INSTITUTO POLITÉCNICO NACIONAL

UNIDAD PROFESIONAL INTERDISCIPLINARIA DE INGENIERÍA Y CIENCIAS SOCIALES Y ADMINISTRATIVAS

MODELO DE SEGURIDAD EN LAS APLICACIONES WEB DESARROLLADAS POR UN TERCERO

T E S I N A

QUE PARA OBTENER EL TÍTULO DE:

I N G E N I E R O E N I N F O R M Á T I C A

P R E S E N T A N :

S A N D R A C A B R E R A G A R C Í A

M A R Í A D E L C A R M E N G A R C Í A C A S T R O

J U A N P A B L O S A L I N A S R O M E R O

QUE PARA OBTENER EL TÍTULO DE:

LICENCIADO EN CIENCIAS DE LA INFORMÁTICA

P R E S E N T A N :

E N R I Q U E M O N T A L V O G O N Z Á L E Z

M I G U E L Á N G E L R O D R Í G U E Z A R C E

MÉXICO D.F. 2009

ÍNDICE

Resumen i
Introducción ii
Capítulo I Marco Metodológico 1

1.1 Planteamiento del Problema 1
1.2 Objetivo 4
1.3 Técnicas e Instrumentos de Medición 4
1.4 Universo y/o Muestra 5
1.5 Justificación 7

Capítulo II Conceptos Básicos 10

2.1 Aplicaciones Web 10
2.1.1 Antecedentes 10
2.1.2 Consideraciones Técnicas 11
2.1.3 Estructura de las Aplicaciones Web 12
2.1.4 Lenguajes de Programación 13
2.2 Servicios Web 13
2.2.1 Comunicación de Servicios Web 14
2.2.2 Seguridad en Webservices 15
2.2.2.1 Autenticación del Cliente de Webservices 15
2.2.2.2 Utilización de SSL 16
2.3 Outsourcing 17
2.3.1 Conceptos y Funcionalidades Básicas 17
2.3.2 Ventajas 18
2.3.3 Desventajas 18
2.4 Seguridad en Informática 19
2.4.1 Ataque 21
2.4.1.1 Clasificación de Ataques 22
2.4.1.1.1 Denegaciones de Servicio 22
2.4.1.1.2 Intrusiones 22
2.4.1.1.3 Ingeniería Social 23
2.4.1.1.4 Puertas Trampa 23
2.4.2 Riesgos en las Aplicaciones Web 23
2.4.2.1 Code Injection 24
2.4.2.1.1 Plataformas Afectadas 24
2.4.2.1.2 Tipos de Inyección de Datos 25
2.4.2.1.2.1 Inyección de Datos Maliciosa 25
2.4.2.1.2.2 Inyección de Datos Benéfica 26
2.4.2.1.2.3 Inyección de Código Inesperada 26
2.4.2.2 Remote Code-Inclusion 26
2.4.2.2.1 Plataformas Afectadas 26
2.4.2.3 SQL Injection 26
2.4.2.3.1 Plataformas Afectadas 27
2.4.2.4 Cross-Site Scripting 27
2.4.2.5 Web Spoofing 28
2.4.2.5.1 DNS Spoofing (Suplantación de Identidad por Nombre de Dominio) 28
2.4.2.5.2 Arp Spoofing (Suplantación de Identidad por Falsificación de Tabla ARP) 29
2.4.2.6 Denegación de Servicios DOS 30
2.4.2.6.1 Plataformas Afectadas 30
2.4.2.7 Usurpación de Privilegios en las Aplicaciones 30
2.4.2.8 Ataques a la Autenticación de Sesión 30

2.4.2.9 Fuerza Bruta 31
2.4.2.9.1 Determinar Vulnerabilidades 31
2.4.2.9.2 Prevención 32
2.4.3 Riesgos en las Aplicaciones Web Generados por la Empresa que Presta el Outsourcing. 32
2.4.3.1 Robo de Información Confidencial 33
2.4.3.1.1 Robo de Código Fuente. 33
2.4.3.1.2 Robo de los Datos de Negocio en BD 34
2.4.3.1.3 Acceso no Autorizado a Otros Sistemas de la Empresa. 34
2.4.3.2 Inserción de Código Malicioso 35
2.4.4 Controles 35
2.4.4.1 Definición 35
2.4.4.2 Estándares y Métodos 36
2.4.5 Autenticación y Autorización 37
2.4.5.1 Objetivo y Definición 37
2.4.5.2 Tipos de Autenticación y Controles 38
2.4.5.3 Recomendaciones Para Autenticación y Autorización 39
2.4.6 Manejo de Sesiones 40
2.4.6.1 Seguridad de Sesiones 41
2.4.6.1.1 Cookies y Sesiones 41
2.4.6.1.2 Cookies de Sesión 42
2.4.6.2 Recomendaciones para Manejo de Sesiones 42
2.4.7 Validación de Datos 43
2.4.7.1 Manejo de Validación de Datos 44
2.5 Principios de Programación Segura 44
2.5.1 Controles 44
2.5.2 Atacantes 45
2.5.3 Bases de la Seguridad de la Información 45
2.5.3.1 Arquitectura de Seguridad 46
2.6 Arquitectura 47
2.6.1 Tipos de Arquitecturas en Java J2EE 48
2.6.2 Tipos de Arquitecturas en .Net 49
2.7 Patrones de Diseño 52
2.7.1 Metodologías de Desarrollo de Software. 53
2.8 Modelo de Riesgo Amenaza 55
2.8.1 Modelado de Riesgo Utilizando el Proceso de Microsoft 55
2.8.2 Sistema de Marcación de Vulnerabilidades Comunes (CVSS) 57
2.8.2.1 Ventajas al Utilizar CVSS 57
2.8.2.2 Desventajas al Utilizar CVSS 58

Capítulo III Principales Riesgos en Aplicaciones Web 59

3.1 Estadísticas de Ataques Relacionadas Con Detección de Intrusos 59
3.1.1 Detección de Riesgos de las Aplicaciones Web. 59
3.1.2 Detección de Riesgos del Lado del Cliente 60
3.1.3 Detección de Riesgos del Lado del Servidor. 62
3.1.4 Detección de Riesgos en el Canal de Comunicación. 64
3.2 Estadísticas de Ataque DDOS 64
3.3 Estadísticas Acerca de Code Injection 68
3.3.1 Inyección de Datos Maliciosa 69
3.3.2 Inyección de Datos Benéfica 69
3.3.3 Inyección de Código Inesperada 69

Capítulo IV Legislación Informática Aplicada al Desarrollo Web 72

4.1 Legislación Informática en México 72
4.2 Delitos Informáticos 72
4.2.1 Intervención de Correo Electrónico 73
4.2.2 Acceso no Autorizado a Sistemas o Servicios 73
4.2.3 Reproducción no Autorizada de Programas Informáticos. 75
4.2.4 Uso de Programas y de Datos Con o Sin Autorización. 79
4.3 Legislación Aplicada al Outsourcing 82
4.4 Mejores Practicas Aplicadas a Terceros 84
4.4.1 ITIL 84
4.4.1.1 Características de ITIL 85
4.4.1.2 Administración o Gestión de Servicios de TI 86
4.4.1.3 Procesos de Gestión de Servicios 86
4.4.2 COBIT 87
4.4.2.1 Planificación y Organización 88
4.4.2.2 Adquisición e Implementación 89
4.4.2.3 Entrega y Soporte 90
4.4.2.4 Supervisión y Evaluación 91
4.5 ISO 27001 91
4.5.1 Beneficios al Implantar la Norma 93
4.5.2 Norma ISO 27001 94

Capítulo V Modelo de Seguridad en las Aplicaciones Web Desarrolladas por un Tercero 97

5.1 Modelo de Seguridad en Aplicaciones Web 97
5.1.1 Capa de Presentación 97
5.1.1.1 Técnicas y Herramientas de Seguridad 97
5.1.1.1.1 Uso Adecuado de Frameworks de Javascript(AJAX) 97
5.1.1.1.2 Validación Antisamy 98
5.1.1.1.3 Componente de Encriptación CRYTTR 98
5.1.1.1.4 Anti Tampering 98
5.1.2 Capa de Negocio 98
5.1.3 Técnicas y Herramientas de Seguridad 99
5.1.3.1.1 Limitación de Intentos de Logueo 99
5.1.3.1.2 Componentes de Validación Aplicados al Negocio. 99
5.1.3.1.3 Eliminación de Ciclos Innecesarios. 99
5.1.3.1.4 Componente de Log de Sucesos 99
5.1.3.1.5 Componentes de Transaccionabilidad 99
5.1.3.1.6 Componentes SQL de Tipo "Bind Variable" 100
5.1.4 Capa de Datos 100
5.1.5 Capa de Servicios Web. 100
5.1.5.1 Técnicas y Herramientas de Seguridad 100
5.1.5.1.1 Credenciales de Autenticación 100
5.1.5.1.2 Limitación de Consumo 100
5.1.6 Capa de Arquitectura 101
5.1.6.1 Técnicas y Herramientas de Seguridad 101
5.1.6.1.1 Estructura de Balanceo de Carga 101
5.1.6.1.2 Componentes ORM 101
5.1.6.1.3 Servidor de Cache Apache 101
5.1.6.1.4 Componentes Firewall de Aplicación Modsecurity 101
5.1.6.1.5 Implementación de SSL(HTTPS) 102

Capítulo VI Aplicación del Modelo de Seguridad 103
Generalidades de la Empresa 103
6.1 Aplicación del Modelo 103
6.2 Resultados 108

6.3 Ventajas 109
6.4 Desventajas 110
6.5 Conclusiones 111
Conclusiones 112
Bibliografía 114
Glosario 119

RESUMEN

El desarrollo del presente trabajo, constituye la aplicación de un modelo de seguridad en las aplicaciones Web desarrolladas por un tercero, debido a que en México existen muchas organizaciones que actualmente hacen uso de aplicaciones Web para el manejo de su información y la mayoría de las veces, estos portales son desarrollados por un tercero, es decir, contratación de Outsourcing.

Muchas organizaciones que contratan el servicio de Outsourcing no tienen bien establecidos sus lineamientos y políticas ante la adquisición de este servicio; de igual forma las organizaciones que prestan el servicio no poseen lineamientos que certifiquen que el servicio que brindan es 100% ético. El propósito del modelo de seguridad en las aplicaciones Web desarrolladas por un tercero, es proporcionar un respaldo hacia las empresas que contratan el Outsourcing con el fin de proteger la información que se encuentra en juego durante el desarrollo de la aplicación. El modelo se concentra en una serie de lineamientos y políticas basados en las leyes mexicanas como la Ley Federal de Derechos de Autor, la Ley Federal del Trabajo y la Constitución Política de los Estados Unidos Mexicanos; incluyendo también las mejores prácticas como ITIL y COBIT.

El modelo se aplicó en la empresa Dominion, esto se decidió después de haber evaluado diez diferentes organizaciones y tomando en cuenta que Dominion es una de las principales organizaciones que brinda servicios de Outsourcing. Durante la aplicación del modelo se determinaron diferentes técnicas para la implementación de aplicaciones Web, como los son el uso de Framework de JavaScript, Componentes de Encriptación, Componentes de SQL, Credenciales de Autenticación e Implementación de SSL. En la aplicación de estas diferentes técnicas se logró observar que al desarrollar aplicaciones Web, el Outsourcing lo hace bajo un plan de trabajo con fechas establecidas, el cual la mayoría de las veces no contempla los Bugs o problemas que se pudieran desatar durante la implementación, por ello cuando el desarrollador se topa con esta situación, no tiene mayor opción que el entregar la aplicación en tiempo aun y con errores.

De esta manera el objetivo planteado fue concluido satisfactoriamente, quedando comprobado que con la aplicación de un Modelo de Seguridad en las Aplicaciones Web se disminuyen los riesgos que implica el contratar el servicio de Outsourcing para la implementación de estas; así mismo se considera la probabilidad de que pudiese existir un Modelo de Seguridad que refuerce la contraparte en las implementaciones, es decir, un modelo que establezca los lineamientos y políticas en el lado del prestamista del servicio u Outsourcing.

INTRODUCCIÓN

Hoy en día la mayoría de las aplicaciones se ejecutan bajo un ambiente Web, en estas se manejan distintos tipos de información como lo es financiera o bien de uso general; las empresas han adoptado este medio para tener acceso inmediato a sus recursos informativos. Entre los recursos informativos se encuentran las bases de datos propias del giro de la empresa, así como también información relacionada con los clientes o proveedores de la misma, esto hace que este portal se vuelva atractivo a cualquier usuario mal intencionado. Por ello en este trabajo se desarrolla un modelo de seguridad orientado al desarrollo de aplicaciones Web, las cuales se vuelven vulnerables al momento de ser publicadas en la red; aplicando el modelo de seguridad se reduce la vulnerabilidad de las aplicaciones desde el momento en que se comienzan a implementar, ya que el modelo de seguridad regula los niveles de acceso y que debe tener cada uno de los usuarios que interactúan con el sistema. La seguridad en las aplicaciones Web se presenta desde el momento en el que un usuario se firma a esta, ya que puede sufrir de robo de identidad o bien de pérdida de conexión entre la base de datos y la aplicación; en las aplicaciones Web desarrolladas por un tercero además de correr con los riesgos antes mencionados, también existe la probabilidad de que desarrollador hurte información importante para la organización.

El objetivo de esta tesina es disminuir los riesgos que se presentan en una aplicación Web desarrollada por un tercero mejor conocido como Outsourcing, mediante la elaboración de un modelo de seguridad que enfrente a las principales vulnerabilidades que se encuentran en un entorno Web tomando en cuenta la mayoría de las técnica de ataque conocidas a nivel mundial, y como resultado la elaboración de las mejores prácticas con la finalidad de tener las bases para generar una aplicación estable. El modelo se aplicó en la empresa Dominion ya que esta se dedica a desarrollar aplicaciones Web. Dominio brinda personal externo en el desarrollo de sus sistemas, dicho personal tiene que seguir normas y reglamentos definidos por cada uno de los clientes. La aplicación de un modelo de seguridad eleva la seguridad del software desarrollado.

