarcia/sd/presentaciones/Arquitectura%20de%20Goblus%20ToolKit%204,%20Seguridad,.ppt: 13/04/2005