Para lograr el objetivo de esta tesina se implementan los siguientes capítulos:

En el Capítulo I nombrado Marco Metodológico se plantean las principales problemáticas de las inseguridades de las aplicaciones Web, así mismo los problemas que representa el contratar a una empresa de Outsourcing como proveedor del sistema.

Para el Capítulo II Conceptos básicos, se definen algunos de los conceptos básicos que se utilizan a lo largo de la tesina. La finalidad de este capítulo es que el lector entienda los conceptos que están relacionados con las aplicaciones Web y la auditoría informática. No se pretende entrar a profundidad en alguna tecnología de programación en particular, puesto que la auditoría informática puede ser aplicada a cualquier lenguaje.

En el Capítulo III Principales Riesgos Aplicaciones Web el cual ya es parte del proyecto, se exponen los principales riesgos de seguridad de las aplicaciones Web, con el fin de que estas se vuelvan más confiables y menos vulnerables a los ataques informáticos, se muestran estadísticas sobre los principales riesgos de las aplicaciones Web.

En el Capítulo IV titulado Legislación y Mejores Prácticas se documenta la legislación existente que se refiere a las aplicaciones Web y Outsourcing, así como las mejores prácticas existentes relativas a la seguridad.

El contenido del Capítulo V muestra la estructura del modelo de seguridad implementado, por ello se nombra Modelo de Seguridad en las Aplicaciones Web Desarrolladas por un Tercero; este capítulo describe el uso de técnicas y herramientas para minimizar los riesgos mencionados en los capítulos anteriores y se adapta a cualquier tecnología que nos permita desarrollar una aplicación Web.

El Capítulo VI se titula Aplicación del Modelo de Seguridad, en donde se muestran los resultados de la aplicación del Modelo de Seguridad en las Aplicaciones Web Desarrolladas por un Tercero, el cual se implementó en la empresa Dominion, dichos resultados están basados en las pruebas comparativas que se realizaron al aplicar el modelo.

De acuerdo a la elaboración de esta tesina y con la implementación del modelo de seguridad en la empresa Dominion, se concluye que al utilizar un modelo de seguridad en los desarrollos de sistemas se reduce sustancialmente vulnerabilidades y riesgos de las aplicaciones Web, así mismo aumenta la seguridad, integridad y disponibilidad de la información utilizada por la empresa que solicita servicios de Outsourcing.

CAPÍTULO I MARCO METODOLÓGICO

A lo largo de este capítulo se plantean algunos motivos por los cuales las empresas deciden contratar a un tercero (Outsourcing) siendo uno de los principales el disminuir los costos de capacitación y al mismo tiempo disminuir los riesgos que implica el desarrollo de una aplicación en ambiente Web, estos riesgos suelen ir desde que el sistema no se implemente de acuerdo a las necesidades de la empresa hasta que este enfrente un gran número de vulnerabilidades al entrar en producción. Mediante la elaboración de un modelo de seguridad aplicado al desarrollo se enfrentan las diferentes vulnerabilidades que presenta un sistema al publicarse en la Web, ya que se aplica desde el planteamiento del sistema hasta su implementación, asegurando que el sistema es confiable y sobretodo que cumple con las necesidades establecidas.

CAPÍTULO I MARCO METODOLÓGICO

1.1 PLANTEAMIENTO DEL PROBLEMA

En la actualidad un gran número de las aplicaciones son ejecutadas en ambiente Web, esto ocasiona que los sistemas y la información de las empresas sean más vulnerables a los ataques, perjudicando la integridad y disponibilidad de la información; así se pone en riesgo la continuidad del negocio. Todas las aplicaciones son vulnerables a los ataques sin importar la tecnología en la cual fueron desarrollados los sistemas Web.

Recientemente la cantidad de spam y nuevos programas zombies han disminuido pero se ha incrementado los ataques a través de contenido Web malicioso y ataques de seguridad combinados tanto a empresas como a usuarios domésticos. Los ataques a los que se enfrentan las compañías, hoy en día, se están volviendo una variedad que afecta tanto al correo electrónico como a la seguridad Web.

De acuerdo con la Figura 1.1 las empresas ponen en riesgo los siguientes puntos de su base de datos al tener sus aplicaciones en sistemas Web.

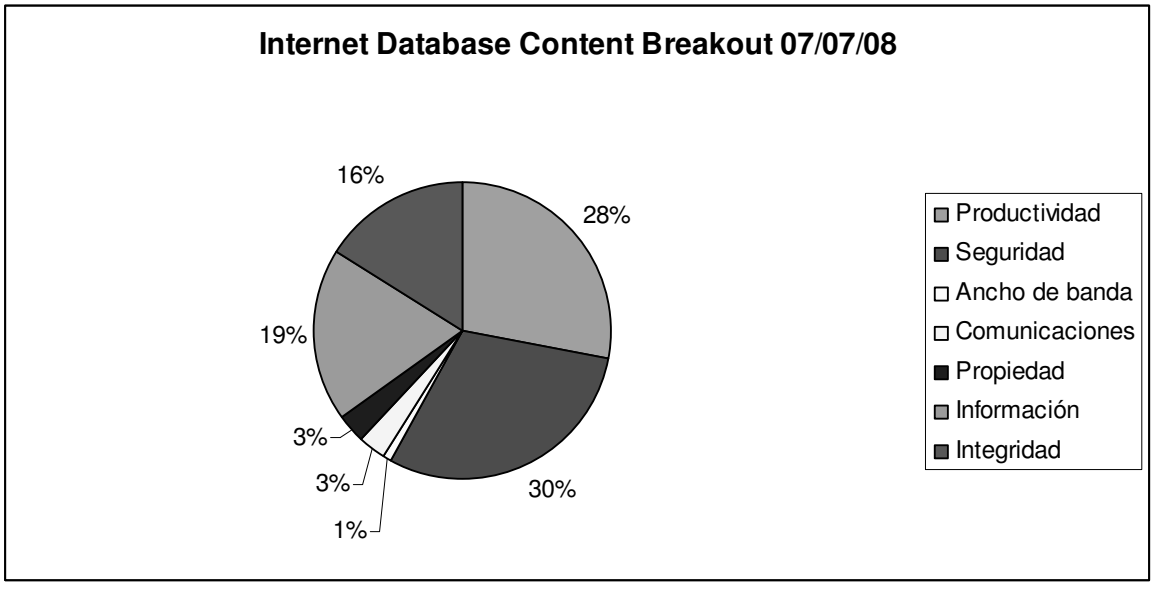

Figura 1.1 Principales Riesgos de las Bases de Datos[1]

[1] Secure Computing Corporation. Secure Computing Internet Threats Report, 2008, http://www.securecomputing.com/

Como se puede observar en la Figura anterior, las empresas pueden perder productividad e información si son víctimas de un ataque a sus sistemas, como bien se sabe la información que se maneja en los sistemas es de vital importancia para el funcionamiento del negocio por lo que se pone en juego la continuidad de este si un ataque tiene éxito.

El uso de un modelo de seguridad a las aplicaciones Web, disminuye los riesgos de pérdida de productividad de los sistemas ya que es inevitable que las empresas usen aplicaciones Web debido a la creciente competitividad que el mundo moderno exige a las empresas y a la globalización y distribución de procesos de las mismas.

Los intentos de encontrar vulnerabilidades a las aplicaciones Web por parte de atacantes se ha incrementado constantemente, es por eso que las aplicaciones deben seguir un modelo de seguridad que disminuya dichas vulnerabilidades. Acorde con el departamento de Internet Security Systems en IBM México para el año 2008 los intentos por encontrar vulnerabilidades alcanzo la cifra de 8000.

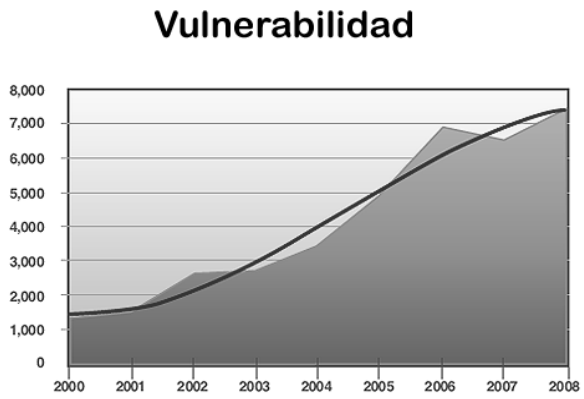

Figura 1.2 Intentos de Vulnerabilidad 2000 - 2008[2]

Las empresas suelen contratar a terceros para desarrollar las aplicaciones Web ya que esto implica algunos beneficios como delegar las funciones y responsabilidades, mejor planeación de tiempos y costos, a continuación algunos puntos por lo que las empresas contratan a terceros para desarrollar las aplicaciones Web:

- Mejor planeación de costos. Se cuenta con un costo desde antes del inicio del proyecto, este costo es propuesto por el tercero con base a los cálculos que desarrollo en cuanto al tiempo y complejidad de la aflicción a desarrollar.

[2] IBM Global Technology Services. IBM Internet Security Systems X-Force® 2008 Trend & Risk Report, 2009, http://www-935.ibm.com/services/us/iss/xforce/trendreports/xforce-2008-annual-report.pdf pag 18.

- Menor costo de licencias. Puesto que las aplicaciones son desarrolladas por un tercero, la empresa contratante no tiene que realizar un gasto por concepto de licencias de Software de desarrollo, esto debido a que no es necesario que las aplicaciones sean desarrolladas en sus instalaciones.
- Menor costo de personal. Las empresas desarrolladoras de software hacen uso de sus recursos acorde a las necesidades de los proyectos, con base a esto es posible que un consultor este asignado a más de un proyecto. Los gastos y administración del personal son delegados al tercero que se encarga de desarrollar la aplicación Web.
- Mejor planeación de entrega. Las empresas contratantes y la encargada del desarrollo suelen definir tiempos de entrega y revisión antes de iniciar el proyecto de desarrollo de aplicación Web, puesto que estos tiempos son definidos desde el inicio la empresa contrátate puede definir penalizaciones al incumplimiento de las fechas pactadas, como resultado la empresa encargada del desarrollo se ve obligada a cumplir con lo acordado en el contrato.
- Menor costo de capacitación. Puesto que un tercero se encarga de desarrollar la aplicación, no es necesario que la empresa contratante invierta en la capacitación de su personal en la tecnología de desarrollo, por consiguiente esto disminuye los costos.[3]

Así como hay ventajas de hacer uso de un tercero en el desarrollo de aplicaciones, también existen riesgos que ponen en peligro el proyecto de desarrollo de aplicaciones Web.

- Inadecuada selección del desarrollador de la aplicación
- Contrato no adecuado
- Incumplimiento del tercero una vez iniciado el proyecto
- Falta de control sobre el tercero

Al hacer uso de un modelo de seguridad en las aplicaciones Web desarrolladas por un tercero se pueden disminuir las vulnerabilidades de los sistemas, obteniendo como resultado una aplicación Web menos propensa a recibir ataques, lo cual permite una disminución en los costos derivados de los ataques informáticos a los sistemas de la empresa; así mismo el modelo es útil para que las empresas que desarrollan los sistemas o los empleados de estas no hagan mal uso de la información o código de las aplicaciones.

[3] KPMG, Las Tendencias del Mercado moderno "Outsourcing", http://www.kpmg.cl/documentos/Final_Presentacion_BPO_July_2004.pdf, 2004.

1.2 OBJETIVO

El objetivo de esta tesis es disminuir los riesgos que se presentan en una aplicación Web desarrollada por un tercero mejor conocido como Outsourcing, mediante la elaboración de un modelo de seguridad que enfrente a las principales vulnerabilidades que se encuentran en un entorno Web tomando en cuenta la mayoría de las técnica de ataque conocidas a nivel mundial, y como resultado la elaboración de las mejores prácticas con la finalidad de tener las bases para generar una aplicación estable.

Al trabajar con un modelo de seguridad se obtiene una mayor perspectiva de certidumbre y confianza hacia la empresa que solicita el servicio de Outsourcing debido a que este prevé los puntos débiles existentes en un sistema Web.

Como objetivos específicos de este trabajo se presentan los siguientes:

- Implementación de técnicas para el desarrollo de un código seguro tomando en cuenta el conocimiento y experiencia por parte de la empresa que desarrolla la aplicación.
- Métodos de autenticación, validación de datos y manejo de sesiones, todo esto dependiendo de un análisis de los perfiles de acuerdo a la jerarquía de los usuarios dentro de la empresa así como también un buen diseño en base de datos que permita limitar accesos.
- Las principales técnicas de amenaza, ejemplos y mejores prácticas para disminuir las probabilidades de ser afectados, se toman en cuenta los principales ataques más comunes a nivel mundial explicación a detalle para comprender como trabajan.
- Estrategias para el manejo de errores así como la utilización de bitácoras para almacenar eventos encontrados dentro del sistema con el fin de dar seguimiento y solución a las futuras amenazas o inconsistencias detectadas en la aplicación.

1.3 TÉCNICAS E INSTRUMENTOS DE MEDICIÓN

En el articulo “Tipos de Investigación” se menciona que “la planeación de una buena metodología permite establecer los hechos y las relaciones con los resultados obtenidos. Científicamente la metodología es un procedimiento general para lograr de una manera precisa el objetivo de la

investigación. De ahí, que la metodología en la investigación presenta los métodos y técnicas para la investigación"[4]

Por lo anterior se concluye que es de suma importancia la elección adecuada del tipo de investigación a realizar, puesto que cada una cuenta con técnicas diferentes. Por ejemplo el objetivo de la investigación explicativa es dar a conocer la realidad, establecer y tratar de contestar las preguntas tales como: cuándo, cómo, dónde y por qué ocurre el fenómeno.
Tomás Austin define que "los estudios explicativos van más allá de la descripción de conceptos o fenómenos o del establecimiento de relaciones entre conceptos; están dirigidos a responder a las causas de los eventos físicos o sociales. Como su nombre lo índica, su interés se centra en explicar por qué ocurre un fenómeno y en qué condiciones se da éste, o por qué dos o más variables están relacionadas".[5]

Tomando como base la definición de Tomás Austin se puede decir que los estudios exploratorios se efectúan cuando existe muy poca información del problema a investigar, su metodología suele ser más flexible a comparación con los estudios descriptivos y uno de los objetivos principales de esta técnica de investigación es el proporcionar una cierta familiaridad con el fenómeno que se está investigando y de esta manera realizar una investigación más extensa sobre un problema de la vida real.

Mediante el presente trabajo se dan respuestas a interrogantes tales como ¿Cuáles son los principales riesgos de las aplicaciones Web?, ¿Existe una legislación que regule los ataques que sufre una aplicación Web? ¿Por qué implementar un modelo de seguridad a las aplicaciones Web desarrolladas por un tercero? Para dar respuesta a estas preguntas se aplicó una serie de encuestas, las cuales están dirigidas a expertos sobre el área que trabajen en consultorías.

1.4 UNIVERSO Y/O MUESTRA

El universo objeto de la presente investigación estuvo constituido por una encuesta a profesionales expertos y se entrevistaron a 10 socios de empresas consultoras grandes, tres socios de consultoras medianas, y 3 socios de Consultoras Pequeñas, todas ellas existentes en la Zona Metropolitana del Distrito Federal, con número de empleados entre 17 a 80, los cuales se

[4] Medina Carrera Matilde Carolina, Tipos de Investigación. 2009. http://www.monografias.com/trabajos59/tipos-investigacion/tipos-investigacion.shtml

[5] Tomás Austin, Diseño de la Investigación. 2009. http://www.angelfire.com/emo/tomaustin/Met/guiacuatrodise_o.htm

encuentran laborando asignados con los distintos clientes, y ninguno trabajando directamente en las oficinas físicas de las consultoras.[6]

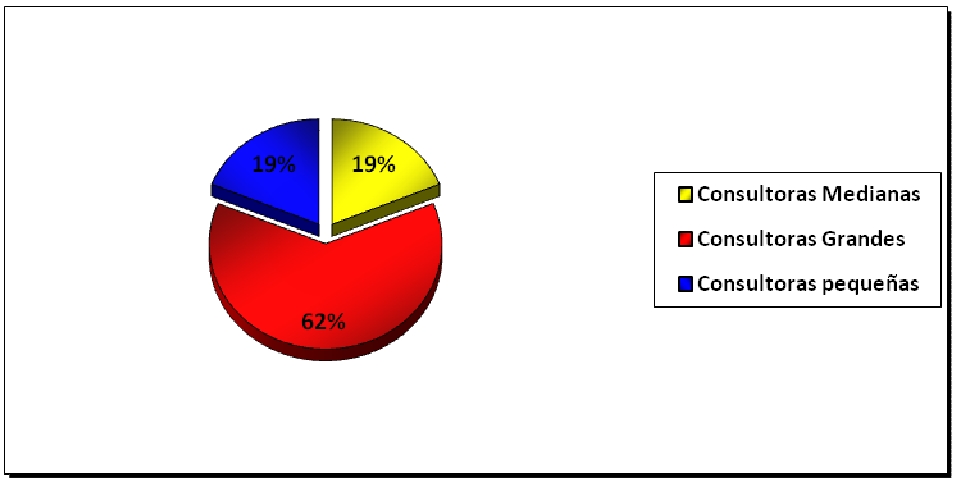

Figura 1.3. Universo de la Población

Se revisaron los perfiles de las consultoras para comparar con las necesidades declaradas por los auditores de estados financieros. La muestra estuvo conformada por 7 Empresas Consultoras, de las cuales se tomó una muestra de 3 de 10 Socios de Consultoras Grandes (30%), 2 de 3 de Consultoras medianas (66.6%) y 1 de las 3 Consultoras pequeñas (33%).

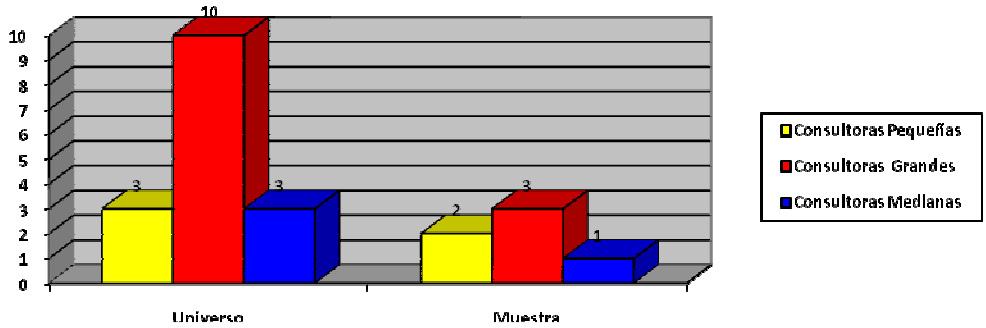

Figura 1.4. Muestra de la Población.

Esta muestra considera la totalidad de las categorías existentes para definir el tamaño de una Consultora en Informática, la cual es el objeto de estudio, las instituciones fueron elegidas de forma arbitraria, ya que, toda la población reunió los requisitos de infraestructura para ser integrados al modelo de Investigación.

[6] Encuesta a Profesionales Expertos (2009)

1.5 JUSTIFICACIÓN

La mayoría de las aplicaciones e información general incluso hasta financiera y en algunos casos estratégica se encuentra en aplicaciones de ambiente Web. Por su accesibilidad en cualquier momento y en cualquier lugar sin estar preponderantemente en un lugar de trabajo específico, sin embargo, la accesibilidad y el manejo de la información deben estar controlados y tener medidas de seguridad adecuadas.

Es bien sabido que las aplicaciones Web y los ambientes Web son propensos y susceptibles a los posibles ataques, el mal uso o incluso el robo de la información, que pudiese abarcar desde el uso estratégico de la información financiera de las empresas competidoras o hasta la intervención de informaciones gubernamentales. Por lo tanto es necesario la implementación de controles internos específicos y detallados, así como el establecimiento de medidas de seguridad adecuadas para cada tipo, tamaño y necesidad de una organización.

Los modelos de seguridad en las aplicaciones Web proporcionan una visión independiente sobre las vulnerabilidades, exposiciones, el nivel de diseño de controles y los riesgos, así como también contribuyen a disminuir los costos de propiedad de los sistemas, puesto que al reducir las vulnerabilidades se reducen los gastos en solucionar los posibles impactos que un ataque pueda ocasionar.

El modelo de seguridad puede abarcar cualquier nivel de información deseable, existen organizaciones en las que se expone información para el público en general, referente a productos o promociones pero el acceso al portal informático se va limitando de acuerdo a la información que se desee ir presentando, tales como niveles de ventas, estándares, flujo gramas e información de uso contable o tecnologías de la información.

El acceso a la información se va dando de acuerdo a los permisos otorgados y a la importancia del usuario así como a la necesidad de información requerida por este. Por lo tanto la seguridad se incrementa de acuerdo a la clasificación de la importancia de esta y solo los usuarios con un nivel de importancia preponderante en un organigrama tienen un espectro más amplio de accesibilidad de la información.

El nivel de vulnerabilidad de un ambiente Web puede representar un riesgo que afecte a toda una organización e incluso a una determinada zona Geográfica, el mismo uso y exposición de la información en portales informáticos, el número de usuarios y por supuesto el nivel de modelos de

seguridad y su eficacia o bien su vulnerabilidad. Marca el nivel de ataques Web en las diferentes naciones.

A continuación se presenta el Cuadro Estadístico (Figura 1.5) sobre ataques informáticos clasificado por país, como se puede observar México figura entre los diez primeros países.[7]

Lugar	País	Cantidad de ataques	Porcentaje del total de ataques
1	China	12.708.285	53,665%
2	Egipto	3.615.355	15,267%
3	Turquía	709.499	2,996%
4	India	479.429	2,025%
5	Estados Unidos	416.437	1,759%
6	Vietnam	346.602	1,464%
7	Rusia	335.656	1,417%
8	México	308.399	1,302%
9	Arabia Saudita	287.300	1,213%
10	Alemania	253.097	1,069%
11	Marruecos	230.199	0,972%
12	Tailandia	204.417	0,863%
13	Indonesia	190.607	0,805%
14	Reino Unido	188.908	0,798%
15	Francia	182.975	0,773%
16	Siria	134.601	0,568%
17	Brasil	123.736	0,523%
18	Taiwán	122.264	0,516%
19	Italia	121.508	0,513%
20	Israel	118.664	0,501%

Figura 1. 5 Países con más Ataques Informáticos[8]

[7] Gostev Alexander, Las principales estadísticas de 2008, http://www.viruslist.com/sp/analysis?pubid=207271019

Debido a que México figura entre los países con más ataques, en este trabajo se destaca la importancia de implementar un modelo de seguridad en las aplicaciones Web que son desarrolladas por un tercero, ya que en muchos casos lo más valioso para una empresa es la información que se encuentra en los sistemas, y esta debe ser integra, disponible y confidencial.
Con base a los ataques registrados en los diferentes países mencionados en la figura anterior destacan los siguientes diez delitos informáticos aplicados a sistemas desarrollados en ambiente Web, los cuales se representan en la Figura 1.6.[9]

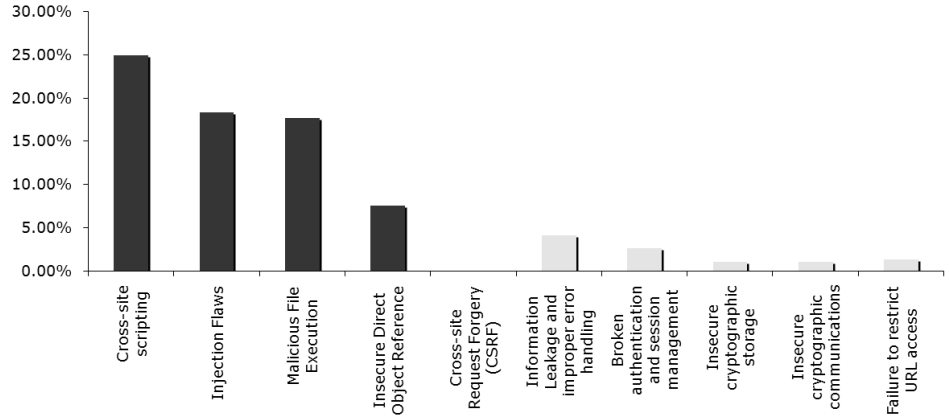

Figura 1.6 Principales Ataques a Aplicaciones Web [10]

Como se puede observar en la Figura anterior, el Cross-Site scripting (alteración de código) es uno de los ataques más recurrentes en las aplicaciones Web.

Actualmente las empresas de Outsourcing poseen desarrollos Web que no tienen un control de seguridad ya que fueron diseñados por personas que no conocían a fondo las mejores prácticas de seguridad enfocadas a desarrollo, por ello en este proyecto se toman en cuenta ciertos controles con el fin de no realizar aplicaciones Web sin considerar los controles básicos de seguridad, ya que estos implican un gasto adicional una vez que se ha implementado la aplicación.

[8] Gostev Alexander, Las principales estadísticas de 2008, http://www.viruslist.com/sp/analysis?pubid=207271019
[9] http://www.owasp.org/
[10] http://www.owasp.org/

CAPÍTULO II CONCEPTOS BÁSICOS

En este capítulo se explican los conceptos básicos para entender un modelo de seguridad para las aplicaciones Web desarrolladas por un tercero; dichos conceptos engloban desde la definición de una aplicación Web, un ataque, que es un tercero o un Outsourcing, entre otros.

Al final de este capítulo se obtiene un mejor panorama de las aplicaciones Web, los posibles ataques y porque es necesario usar un modelo de seguridad; de este modo se contribuye a que las empresas que usan dicho modelo de seguridad puedan disminuir la posibilidad de ataques a sus sistemas, debido a que día a día este tipo de aplicaciones Web tienen más difusión pero a su vez también ha aumentado el número de posibles ataques.

CAPÍTULO II CONCEPTOS BÁSICOS

2.1 APLICACIONES WEB

En Ingeniería de software una aplicación Web es una "aplicación a la cual se tiene acceso vía un navegador Web sobre una red, como Internet o una intranet. Además, es una aplicación de software codificada en un lenguaje soportado por un browser o navegador Web (Como HTML, JavaScript, Java), en la que se confía la ejecución al navegador."[11]

Las aplicaciones Web son populares debido a lo práctico de usar un navegador Web como cliente ligero. La facilidad para actualizar y mantener aplicaciones Web sin distribuir e instalar software a miles de usuarios potenciales es una razón de peso para su popularidad.

Ejemplos bien conocidos de aplicaciones Web son los Webmail, Wiki, Weblog, tiendas en línea y la propia Wikipedia.[12]

Es importante mencionar que una página Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responde a cada una de sus acciones, como por ejemplo; rellenar y enviar formularios, participar en juegos diversos y acceder a gestores de base de datos de todo tipo.

2.1.1 ANTECEDENTES

En los primeros programas de computación que utilizaban una arquitectura cliente-servidor, cada aplicación tenía su propio programa cliente que servía como interfaz de usuario que tenía que ser instalado por separado en cada computador personal de cada usuario. El cliente realiza peticiones a otro programa,-el servidor- , que le daba respuesta. Una mejora en el servidor, como parte de la aplicación, requería normalmente una mejora de los clientes instalados en cada ordenador personal, añadiendo un coste de soporte técnico y disminuyendo la productividad.

En contraste, las aplicaciones Web generan dinámicamente una serie de páginas en un formato estándar como HTML o XHTML que son soportados por los navegadores Web comunes. Para

[11] Pc Magazine, Web Application Definition, http://www.pcmag.com/encyclopedia_term/0,2542,t=Web+application&i=54272,00.asp
[12] Wikipedia, Definition of Web Application, http://en.wikipedia.org/wiki/Web_application

añadir más dinamismo a este proceso, se utilizan lenguajes interpretados en el lado del cliente, tales como JavaScript, que agrega elementos dinámicos a la interfaz de usuario.
Generalmente cada página Web en particular se envía al cliente como un documento estático, pero la secuencia de páginas ofrece al usuario una experiencia interactiva. Durante la sesión, el navegador Web interpreta y muestra en pantalla las páginas, actuando como cliente para cualquier aplicación Web.

A lo largo del tiempo, y con la mayor difusión de navegadores Web en las aplicaciones informáticas de la vida cotidiana, estos han sufrido diversas evoluciones, todas ellas, encaminadas a proporcionar una mayor potencia en la interacción del navegador con el usuario, aplicando la inserción de lenguajes de programación de secuenciado de instrucciones, o scripting.

En 1995, Netscape introdujo un programa de scripting, llamado JavaScript, que permitía a los programadores agregar algunos elementos dinámicos a la interfaz de usuario, que corría del lado del cliente. Hasta entonces, todos los datos tenían que ser mandados al servidor para su procesamiento y los resultados eran entregados al cliente a través de paginas HTML estáticas[13].

En 1996, Macromedia introdujo Flash; un reproductor de animaciones vectoriales que podía ser agregado al navegador Web como un plug-in, o complemento, para incrustar animaciones en las páginas Web. Esto permitía el uso de lenguaje de scripting para programar interacciones en el lado del cliente sin necesidad de comunicarse con el servidor.[14]

En 1999 el concepto “aplicación Web”, fue introducido al lenguaje Java en lo que fue denominado Servlet 2.2. En ese tiempo, ambos, JavaScript y XML ya habían sido lanzados, pero AJAX, aún no había sido concebido, y los objetos XMLHttpRequest habían sido introducidos en Internet Explorer 5.[15]

En 2005, Ajax fue acuñado, y aplicaciones como Gmail empezaron a hacer el lado cliente más y más interactivo. [16]

2.1.2 CONSIDERACIONES TÉCNICAS

[13] Wikipedia, History of Web Applications, http://en.wikipedia.org/wiki/Web_application/History
[14] Ibíd.
[15] Ibíd.
[16] Ibíd.

De acuerdo al sitio Ferran Barba, “una ventaja significativa es que las aplicaciones Web deben funcionar igual, independientemente de la versión del sistema operativo instalado en el cliente.”[17]

En vez de crear clientes para Windows, Mac OS X, GNU/Linux, y otros sistemas operativos, la aplicación Web se escribe una vez y se ejecuta igual en todas partes. Sin embargo, hay aplicaciones inconsistentes escritas con HTML, CSS, DOM y otras especificaciones para navegadores Web que pueden causar problemas en el desarrollo y soporte de las aplicaciones Web.

Adicionalmente, la posibilidad de los usuarios de personalizar muchas de las características de la interfaz (tamaño y color de fuentes, tipos de fuentes, inhabilitar JavaScript) puede interferir con la consistencia de la aplicación Web.

Otra aproximación es utilizar Adobe Flash Player o Java applets para desarrollar parte o toda la interfaz de usuario. Como casi todos los navegadores incluyen soporte para estas tecnologías (usualmente por medio de plug-ins), las aplicaciones basadas en Flash o Java pueden ser implementadas con aproximadamente la misma facilidad. Dado que ignoran las configuraciones de los navegadores, estas tecnologías permiten más control sobre la interfaz, aunque las incompatibilidades entre implementaciones Flash o Java puedan crear nuevas complicaciones.

“Por las similitudes con una arquitectura cliente-servidor, con un cliente no ligero, existen discrepancias sobre el hecho de llamar a estos sistemas aplicaciones Web”; un término alternativo es Aplicación Enriquecida de Internet”.[18]

2.1.3 ESTRUCTURA DE LAS APLICACIONES WEB

Aunque existen muchas variaciones posibles, “una aplicación Web está normalmente estructurada como una aplicación de tres-capas. En su forma más común, el navegador Web ofrece la primera capa y un motor capaz de usar alguna tecnología Web dinámica (PHP, ASP, ASP.NET, CGI, ColdFusion, embPerl, Python (programming language) o Ruby on Rails); que constituye la capa de en medio. Por último, una base de datos constituye la tercera y última capa.

[17] Aplicaciones Web a Medida, http://www.ferranbarba.com/aplicaciones-Web/
[18] Rich, Internet Applications, http://es.wikipedia.org/wiki/Aplicaciones_de_Internet_Ricas

El navegador Web manda peticiones a la capa de en medio que ofrece servicios valiéndose de consultas y actualizaciones a la base de datos y a su vez proporciona una interfaz de usuario"[19].

2.1.4 LENGUAJES DE PROGRAMACIÓN

Existen numerosos lenguajes de programación empleados para el desarrollo de Aplicaciones Web, entre los que destacan:

- PHP
- ASP/ASP.NET
- JAVA, con sus tecnologías Java Servlets y JavaServer Pages (JSP)
- Perl
- HTML
- XML
- Ruby
- Python

De acuerdo a este criterio[20], ASP no es un lenguaje de programación en sí mismo, sino una arquitectura de desarrollo Web en la que se pueden usar por debajo distintos lenguajes (por ejemplo, VB.NET o C# para ASP.NET o VBScript/JScript para ASP).

2.2 SERVICIOS WEB

Es un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios Web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios Web. Para mejorar la interoperabilidad entre distintas implementaciones de servicios Web se ha creado el organismo WS-I, encargado de desarrollar diversos perfiles para definir de manera más exhaustiva estos estándares. Es por eso que no importa en qué plataforma se encuentre desarrollado el servicio Web Java, PHP, .Net este servicio se puede enlazar a cualquier otra aplicación mediante la implementación de este protocolo.

[19] Wikipedia, Framework para aplicaciones WEB. http://es.wikipedia.org/wiki/Framework_para_aplicaciones_Web
[20] Maestros de la WEB, Lenguajes de programación. http://www.maestrosdelWeb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-Web/

2.2.1 COMUNICACIÓN DE SERVICIOS WEB

La comunicación con los servicios Web estaba basada en lenguaje denominado WSDL (Web Services Description Language) y se encuentra basado en XML este describe la forma de comunicación, es decir, los requisitos del protocolo y los formatos de los mensajes necesarios para interactuar con los servicios listados en su catálogo. Las operaciones y mensajes que soporta se describen en abstracto y se ligan después al protocolo concreto de red y al formato del mensaje.
Así, WSDL se usa a menudo en combinación con SOAP y XML Schema. Un programa cliente que se conecta a un servicio Web puede leer el WSDL para determinar qué funciones están disponibles en el servidor. El cliente puede usar SOAP para hacer la llamada a una de las funciones listadas en el WSDL. En la siguiente figura se muestran los tipos de datos especiales que se incluyen en el archivo WSDL en forma de XML Schema.

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="http://services.com" xmlns:apachesoap="http://xml.apache.org/xml-soap"
xmlns:impl="http://services.com" xmlns:intf="http://services.com" xmlns:tns1="http://vo.services.com"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<!--WSDL created by Apache Axis version: 1.4
Built on Apr 22, 2006 (06:55:48 PDT)-->
 <wsdl:types>
  <schema elementFormDefault="qualified" targetNamespace="http://services.com"
xmlns="http://www.w3.org/2001/XMLSchema">
   <import namespace="http://vo.services.com"/>
   <element name="allEmpresas">
    <complexType/>
   </element>
   <element name="allEmpresasResponse">
    <complexType>
     <sequence>
      <element maxOccurs="unbounded" name="allEmpresasReturn" type="tns1:EmpresaVO"/>
     </sequence>
    </complexType>
   </element>
  </schema>
  <schema elementFormDefault="qualified" targetNamespace="http://vo.services.com"
xmlns="http://www.w3.org/2001/XMLSchema">
   <complexType name="EmpresaVO">
    <sequence>
     <element name="descripcion" nillable="true" type="xsd:string"/>
     <element name="idEmpresa" type="xsd:long"/>
     <element name="nombreempresa" nillable="true" type="xsd:string"/>
    </sequence>
```

```
  </complexType>
 </schema>
</wsdl:types>
  <wsdl:message name="allEmpresasRequest">
    <wsdl:part element="impl:allEmpresas" name="parameters"/>
  </wsdl:message>
  <wsdl:message name="allEmpresasResponse">
    <wsdl:part element="impl:allEmpresasResponse" name="parameters"/>
  </wsdl:message>
  <wsdl:portType name="EmpresaServices">
    <wsdl:operation name="allEmpresas">
      <wsdl:input message="impl:allEmpresasRequest" name="allEmpresasRequest"/>
      <wsdl:output message="impl:allEmpresasResponse" name="allEmpresasResponse"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="EmpresaServicesSoapBinding" type="impl:EmpresaServices">
    <wsdlsoap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="allEmpresas">
      <wsdlsoap:operation soapAction=""/>
      <wsdl:input name="allEmpresasRequest">
        <wsdlsoap:body use="literal"/>
      </wsdl:input>
      <wsdl:output name="allEmpresasResponse">
        <wsdlsoap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="EmpresaServicesService">
    <wsdl:port binding="impl:EmpresaServicesSoapBinding" name="EmpresaServices">
      <wsdlsoap:address location="http://localhost:8085/WebServices/services/EmpresaServices"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

Figura 2.1 Tipos de Datos WSDL

2.2.2 SEGURIDAD EN WEBSERVICES

2.2.2.1 AUTENTICACIÓN DEL CLIENTE DE WEBSERVICES

Para la autenticación del cliente que se conecta a un Web Services es necesario la implementación del SOAPHeaders en el archivo de configuración WSDL, en la siguiente figura se resaltan en negritas las partes que se deben de añadir.

```
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Header>
```

```
      <ns1:UserName xmlns:ns1="urn:thisNamespace">John Doe</ns1:UserName>
   </SOAP-ENV:Header>

   <SOAP-ENV:Body>
      <ns2:GetStockInfo xmlns:ns2="urn:thisNamespace">
         <ns2:symbol>FOO</ns2:symbol>
      </ns2:GetStockInfo>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Figura 2.2 Implementación SOAPHeaders

En la figura 2.3 se muestra la información que debe llenar el cliente, (utilizando el API de AXIS java 1.6) se genera el objeto SOAPHeaderElement y le añade la propiedad UserName con valor de "JohnDoe" este valor viaja hasta el servidor para ser validado.

```
import StockInfoNamespace.*;
import org.apache.axis.client.Stub;
import org.apache.axis.message.SOAPHeaderElement;
import javax.xml.soap.SOAPElement;

StockInfoServiceLocator locator = new StockInfoServiceLocator();
StockInfoService service = locator.getStockService();

// add an <AuthenticationInfo> node with a <UserName> subnode
// to the SOAP Header
SOAPHeaderElement header = new SOAPHeaderElement(
   "urn:thisNamespace", "AuthenticationInfo");
SOAPElement node = header.addChildElement("UserName");
node.addTextNode("John Doe");
((Stub) service).setHeader(header);

service.GetStockInfo("FOO");
```

Figura 2.3 Generación del Objeto SOAPHeaderElement

2.2.2.2 UTILIZACIÓN DE SSL

Consiste en la encriptación de los datos que viajan desde el servidor al cliente es comúnmente conocido como HTTPS. Proporciona sus servicios de seguridad cifrando los datos intercambiados entre el servidor y el cliente con un algoritmo de cifrado simétrico, típicamente el RC4 o IDEA, y cifrando la clave de sesión de RC4 o IDEA mediante un algoritmo de cifrado de clave pública, típicamente el RSA.

A continuación se describe un software para la generación de certificados digitales para utilizar el protocolo SSL (https), "OpenSSL es un proyecto de software desarrollado por los miembros de la comunidad Open Source para libre descarga y está basado en SSLeay desarrollado por Eric Young y Tim Hudson.

Consiste en un robusto paquete de herramientas de administración y librerías relacionadas con la criptografía, que suministran funciones criptográficas a otros paquetes como OpenSSH y navegadores Web (para acceso seguro a sitios HTTPS).

Estas herramientas ayudan al sistema a implementar el Secure Sockets Layer (SSL), así como otros protocolos relacionados con la seguridad, como el Transport Layer Security (TLS). Este paquete de software es importante para cualquiera que esté planeando usar cierto nivel de seguridad en su máquina con un sistema operativo Libre basado en GNU/Linux. OpenSSL también permite crear certificados digitales que se pueden aplicar al servidor, por ejemplo Apache." [21]

2.3 OUTSOURCING

Outsourcing es una técnica de administración, que consiste en la transferencia a terceros de ciertos procesos que no forman parte del giro principal del negocio; permitiendo la concentración de los esfuerzos en las actividades esenciales a fin de obtener competitividad y mejores resultados. La subcontratación también implica un considerable grado de intercambio bidireccional de información, coordinación y confianza.[22]

Muchas compañías contratan a empresas especializadas en la subcontratación para delegar la administración de las áreas más propicias a ello. Entre éstas se pueden encontrar las de informática, recursos humanos, administración de activos e inmuebles y contabilidad.

2.3.1 CONCEPTOS Y FUNCIONALIDADES BÁSICAS

Un contrato es un documento de carácter legal que recoge el alcance y características del servicio de Outsourcing. El contrato de Outsourcing debe definir principalmente los siguientes conceptos básicos:

Duración del servicio de Outsourcing, en el caso de la programación Web se deben definir los tiempos de entrega. En el contrato se deben especificar los acuerdos de niveles de servicio, esto es para que se puedan medir los cumplimientos de dichos acuerdo.
Un punto importante en la utilización de una empresa de Outsourcing es la propiedad intelectual, especialmente si se traspasa al proveedor la responsabilidad del desarrollo de aplicaciones. Si el

[21] Wikipedia , OpenSSL, http://es.wikipedia.org/wiki/OpenSSL
[22] http://www.gestiopolis.com/recursos/documentos/fulldocs/ger/outsourcingantonio.htm

proveedor no cumple con las fechas de entrega acordadas en el contrato o con los acuerdos de niveles de servicio puede ser acreedora a penalizaciones por parte de la empresa contratante.

Los niveles de servicio define el ámbito de aplicación del servicio (operación, mantenimiento, desarrollo), para sistemas de información concretos y la forma exacta de llevarlo a cabo. Es uno de los puntos más importantes de un contrato de Outsourcing y debe ser fácilmente medible. Para el establecimiento del nivel de servicio suele ser usual la realización conjunta, entre la organización contratante y la empresa de Outsourcing, de las siguientes actividades:

- Análisis de viabilidad que de fina el ámbito de aplicación
- Análisis detallado que determine todos y cada uno de los compromisos que van a ser contraídos por ambas partes.

Los activos son el conjunto de recursos informáticos que son propiedad de la organización contratante y que son susceptibles de ser traspasados a la empresa que proporciona el servicio de Outsourcing e incluso posteriormente ser recuperados. Estos activos pueden clasificarse en:

- Físicos: equipamiento físico de la organización.
- Lógicos: equipamiento lógico básico.
- Humanos: corresponden a transferencia de personal.

2.3.2 VENTAJAS

- Permite abaratar los costos de producción
- Permite obtener productos de mejor calidad si se hace un análisis detallado de la empresa a contratar y si se tiene un contrato adecuado, esto se debe a que se contrata un Outsourcing especializado en la tarea a realizar
- Reduce el número de tareas rutinarias
- Permite a la empresa dedicarse a tareas de mayor rentabilidad

2.3.3 DESVENTAJAS

Los trabajadores subcontratados no son empleados pagados de la empresa que de hecho presta el servicio, por lo cual no tienen un incentivo de lealtad hacia ésta. Muchos de los empleados de las empresas de Outsourcing son contratados por honorarios o por proyecto, es decir no tienen

una relación laboral con la empresa de Outsourcing, por este motivo en algunos casos existe una rotación de personal, ya que los empleados buscan mejores condiciones y estabilidad laboral. Sí no se tiene un contrato con las especificaciones de tiempos de entrega y de acuerdos de niveles de servicio puede generar mal servicio o productos de mala calidad por parte del Outsourcing.

2.4 SEGURIDAD EN INFORMÁTICA

Las organizaciones necesitan una serie de normas y reglas, así como diversos controles que le brinden rangos y grados de seguridad. El establecimiento de esas normas y reglas se pueden entender como seguridad informática; la cual consiste en asegurar que los recursos del sistema de información sean utilizados de forma correcta y de acuerdo al personal también seleccionado desde un principio. En el rango de seguridad informática se deben establecer los accesos a los niveles de información que corresponden también a los niveles de la organización.

La seguridad informática debe garantizar la intimidad y confidencialidad de la información a usuarios o clientes de una organización; incluso se busca que para garantizar dicha seguridad se establezcan políticas de seguridad, se proporcionen entornos seguros y legislaciones que brinden la confianza adecuada para emitir, usar o transferir información.

La seguridad se entiende como un estado en el que se indica que un objeto, sistema o producto está libre de peligros, daños o riesgos, o en el caso de la información de accesos indeseados por personas ajenas que incluso dentro de una organización no deban conocerla. La seguridad informática debe garantizar que los recursos informáticos de una compañía estén disponibles para cumplir sus propósitos, es decir, que no estén dañados o alterados por circunstancias o factores externos. La seguridad debe vigilar que se eviten al máximo la alteración en el funcionamiento o bien los resultados de la información.

Algunos términos de la seguridad en informática se enlistan a continuación:

- Activo: Recurso del sistema de información o relacionado con éste, necesario para que la organización funcione correctamente y alcance los objetivos propuestos.
- Amenaza: Es un evento que pueden desencadenar un incidente en la organización, produciendo daños materiales o pérdidas inmateriales en sus activos.
- Impacto: Medir la consecuencia al materializarse una amenaza.
- Riesgo: Posibilidad de que se produzca un impacto determinado en un activo, en un dominio o en toda la organización.

- Vulnerabilidad: Posibilidad de ocurrencia de la materialización de una amenaza sobre un activo.
- Ataque: Evento, exitoso o no, que atenta sobre el buen funcionamiento del sistema.
- Desastre o Contingencia: Interrupción de la capacidad de acceso a información y procesamiento de la misma a través de computadoras necesarias para la operación normal de un negocio. [23]

Un sistema de seguridad informática vigila que los activos tangibles (hardware) o intangibles (software), estén libres de amenazas, cuantifiquen los impactos, disminuyan los riesgos y eviten la vulnerabilidad ante posibles ataques, desastres o contingencias futuras, provenientes de fuentes como virus o usuarios mal intencionados.

"La vulnerabilidad es la exposición latente a un riesgo. En el área de informática, existen varios riesgos tales como: ataque de virus, códigos maliciosos, gusanos, caballos de Troya y hackers; no obstante, con la adopción de Internet como instrumento de comunicación y colaboración, los riesgos han evolucionado y ahora las empresas deben enfrentar ataques de negación de servicio y amenazas combinadas; es decir, la integración de herramientas automáticas de "hackeo", accesos no autorizados a los sistemas y capacidad de identificar y explotar las vulnerabilidades de los sistemas operativos o aplicaciones para dañar los recursos informáticos." [24]

"Para entender un sistema como seguro se deben tener en cuenta las siguientes cuatro características:

- Integridad: La información sólo puede ser modificada por quien está autorizado y de manera controlada.
- Confidencialidad: La información sólo debe ser legible para los autorizados.
- Disponibilidad: Debe estar disponible cuando se necesita.
- Irrefutabilidad (No repudio): El uso y/o modificación de la información por parte de un usuario debe ser irrefutable, es decir, que el usuario no puede negar dicha acción." [25]

Un sistema de seguridad informática debe establecer los lineamientos y la serie de controles, así como los procesos que brinden la confianza en que la información solo es accedida, visualizada o modificada por personal de manera delimitada y de acuerdo a las diversas jerarquías de una

[23] Wikipedia, Seguridad en Informática (2009). http://es.wikipedia.org/wiki/Seguridad_inform%C3%A1tica
[24] Revista RED, Seguridad Informática (2002).
[25] Wikipedia, Seguridad en Informática (2009). http://es.wikipedia.org/wiki/Seguridad_inform%C3%A1tica

empresa o bien a la información que deben conocer los diversos departamentos; puesto que es por demás entendido que el departamento o el área contable tiene un acceso de información distinto al del área operativa de una organización.

El sistema de seguridad informática debe garantizar en el rango más amplio posible que la información puede ser visualizada o modificada por personas delimitadas desde la descripción de las actividades de un puesto. El mismo sistema debe de procurar que dicha información sea actualizada de forma inmediata y que los resultados que presente sean verosímiles y verificables en cualquier momento.

Las cuatro características vigilan así entonces la accesibilidad a la información y a la presentación de esta. Ahora bien, es indispensable que, cualquier usuario de un sistema electrónico de información o bien un usuario de Internet tenga sus propios medios de seguridad para evitar que su información pueda verse comprometida.

El material informático o los programas deben contar con personal que vigile la seguridad de ese material y esos programas; en cuanto a la información que presentan, al funcionamiento, accesibilidad.; con mecanismos y controles adecuados que proporcionen y garanticen las características para considerar un sistema seguro. Un sistema de seguridad debe proteger:

- Información
- Equipos que la soportan.
- Usuarios

2.4.1 ATAQUE

En términos generales un ataque es una acción violenta en contra de una persona o cosa para hacerle daño. En informática la palabra “ataque” se refiere a una acción o un método por el cual un individuo haciendo uso de un sistema informático intenta dañar otro sistema (servidor, un equipo de cómputo, una red).

El ataque informático es un intento organizado y deliberado de una o más personas para causar daño o problemas a un sistema informático o red. Los blancos preferidos suelen ser los sistemas de grandes corporaciones o estados, pero ningún usuario de Internet u otras redes está exento.[26]

Un ataque informático siempre se efectúa por medio de la Internet aprovechando las vulnerabilidades de un sistema operativo, aplicación, servidor, red y por otros medios. Este se realiza con la finalidad de:

- Obtener accesos a una aplicación
- Robar información de la empresa, de los procesos de esta, clientes (cuentas bancarias, dirección, teléfono)
- Afectar el funcionamiento normal del servicio que presta la organización
- Utilizar el sistema de un usuario como un "rebote" para un ataque

2.4.1.1 CLASIFICACIÓN DE ATAQUES

2.4.1.1.1 DENEGACIONES DE SERVICIO

El objetivo del ataque de denegación de servicios también conocido como ataque DoS (Denial of Service), es interrumpir el funcionamiento normal de un servicio y por lo tanto este se vuelve inaccesible para los usuarios legítimos del sistema o de la red. Este tipo de ataque se efectúa mediante la saturación de los puertos con flujo de información, lo cual provoca que el servidor se sobrecargue y debido a esto ya no pueda continuar dado servicios. Por lo general, las denegaciones de servicio se dividen de la siguiente manera:

- Explotación de las debilidades del protocolo TCP/IP
- Explotación de las vulnerabilidades del software del servidor

2.4.1.1.2 INTRUSIONES

- Elevación de Privilegios: Este tipo de ataque consiste en aprovechar una vulnerabilidad en una aplicación al enviar una solicitud específica (no planeada por su diseñador). En ciertos casos esto genera comportamientos atípicos que permiten acceder al sistema con

[26]ALEGSA, Definición de ataque informático. http://www.alegsa.com.ar/Dic/ataque%20informatico.php

derechos de aplicación. Los ataques de desbordamiento de la memoria intermedia (búfer) usan este principio.27

- Ataques Malintencionados: Dentro de esta clasificación se encuentran los virus, gusanos y troyanos.

2.4.1.1.3 INGENIERÍA SOCIAL

La mayoría de los casos, por ignorancia o a causa de algún daño un usuario brinda información); sin embargo, el eslabón más débil es el mismo usuario. Muchas veces es él quien por ignorancia o a causa de un engaño, genera una vulnerabilidad en el sistema al brindar información (la contraseña, por ejemplo) o al abrir un archivo adjunto poniendo en peligro así la integridad del negocio.

2.4.1.1.4 PUERTAS TRAMPA

Son puertas traseras ocultas en un programa de software que brindan acceso a su diseñador en todo momento. Es por ello que los errores de programación de los programas son corregidos con bastante rapidez por su diseñador apenas publicada la vulnerabilidad. En consecuencia, queda en manos de los administradores (o usuarios privados con un buen conocimiento), mantenerse informados acerca de las actualizaciones de los programas que usan a fin de limitar los riesgos de ataques.

2.4.2 RIESGOS EN LAS APLICACIONES WEB

El uso cada vez más común del Internet se ha convertido en una herramienta para que las empresas hagan uso de las aplicaciones Web para sus sistemas ya que muchas de estas tienen sistemas distribuidos físicamente en diferentes geografías y con sistemas heterogéneos, es por eso que las aplicaciones Web se han convertido en la tendencia dominante en la actualidad.

En la actualidad la seguridad informática es un concepto que todo software debe incorporar, desde la fase inicial de su diseño hasta su puesta en funcionamiento, sin embargo en la práctica muchas veces la seguridad en las aplicaciones es tomada en cuenta hasta que los problemas se presentan. El desarrollador no sólo debe concentrarse únicamente en los usuarios y sus

[27] KIOSKEA, Introducción a los ataques. http://es.kioskea.net/contents/attaques/attaques.php3

requerimientos, sino también en los eventos que puedan interferir con la integridad del software y la información que éste maneja.

2.4.2.1 CODE INJECTION

La inyección de código o code injection en ingles hace uso de los errores al procesar información errónea, esto puede ser usado por un atacante al introducir condigo en un programa para cambiar la ejecución normal; incluso esta vulnerabilidad en algunos programas puede ser usada para la propagación de gusanos informáticos.[28]

2.4.2.1.1 PLATAFORMAS AFECTADAS

Todas las plataformas pueden ser vulnerables de ser atacadas con code injection, desde el HTML, hasta el Shell de un sistema operativo. A continuación se describen algunas de las principales plataformas para el desarrollo de aplicaciones Web.

"ASP.NET no contiene ninguna función que para incluir código inyectado, pero puede hacerlo a través del uso de las clases CodeProvider junto con la reflexión. Cualquier código PHP que utilice la función eval() corre el riesgo de sufrir un ataque de inyección de código. Java generalmente no brinda la habilidad para evaluar JSP's dinámicas".[29]

Sin embargo existen dos excepciones en este rubro:

- Inclusión Dinámica de JSP (<jsp:include>)
- Utilizar etiquetas de evaluación de JSP's proporcionadas por terceros

Portales y software desarrollado por comunidades a menudo requieren validación de Código dinámico en las plantillas y temas del sitio intercambiables. Si el portal requiere inclusiones dinámica y ejecución de código dinámico, hay un riesgo de inyección de código Java o JSP.

Para combatir estos, la primera línea de defensa la integran:

[28] OWASP, Direct Dynamic Code Evaluation , http://www.owasp.org/index.php/Direct_Dynamic_Code_Evaluation_('Eval_Injection')
[29] OWASP, Una Guía para Construir Aplicaciones y Servicios Web Seguros. 2005, pag 195.

- Preferir siempre inclusiones estáticas (<%include%>)
- Restringir la inclusión de archivos externos al servidor utilizando las políticas de seguridad de Java 2.
- Establecer reglas de cortafuegos para prevenir conexiones fuera de los límites de Internet.
- Asegurar que el código no interprete información proporcionada sin haberla validado.

En un ejemplo hipotético, el usuario puede seleccionar el uso de "X" como tema inicial. En este ejemplo, el código incluye dinámicamente un archivo llamado "X.tema.jsp" utilizando concatenación simple. Sin embargo, si el usuario ingresa otro tipo de información, puede tener la facultad de obtener código Java interpretado en el servidor. Bajo este escenario, el servidor de aplicaciones ya no es propiedad del usuario. En general, la inclusión dinámica y la evaluación dinámica de código debe ser mal vista dentro de los programas y se debe evitar lo más posible.[30]

2.4.2.1.2 TIPOS DE INYECCIÓN DE DATOS

La inyección de datos es común entre los individuos que gustan por atacar las páginas Web, generalmente usan esta técnica para obtener otra información que les ayude a atacar en mayor grado los sistemas.

Algunos de los tipos de inyección de datos buscan modificar los datos de la base de datos, este tipo de inyección se llama SQL Injection. El impacto de este tipo de inyección puede ser desde atacar el sistema Web hasta la perdida de información sensible a la empresa.

2.4.2.1.2.1 INYECCIÓN DE DATOS MALICIOSA

- Instalación de malware en alguna computadora por medio de la inyección de código a un navegador Web o en sus plug ins.
- Instalación de malware inyectando código a sistemas Web desarrollados en cualquiera de las plataformas disponibles.
- La inyección de código puede ser usada por medio del Shell del sistema operativo, para obtener mayores privilegios de los permitidos.
- Robo de sesiones desde el navegador Web usando inyección HTML/Script (Cross-site scripting).

[30] OWASP, Una Guía para Construir Aplicaciones y Servicios Web Seguros. 2005, pag 196

2.4.2.1.2.2 INYECCIÓN DE DATOS BENÉFICA

Así como hay inyección de código maliciosa, también hay inyección de código benéfica para el programador, por ejemplo, se puede modificar una tabla de la base de datos de un sistema existente usando la inyección de datos. Básicamente la inyección de código benéfica es útil para modificar el sistema de alguna manera eficiente y con menores costos.

2.4.2.1.2.3 INYECCIÓN DE CÓDIGO INESPERADA

Existe también la inyección de código inesperada, que es cuando el usuario ingresa caracteres inválidos en el sistema lo cual puede ocasionar que este funciona indebidamente con comportamientos inesperados, es por eso que se recomienda que se tenga un control de caracteres en los campos donde se requiere que el usuario ingrese datos al sistema.

2.4.2.2 REMOTE CODE-INCLUSION

De acuerdo al portal de seguridad HoneyNet, Remote Code Inclusion "es un ataque, que explota algún agujero de seguridad en la interfaz Web de una aplicación y con ello logran un ataque del sistema operativo subyacente, y la ejecución de código arbitrario."[31]

2.4.2.2.1 PLATAFORMAS AFECTADAS

Ataques de Remote Code-Inclusion pueden ocurrir en una gran variedad de aplicaciones PHP, notablemente en MAMBO CMS. Típicamente, el atacante incluye un script que intenta ejecutar un comando que contenga algún malware adjunto (Fetching Further malware). Estos scripts cuentan a menudo con bastantes características dañinas, como integración con bases de datos, y el permitir la invocación de comandos Shell, envío de mails, y permiten ver archivos dentro del Servidor Web. Este tipo de ataque es muy dañino, y ha resultado muy utilizado en un alto porcentaje de ataques a servidores Web que utilizan MAMBO CMS como aplicación de Sistema de Portales.

2.4.2.3 SQL INJECTION

La definición de SQL (**S**tructured **Q**uery **L**anguage) es un lenguaje de acceso a bases de datos

[31] HONEYNEt, Definición de Remote Code-Inclusion. http://www.honeynet.org/node/6

relacionales que permite especificar diversos tipos de operaciones en éstas como consultas, modificaciones, inserciones eliminaciones de datos. La inyección de SQL consiste en insertar código SQL invasor dentro de otra sentencia SQL con la finalidad de alterar su funcionamiento normal y hacer que se ejecute el código invasor dentro de la base de datos.

La inyección de SQL es comúnmente un problema de programación ya que consiste en ejecutar una sentencia concatenando parámetros que se reciben de lado del cliente. La solución principal es el uso de bindings. En el siguiente tema se explica el uso de estos.

2.4.2.3.1 PLATAFORMAS AFECTADAS

Cualquier lenguaje de programación que no utilice bindings en la comunicación con la base de datos.

2.4.2.4 CROSS-SITE SCRIPTING

El Cross-Site scripting conocido también como XSS "HTML Injection" es un tipo de inseguridad informática que consiste en realizar una serie de ataques por secuencia de comando entre aplicaciones Web, esto es ejecutar código scripting como VBScript o JavaScript.

La definición que se propone en el artículo "Cross Site Scripting" el Instituto de Seguridad de Internet menciona que "es una vulnerabilidad que afecta no tanto a los servidores como a los usuarios que navegan a páginas de Internet. La causa de la vulnerabilidad radica en la pobre verificación por parte de los sitios Web de las cadenas de entrada enviadas por los usuarios a través de formularios, o directamente a través del URL. Estas cadenas, en el caso de ser maliciosas, podrían llegar a contener scripts completos. Cuando esta entrada se le muestra dinámicamente a un usuario dentro de una página Web, en caso de contener un script, éste se ejecuta en el navegador del usuario dentro del contexto de seguridad de la página Web visitada. Como consecuencia en el ordenador del usuario se realizan todas las acciones que le sean permitidas a ese sitio Web, como por ejemplo interceptar entradas del usuario víctima o leer sus cookies."[32]

En el mismo artículo se explica que el funcionamiento este tipo de ataque comienza en cuanto el usuario ingresa a alguna aplicación solicitando para esto un usuario y password, entonces la

[32] Instituto de seguridad de Internet, XSS. http://www.instisec.com/publico/xss.asp

aplicación realiza una supuesta "validación", y se redirecciona a una página HTML la cual incluye el código por parte del atacante, una vez que se ejecuto el código este tiene los mismo privilegios de cualquier otro código legitimo en el mismo sitio Web.

En el libro Seth Fogie, Cross Site Scripting Attacks: Xss Exploits and Defense. 2007 se mencionan dos tipos de vulnerabilidad que pueden presentar:

- **Directa (Persistente):** Este tipo de ataque es poco común y su forma de actuar es identificando los puntos débiles dentro de la programación y de esta forma realizar inserciones de tags como pueden ser <iframe>, o <script>.
- **Indirecta (Reflejada):** Esta es un tipo de vulnerabilidad muy común, consiste en modificar valores que la aplicación Web utiliza para pasar variables entre dos páginas, sin usar sesiones y sucede cuando hay un mensaje o una ruta en la URL del navegador o en una cookie.

2.4.2.5 WEB SPOOFING

En el departamento de ciencias de la Computación de la Universidad de Princeton en el año de 1996 se publico el articulo "Web Spoofing: and Internet Game" en el cual se menciona que el Spoofing es un ataque que consiste en la suplantación de identidad y el cual consiste en la creación de tramas TCP/IP utilizando una dirección falseada, para poder llevar a cabo este ataque es necesario contar con tres; el atacante, atacado y un sistema suplantado."La idea de este ataque - al menos la idea - es muy sencilla: desde su equipo, un pirata simula la identidad de otra máquina de la red para conseguir acceso a recursos de un tercer sistema que ha establecido algún tipo de confianza basada en el nombre o la dirección IP del Host suplantado."[33]

El ataque antes mencionado es quizás el más conocido, IP Spoofing, pero existen diferentes tipos de ataques dependiendo de la tecnología a la que se refiera, por ejemplo existe el DNS Spoofing, ARP Spoofing, Web Spoofing.

2.4.2.5.1 DNS SPOOFING (SUPLANTACIÓN DE IDENTIDAD POR NOMBRE DE DOMINIO)

[33] Princeton University Department of Computer Science, Technical Report 540-96. Web Spoofing: an Internet Con Game. http://www.cs.princeton.edu/ 1996.

En el libro de seguridad en UNIX y redes se hace mención a este tipo de ataque, consiste en el falseamiento de “Nombre de dominio-IP”, esto se puede conseguir de diferentes formas, “desde modificando las entradas del servidor encargado de resolver una cierta petición para falsear las relaciones nombre-dirección, hasta comprometiendo un servidor que infecte la caché de otro.”[34]

2.4.2.5.2 ARP SPOOFING (SUPLANTACIÓN DE IDENTIDAD POR FALSIFICACIÓN DE TABLA ARP)

Tomando en cuenta que el protocolo ARP (Address Resolution Protocol) es el encargado de traducir direcciones IP a direcciones MAC para establecer la comunicación, es decir, para envía un paquete IP a otro Host se tiene que conocer la dirección Hardware de la maquina destino, el protocolo ARP es el encargado de determinar la dirección MAC (dirección Hardware) la cual corresponde a su vez a una dirección IP. Los Host que se encuentran en la red procesan la petición hasta que alguno responda a la solicitud comenzando así el envío de los paquetes a su destino.

La forma de ataque ARP Spoofing se basa en “la construcción de tramas de solicitud y respuesta ARP falseadas, de forma que en una red local se puede forzar a una determinada máquina a que envíe los paquetes a un Host atacante en lugar de hacerlo su destino legítimo”[35]. Una de las consecuencias de este tipo de ataque puede ir desde la negación de servicios, el robo o intercepción de datos.
Por otro lado en el artículo publicado en la página de Wikipedia se explica que la forma en la que actúa el ARP Spoofing es alterando la tabla ARP (IP-MAC) del atacado forzando así que esta envíe los paquetes de datos al Host atacante en lugar del destino correcto.

Tomando en cuenta los dos tipos de ataques anteriores el Web Spoofing se enfoca a las páginas Web, cuando la víctima accede a una aplicación Web el atacante direcciona la conexión a través de una página falsa con el fin de obtener información, como puede ser contraseñas o información subida a formularios. La página falsa se comporta como si fuera un proxy solicitando la información requerida por la victima e incluso saltándose la protección SSL.

El articulo “Spoofing: and Internet Con Game” de la Universidad de Princeton señala que este tipo de ataque “permite a un pirata visualizar y modificar cualquier página Web que su víctima solicite a través de un navegador, incluyendo las conexiones seguras vía SSL, mediante código malicioso un atacante crea una ventana del navegador correspondiente, de apariencia inofensiva, en la máquina

[34] Huerta, Antonio Villalón. Seguridad en Unix y redes
[35] Huerta, Antonio Villalón. "Seguridad en Unix y redes"

de su víctima; a partir de ahí, enruta todas las páginas dirigidas al equipo atacado (incluyendo las cargadas en nuevas ventanas del navegador) a través de su propia máquina, donde son modificadas para que cualquier evento generado por el cliente sea registrado (esto implica registrar cualquier dato introducido en un formulario, cualquier click en un enlace).[36]

2.4.2.6 DENEGACIÓN DE SERVICIOS Dos

Consiste en enviar una cantidad de mensajes o peticiones mayores de la que el servidor puede soportar llegando el momento en que este se satura y ya no puede responder. En las aplicaciones Web consistiría en que el cliente abriera un número ilimitado de conexiones para estresar el servidor y saturarlo.

2.4.2.6.1 PLATAFORMAS AFECTADAS

Esto afecta a todo tipo de plataformas la manera de evitar esto es monitoreando las ips que realizan la petición a nuestra aplicación si estas sobrepasan de un número de conexiones http en rango de tiempo corto es considerado un ataque.

2.4.2.7 USURPACIÓN DE PRIVILEGIOS EN LAS APLICACIONES

De acuerdo con el Diccionario de la Real Academia Española se define usurpación como el apoderamiento de una propiedad o un derecho que legítimamente pertenece a otro, en donde dicho poder generalmente se obtiene con violencia.[37]

En rama de la informática la usurpación es aplicada para distintos conceptos como lo son: usurpación de identidad, usurpación de privilegios o la usurpación de diseños; para fines de este trabajo se define la usurpación de privilegios como la toma de permisos que fueron asignados a una persona en específico, pero que un tercero ha utilizado para obtener beneficios personales.

2.4.2.8 ATAQUES A LA AUTENTICACIÓN DE SESIÓN

[36] Technical Report 540-96, Princeton University Department of Computer Science, Web Spoofing: an Internet Con Game. 1996.
[37] Rae Diccionario De La Lengua Española - Vigésima segunda edición. http://buscon.rae.es/draeI/SrvltConsulta?TIPO_BUS=3&LEMA=usurpar

La autenticación puede ser definida como un proceso mediante el cual se intenta verificar y asegurar la identidad de un usuario, es decir, que el usuario realmente sea el dueño de la cuenta con la que se está firmando a una aplicación.

De acuerdo con un artículo publicado en la página del Instituto Mexicano de Contadores Públicos se define la autenticación el hecho de que una computadora verifique que un usuario es quien dice ser, esto se hace con el password, conocido como el mecanismo de autenticación más utilizado; dicho mecanismo se activa cuando el usuario le facilita a la computadora su identidad y el password asociado a éste, la maquina verifica que el password introducido está relacionado con la identidad proporcionada, si éste es el caso, el usuario está autenticado; en caso contrario, el usuario no puede acceder al sistema.[38]

Relacionando las definiciones anteriores, se definen los ataques a la autenticación de sesión como el robo o plagio de identidad al firmarse en una aplicación, correo o sistema de seguridad con un acceso que no es propio.

2.4.2.9 FUERZA BRUTA

"Se denomina ataque de fuerza bruta cuando el atacante intenta ingresar al sistema probando todas las combinaciones posibles hasta encontrar aquello que permita el acceso."[39]

Un factor importante para este tipo de ataque es el costo que implica, es decir los caracteres usados para la clave. Cuando la clave solo utiliza números es más fácil de descifrar que las que incluyen diversos caracteres como letras y números. Para determinar el esfuerzo requerido existe una fórmula:

$$2^{n-1}$$

Donde n es la longitud de la clave, los ataques de fuerza bruta usan un método de prueba y error, son muy costosos en tiempo computacional.

2.4.2.9.1 DETERMINAR VULNERABILIDADES

Para evitar que un sistema Web sea vulnerable se deben seguir las siguientes recomendaciones:

[38] IMCP, Criptología y Seguridad en Internet (Enero 2009. http://www.imcp.org.mx/spip.php?article108
[39] Wikipedia, Ataque de fuerza bruta. http://es.wikipedia.org/wiki/Ataque_de_fuerza_bruta

- No usar diccionarios en un solo idioma
- Verificar diccionario de contraseñas comunes
- No mostrar mensajes de error a los usuarios comunes
- Se sugiere tener un registro de autentificación fallida y ligarlo a un mecanismo de bloqueo
- Restringir el número de intentos con la misma dirección IP o usuario a menos de 5.[40]

2.4.2.9.2 PREVENCIÓN

Para aminorar los ataques de fuerza bruta se recomienda que exista un retardo desde que el usuario manda sus credenciales hasta que la falla es reportada, este retardo puede ser incremental, esto disminuye el ataque de fuerza bruta por elevar los tiempo entre cada intento.

Se debe informar a los usuarios del error con mensajes adecuados que no expongan información que pueda ser usada por el atacante. Se recomienda llevar un registro de los intentos de autentificación fallidos, esto puede ser incluso usado en auditorias al sistema.

También se recomienda que se destruyan las sesiones de usuarios después de varios intentos fallidos, el registro de autentificación puede ser usado para evitar múltiples accesos en la misma página.

Estas son algunas de las sugerencias para evitar ser víctimas de ataques de fuerza bruta, como la mayoría de los ataques se pueden evitar si se planea adecuadamente la seguridad al diseñar el sistema Web.

2.4.3 RIESGOS EN LAS APLICACIONES WEB GENERADOS POR LA EMPRESA QUE PRESTA EL OUTSOURCING.

De acuerdo al Portal Gestiopolis.com, “El uso de servicios de Subcontratación en las empresas consiste en la transferencia a terceros de ciertos procesos complementarios que no forman parte del giro principal del negocio, permitiendo la concentración de los esfuerzos en las actividades esenciales a fin de obtener competitividad y resultados tangibles”.

[40] OWASP. Una Guía para Construir Aplicaciones y Servicios Web Seguros, 2005, pag 138.

En los últimos años, a través de este tipo de servicios, también llamado Outsourcing, " se busca que exista una cooperación intensa entre el cliente y el proveedor en la que los proveedores adoptan los mismos sistemas que los clientes, de manera de proporcionar así una mejor relación de trabajo."[41]

A pesar de las grandes ventajas que le brinda a una empresa manejar este tipo de contratación, también existen riesgos que se deben tomar en cuenta al realizar un esquema de trabajo de este tipo.

2.4.3.1 ROBO DE INFORMACIÓN CONFIDENCIAL

Por exceso de confianza en su personal o en sus programas de seguridad, las empresas sufren de robo de información, esto de acuerdo a Sandra Hernández, Consultora en Seguridad y colaboradora del medio Radiotrece Noticias, "El 60 por ciento de las empresas enfrentan el robo de información confidencial. Los empresarios creen que nada le puede pasar a la información de su red, hasta que un día, los hackers o trabajadores de la empresa penetran a sus archivos extrayendo información contable, administrativa, así como dinero de sus cuentas bancarias."[42]

Según el socio de Tecnología Andrés Acuña, de la firma Ernst & Young afirma que "el 55% de los defraudadores están en cargos gerenciales y, en esa línea, lo que hay que hacer es "implementar todas las medidas que permiten restringir el acceso a sistemas operativos, bases de datos, redes y comunicaciones".[43]

2.4.3.1.1 ROBO DE CÓDIGO FUENTE.

"El robo de Código fuente es aquel ocurrido cuando un código propietario de la empresa es sustraído ilegalmente del equipo que lo contiene". [44]

Hoy en día, la mayoría de empresas que manejan información crítica cuentan con políticas relativas al resguardo de la información que manipulan sus empleados del área de Sistemas, tal es el caso de la Aseguradora Zurich México,[45] que cuenta con una carta de Aceptación y compromiso

[41] Gestiopolis, Outsourcing, http://www.gestiopolis.com/recursos/documentos/fulldocs/ger/outsourcingantonio.htm
[42] RadioTrece, Robo de información y datos en las empresas. http://www.radiotrece.com.mx/2007/10/29/robo-de-informacion-y-datos-en-las-empresas/
[43] Andrés Acuña, Economía y negocios. http://www.economiaynegocios.cl/noticias/noticias.asp?id=45823
[44] http://www.economiaynegocios.cl/noticias/noticias.asp?id=45823
[45] Grupo Zúrich México, Carta de Bienvenida y Políticas de Seguimiento, 2009, página 5.

de confidencialidad, que cita, entre otras cosas, la prohibición de difundir el código fuente a través de cualquier medio de salida externa, (correo electrónico, Mensajero electrónico, Dispositivo USB).

2.4.3.1.2 ROBO DE LOS DATOS DE NEGOCIO EN BD

"Ante el vacío legal y la negligencia judicial, robar datos es un delito que avanza jaqueando la privacidad y confiabilidad informática por la adquisición en la captación de base de datos en el mercado negro ilegal. Se trata de datos robados a organismos oficiales y vendidos por bajo precio a los delincuentes que operan en desde un equipo con acceso a Internet."[46]

Para que disminuya el robo de información de base de datos, se pueden utilizar diferentes tecnologías capaces de reducir estos riesgos. Entre las opciones disponibles se encuentra el "firewall de base de datos"[47], que permite conocer la actividad de programadores, personal de mantenimiento, usuarios y administradores.

Esto posibilita un alto nivel de conocimiento de lo que sucede en los almacenes de información, ya que como dicen los expertos: el software es inocente, pero quienes lo operan no. Por otra parte, también existe la encriptación de datos que permite una granularidad donde la información sólo es accesible para quienes deben usarla y no para quienes deben administrar los sistemas donde se encuentra.

2.4.3.1.3 ACCESO NO AUTORIZADO A OTROS SISTEMAS DE LA EMPRESA.

"El constante manejo de información en las empresas de personal interno y externo, bajo el esquema de Outsourcing implica que estas últimas puedan tener a su disposición información de acceso a otros sistemas de la empresa, algunas veces no autorizado. Esto según el Instituto Nacional de las Tecnologías de Comunicación, que refiere que se "hace necesaria la labor de un CISO (Chief Information Security Officer) que gestione y controle todos los accesos a datos de la organización. Algunas entidades instan incluso a la incorporación de estos responsables de la seguridad de la información a los equipos directivos."[48]

[46]Noticias, Ciberdelito lo nuevo informática. http://www.noticias.com/articulo/18-08-2006/emil-domec/ciberdelito-lo-nuevo-informatica-como-robar-base-datos-570j.html
[47]Seguridad información, La otra cara del Robo de Información. http://seguridad-informacion.blogspot.com/2007/11/la-otra-cara-del-robo-de-informacin.html
[48] Inteco, Seguridad de información en equipos directivos. http://www.inteco.es/

Se trata de una figura cuya labor pasa por garantizar la seguridad en la difusión de la información confidencial teniendo en cuenta los riesgos que supone perder el control de contenidos privados o restringidos como contratos, transacciones financieras o datos de clientes.

"El Responsable de Seguridad tiene que saber conjugar la aplicación de las nuevas tecnologías reduciendo, al mismo tiempo, el riesgo de fugas de información", explica Marie-Claire Pfeifer.[49]

2.4.3.2 INSERCIÓN DE CÓDIGO MALICIOSO

Code Injection es "la explotación de un bug en una computadora causado por procesar datos inválidos,[50] La inyección de código puede ser usada por un atacante para introducir (o inyectar) código dentro de un programa de computadora para cambiar el curso de ejecución".

El resultado de un ataque de inserción de código puede ser muy dañino, por ejemplo este tipo de vulnerabilidad es usada por algunos Gusanos (Worms) para propagarse.

2.4.4 CONTROLES

2.4.4.1 DEFINICIÓN

Un control se define como la verificación o validación de los resultados obtenidos dentro de la administración de un sistema, ya que permite definir si se han ejecutado correctamente cada uno de los pasos que definen si un sistema trabaja en forma eficaz y eficiente.

El control interno son un conjunto de reglas y actividades que tienen como objetivo el asegurar que los activos de la organización sean administrados bajo los procedimientos establecidos y así lograr ofrecer servicios de calidad y orientados hacia la mejora continua.

Los objetivos del control interno informático son los siguientes:

- Establecer como prioridad la seguridad y protección de la información del sistema computacional y de los recursos informáticos de la empresa
- Promover la confiabilidad, oportunidad y veracidad de la captación de datos, su procesamiento en el sistema y la emisión de informes en la empresa

[49] TechWeek, Marie-Claire Pfeifer, Riesgo de fugas de información. http://www.techweek.es/autores/aut112
[50] Wikipedia, Code Injection. http://www.en.wikipedia.org/wiki/Code_injection

- Implementar los métodos, técnicas y procedimientos necesarios para coadyuvar al eficiente desarrollo de las funciones, actividades y tareas de los servicios computacionales, para satisfacer los requerimientos de sistemas en la empresa
- Instaurar y hacer cumplir las normas, políticas y procedimientos que regulen las actividades de sistematización de la empresa
- Establecer las acciones necesarias para el adecuado diseño e implementación de sistemas computarizados, a fin de que permitan proporcionar eficientemente los servicios de procesamiento de información en la empresa. [51]

Los elementos fundamentales del control interno informático se enlistan a continuación:

- Controles internos sobre la organización del área de informática
- Controles internos sobre el análisis, desarrollo e implementación de sistemas.
- Controles internos sobre operación del sistema
- Controles internos sobre los procedimientos de entrada de datos, el procesamiento de información y la emisión de resultados
- Controles internos sobre la seguridad del área de sistemas. [52]

2.4.4.2 ESTÁNDARES Y MÉTODOS

Para tener un buen control interno se deben especificar ciertos métodos y estándares que apliquen a actividad ejecutada dentro del área de informática. “Es de suma importancia estandarizar el desarrollo de todas las actividades y funciones a fin de que estas se realicen de manera uniforme conforme a las necesidades concretas de las unidades de informática que integran la empresa. Se deben establecer de manera homogénea y uniforme todos aquellos procedimientos y metodologías informáticas que permitan estandarizar la operación de los sistemas, así como para el desarrollo de nuevos sistemas.”[53]

Es recomendable que la estandarización contemple la instalación del hardware, el diseño del software por adquirir o bien a desarrollar, la implementación y manejo de las bases de datos, la arquitectura de la red y la seguridad establecida dentro de la organización para que esta pueda adecuarse al sistema adquirido, así como cada uno de los elementos mencionados.

[51] La Productividad en la Informática pág. 1 (2004), http://yaqui.mxl.uabc.mx/~halbarran/control.doc
[52] La Productividad en la Informática pág. 1 (2004), http://yaqui.mxl.uabc.mx/~halbarran/control.doc
[53] La Productividad en la Informática pág. 4 (2004), http://yaqui.mxl.uabc.mx/~halbarran/control.doc

Para efectos de este trabajo se mencionan algunos aspectos relacionados con la estandarización de metodologías para el desarrollo de sistemas. “La empresa debe adoptar alguna metodología que sea acorde al desarrollo de sus proyectos de sistemas, la aplicación de una metodología estandarizada para el desarrollo de un proyecto informático garantiza la uniformidad en la aplicación de cualquier sistema y contribuye en gran medida a la máxima eficiencia en el uso de los recursos informáticos del área de sistemas.”[54]

Dentro de los desarrollos realizados por personas ajenas a la organización, es decir, Outsourcing, no existen estándares que puedan ser aplicados en la implementación de estos, por ello es de suma importancia el establecer una metodología que permita apegar el desarrollo de estos sistemas a ciertas técnicas, que dan como resultado sistemas similares y compatibles con los que ya existen dentro de la organización.

Algunos elementos a incluir en las metodologías son los siguientes:

- Estandarización de métodos para el diseño de sistemas
- Lineamientos en la realización de sistemas
- Uniformidad de funciones para desarrollar sistemas
- Políticas para el desarrollo de sistemas
- Normas para regular el desarrollo de proyectos.[55]

2.4.5 AUTENTICACIÓN Y AUTORIZACIÓN

2.4.5.1 OBJETIVO Y DEFINICIÓN

Se define la autenticación y la autorización como algo interiormente ligado debido a que la autenticación es el establecimiento y la confirmación de algo y la autorización es aquella que después de verificar la identidad auténtica permite el acceso a zonas restringidas de información.

Es importante resaltar que la autenticación y la autorización va ligada principalmente a los accesos de los usuarios a distintos niveles de información. La autenticación se encuentra en verificar la compatibilidad y la procedencia ya sea de un programa una función, una secuencia o incluso una persona. Ahora bien, una vez verificado lo anterior un sistema puede autorizar el acceso a los

[54]La Productividad en la Informática pág. 6 (2004), http://yaqui.mxl.uabc.mx/~halbarran/control.doc
[55]La Productividad en la Informática pág. 6 (2004), http://yaqui.mxl.uabc.mx/~halbarran/control.doc

recursos del mismo sistema o programa o a distintos niveles de identificación. Estos dos niveles de seguridad determinan si una persona tiene la autoridad para acceder o realizar dicha operación.

Así, la autenticación y la autorización son actividades o partes de un proceso o sistema estrechamente relacionados. Un sistema de autenticación y autorización debe tener características que lo hagan más viable, tales como la confiabilidad, que no sea oneroso para una organización (económicamente factible), que soporte con éxito cierto tipo de ataques o riesgos y por último que sea aceptable de fácil manejo y comprensión para los usuarios.

2.4.5.2 TIPOS DE AUTENTICACIÓN Y CONTROLES

Los tipos de autenticación están en función de lo que se utiliza para la verificación de una identidad y se dividen en:

- Sistemas basados en algo conocido. Este nivel de veracidad de una identidad corresponde a un nivel de seguridad que reconoce un parámetro establecido tal como un Password o passphrase el cuál al momento de identificar un código en letras y/o números permite la utilización de un programa o herramienta. Este tipo de sistemas posee la desventaja de que un código de este tipo puede ser vulnerado por otra persona sin que confirme una identidad en el sentido estricto o literal de la palabra (identifica un algo más que identificar un quién).
- Sistemas basados en algo poseído. Aquellos medios que determinan un acceso y otorgan la autorización por medios preestablecidos anteriormente en los cuales se otorga un objeto a un usuario para acceder a cierto programa, sistema, información o incluso a otro lugar físico. Por ejemplo las tarjetas de identidad, las tarjetas inteligentes, algunos dispositivos USB en epass. Este tipo de sistema corre con la desventaja de que el instrumento para la autenticación-autorización pueda ser robado por una persona ajena a un sistema de seguridad determinado
- Sistemas basados en una característica física del usuario. Principalmente este tipo de autenticación y autorización es para determinar el acceso a lugares también físico. Debido a que las características humanas son únicas e irrepetibles se puede considerar menos vulnerable que los dos anteriores pues está basado generalmente en timbres de voz, huellas dactilares o verificación de retina en sus formas más comunes.

Cabe mencionar que para tener mejores niveles de seguridad en el acceso a la información y para confirmar de mejor manera la identidad de un usuario se han realizado sistemas en los cuales se

combinan las características de los anteriores mencionados, como los Netkeys bancarios y los Token, a sabiendas de que la información financiera y el uso de esos recursos (financieros) debe ser mucho más restrictivo y seguro. De acuerdo a lo anterior un Netkey es un claro ejemplo de la combinación de un sistema de seguridad combinado en el cual además de que el elemento de seguridad debe estar en posesión de alguien, ese alguien debe conocer cierto código o password para acceder a la información e incluso para usarla. En sistemas bancarios que incluyen fondos de inversiones más elevados, se combinan incluso los tres sistemas anteriormente descritos un Token que alguien posee, más un código que solo el usuario conozca mas el reconocimiento de escritura estilográfico para la utilización de ese fondo.

A la combinación de sistemas de autenticación y futuro acceso se le conoce como sistema de autenticación por multifactor en el cual se describe algo que el usuario es, más algo que el usuario sabe, mas algo que el usuario tiene, de esta manera después de revelarse una identidad se permita entonces una inmediata autorización.

Control de acceso. El control de acceso es un ejemplo básico también de la utilización de uno o más de los sistemas descritos debido a que represente un sistema informático que solamente puede ser utilizado por aquella persona que estén autorizado y casi de manera obvia cuente con el conocimiento para lograr un acceso.

2.4.5.3 RECOMENDACIONES PARA AUTENTICACIÓN Y AUTORIZACIÓN

La utilización de los métodos o sistemas de autenticación-autorización depende del grado de restricción y evaluación al riesgo y depende de que tan fuertes sean los procesos de administración de usuarios y de la forma en la que se use de una manera mas apropiada la clasificación y evaluación de bienes. De acuerdo a lo anterior se deben establecer parámetros y lineamientos para que los sistemas de autenticación y autorización puedan ser de una forma más prácticos y mejor utilizados.

Se recomienda también re-autenticar al usuario en áreas de acceso más restringido y de valor informático más limitado, así como atentas a la transacción y no al usuario para aquellas operaciones en las que se pueda disminuir el error humano. Se debe aceptar que las contraseñas son trivialmente rotas y pueden no ser adecuadas para sistemas de alto valor por lo que se recomienda que las contraseñas de caracteres deban de ser modificadas en periodos de tiempo.

Se recomienda que la persona encargada de limitar los accesos al uso o a la visualización de la información entrene a los usuarios a usar los sistemas de autenticación, que los aliente a cambiar las contraseñas, al uso de claves para los accesos a través de passwords para que posteriormente relaje los requerimientos de expiración de una contraseña.

2.4.6 MANEJO DE SESIONES

El uso de sesiones es un método ampliamente extendido en muchos tipos de aplicación Web. Una sesión es la secuencia de páginas que un usuario visita en un sitio Web desde que entra en un sitio hasta que lo abandona.[56]

El termino sesión en PHP, se aplica a esta secuencia de navegación, para ello se crea un identificador único que es asignado a cada una de las sesiones de navegación. A este identificador de sesión se le denomina comúnmente, como sesión.

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas Web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica.

El proceso en cualquier lenguaje de programación debe responder a las siguientes consideraciones:

- ¿Existe la sesión?
- Si existe la sesión se retorna
- Si no existe la sesión se crea una nueva
- se genera un identificador único

El manejo de sesiones tiene como objetivo primordial asegurar que los usuarios autenticados posean una amplia y criptográfica segura asociación con sus sesiones, haciendo cumplir los controles de autorización y previniendo los ataques Web como la reutilización, falsificación e intercepción de sesiones.[57]

[56] ZULCA Manani, John. PHP y MySQL: Aplicaciones Web. 2008.
[57] The Open Web Aplication Security Project. Una guía para construir aplicaciones y servicios Web seguros. Black Hay, Segunda edición, 2005, pp.155-158.

La descripción del manejo de sesiones se explica dado que los clientes pesados guardan información en memoria asignada por el sistema operativo durante la duración de la ejecución del programa. Con las aplicaciones Web.

2.4.6.1 SEGURIDAD DE SESIONES

El manejo de sesiones Web es una técnica o herramienta que permite vincular información a un usuario en concreto durante el proceso de visita a un sitio Web.

Esta herramienta se utiliza habitualmente para labores de autenticación y seguimiento de la actividad de los usuarios en aplicaciones que tienen partes privadas para las que se necesita algún tipo de control de acceso. El manejo de sesión facilita y unifica las tareas de control y supervisión de accesos, pero si presenta alguna vulnerabilidad puede dar al traste con la seguridad de toda la aplicación.

2.4.6.1.1 COOKIES Y SESIONES

Una cookie es un fragmento de información que se almacena en el navegador del visitante de una página, a petición del servidor de la misma. Esta información puede ser luego recuperada por el servidor en posteriores visitas para que se pueda conservar información entre una página y otra ya que el protocolo HTTP es incapaz de mantener información por sí mismo.[58]
Los usos más frecuentes de las cookies son:

- Mantener opciones de visualización.
- Almacenar variables.
- Realizar un seguimiento de la actividad de los usuarios.
- Autenticación.

Una sesión Web consiste en un arreglo de datos que se mantiene en el servidor. Cada sesión se identifica por un código único que se utiliza para hacer referencia a la misma. En la sesión se pueden almacenar una serie de variables que son conservadas hasta que se produzca su caducidad o sea explícitamente borrada.

[58] Pinuaga, Ramón. Seguridad en las sesiones de las aplicaciones Web I. S21s

2.4.6.1.2 COOKIES DE SESIÓN

Cuando una cookie se usa para autenticación normalmente se hace mediante la utilización de sesiones. En la cookie se almacena el identificador de una sesión que es asociado al usuario que accede a la aplicación.

Para que una cookie de sesión se considere segura debe cumplir una serie de condiciones:

- La única información que debe contener es el identificador de la sesión asociada. El resto de variables se almacenan internamente en el array que se encuentra en el servidor.
- El identificador de sesión debe ser único, aleatorio y no predecible. Con el fin de evitar suplantaciones de identidad.
- Las variables almacenadas en la sesión, deben permanecer lo más protegidas posible del exterior. El usuario no debe conocer su nombre ni su valor, y tampoco puede modificarlas a voluntad.
- Cuando el usuario ha terminado su actividad o ha transcurrido un periodo de tiempo prudencial, la sesión debe borrarse.

2.4.6.2 RECOMENDACIONES PARA MANEJO DE SESIONES

Los marcos de aplicaciones más populares contienen una adecuada implementación que utilizan robustos y bien conocidos manejadores de sesiones, sin embargo las primeras versiones frecuentemente tienen vulnerabilidades. Se recomienda siempre utilizar la última versión de la tecnología elegida, para que el manejador de sesiones sea más robusto y utilice credenciales criptográficas fuertes.

Considere cuidadosamente donde guarda el estado de la aplicación:

- Datos sobre autorización y roles deben ser guardados solamente del lado del servidor.
- Datos sobre la navegación son ciertamente aceptables en la URL siempre y cuando los controles de validación y autorización sean efectivos.
- Las preferencias del usuario (temas y lenguaje del usuario) puede ser almacenado en cookies.
- Datos de formularios no deberían contener campos ocultos, si se encuentran ocultos, probablemente necesiten estar protegidos y solo disponibles del lado del servidor. Sin

embargo, los campos ocultos deben ser utilizados para la protección de secuencias y ataques de pharming

Los datos de formularios de varias páginas pueden ser enviados de vuelta al usuario en los siguientes dos casos:

- Cuando existen controles de integridad para prevenir la manipulación.
- Cuando los datos son validados luego de cada envío del formulario, o al menos al final del
- proceso de envío.

Los secretos de la aplicación como credenciales del lado del servidor e información sobre roles nunca debería ser visible al cliente. Estos deben ser guardados en una sesión o del lado del servidor.

2.4.7 VALIDACIÓN DE DATOS

La validación es un proceso que consiste en realizar un filtro de los datos proporcionados por el usuario si estos son correctos permite el acceso, si los datos son erróneos regresa mensajes de error basándose en procedimientos definidos.

Para la seguridad en informática la validación de datos es "una de las áreas más importantes a tener en cuenta, especialmente en el desarrollo de sistemas conectados a redes como internet. Validar datos hace referencia a verificar, controlar o filtrar cada una de las entradas de datos que provienen desde el exterior del sistema."[59] Por lo tanto se debe de identificar los datos de entrada y que estos sean los esperados y no otros, identificar que no exista código oculto, entre otras cosas.

En las aplicaciones Web la validación de datos se realiza desde los formularios basándose en la arquitectura cliente - servidor. Del lado del cliente la validación de datos consiste en enviar un mensaje de error, por ejemplo que el usuario es incorrecto o bien que los datos que está proporcionando están incompletos, mientras que del lado del servidor vuelve a realizar la validación de los datos para evitar así que el usuario pueda realizar alguna modificación de la información esto puede realizarse por medio de Java Script.

[59] http://www.alegsa.com.ar/Dic/validacion%20de%20datos.sphp

2.4.7.1 MANEJO DE VALIDACIÓN DE DATOS

Una parte importante en el desarrollo de una aplicación Web es la validación de los datos ya que se debe de asegura que estos se encuentren conforme a las reglas del negocio, por ejemplo que el password este conformado de un mínimo de caracteres o bien que no exista duplicidad de claves de usuario, estas reglas de validación facilitan el manejo de formularios en las aplicaciones Web desarrolladas por un tercero.

Java Script permite desarrollar aplicaciones Web con la opción de realizar la modificación y en última instancia realizar la validación de los valores introducidos por el usuario. Se debe de considerar que los scripts utilizados para la validación de los datos no sustituyen la seguridad que debe de establecerse en la aplicación del servidor que recibe la información. La aportación que realiza Java Script consiste en enviar mensajes de error instantáneos ya que no se envían los datos al servidor para que este realice la validación de los mismos.

Algunas de las validaciones típicas que realiza Java Script a los formularios que se encuentran en una aplicación Web es la validación de los formularios que son obligatorios, que el formato de entrada sea el esperado, por ejemplo una fecha, teléfono, dirección electrónica, comprobar que los datos no sobrepasen la longitud o el número de líneas permitidas.

2.5 PRINCIPIOS DE PROGRAMACIÓN SEGURA

De acuerdo a la guía de Desarrollo del Open Web Application Security, "el personal encargado de desarrollar soluciones Informaticas necesita una guía para producir aplicaciones seguras por diseño. Los principios de seguridad tales como confidencialidad, integridad, y disponibilidad aunque son importantes no cambian."[60] La aplicación se vuelve más robusta cuanto más sean aplicados.

Esto ha desencadenado en un esfuerzo de la Industria de la tecnología informática para estandarizar la terminología y taxonomía de Programación Segura.

2.5.1 CONTROLES

[60] The Open Web Application Security Project. Una guía para construir aplicaciones y servicios Web seguros. Black Hay, Segunda edición, 2005, pp.

Howard Le Blanc plantea que “La selección de controles sólo es posible después de clasificar los datos a proteger".[61] Por ejemplo, controles aplicables a sistemas de bajo valor tales como blogs y foros son diferentes al nivel y número de controles adecuados para la contabilidad, sistemas de alto valor de banca y comercio electrónico.

2.5.2 ATACANTES

Saitta, Larcom, and Michael Eddington escriben en un artículo la siguiente afirmación: [62] “En el diseño de controles para prevenir el mal uso de su aplicación, debe considerar los atacantes más probables (en orden de posibilidades y pérdidas actualizadas de más a menos):

- Equipo o desarrolladores descontentos.
- Ataques “Accionados por” como efectos secundarios o consecuencias directas de un virus, o ataque de gusano o troyano.
- Atacantes criminales motivados, tales como el crimen organizado.
- Atacantes criminales contra tu organización sin motivo, como defacers.”

2.5.3 BASES DE LA SEGURIDAD DE LA INFORMACIÓN

La seguridad de la información se ha mantenido sobre los siguientes pilares, planteados en la normativa ISO27001, en su apartado dedicado a la Seguridad de la Información.[63]

- **Confidencialidad:** Permitir acceso únicamente a los datos a los cuales el usuario está permitido.
- **Integridad:** Asegurar que los datos no se falsifican o alteran por usuarios no autorizados.
- **Disponibilidad:** Asegurar que los sistemas y datos están disponibles para los usuarios autorizados cuando lo necesiten.

Los siguientes principios están todos relacionados a esos tres pilares. Cuando se considera la construcción de un control, el considerar cada pilar sucesivamente, ayuda en la producción de un robusto control de seguridad.

---

[61] Howard and LeBlanc, Writing Secure Code, 2nd Edition, pp 69 – 124, © 2003 MicrosoftPress, ISBN 0-7356-1722-8
[62] Saitta, Larcom, and Michael Eddington, A conceptual model for threat modeling applications, July 13 2005
[63] ISO27000, Sistema de Gestión de la Seguridad de la Información. http://www.iso27000.es/doc_sgsi_all.htm

2.5.3.1 ARQUITECTURA DE SEGURIDAD

Howard y LeBlanc hacen la siguiente analogía: [64] "Las aplicaciones sin una arquitectura de seguridad son como puentes construidos sin un análisis finito de elementos ni tests de túneles de viento. Seguramente, parecerán puentes, pero caerán a la primera sacudida de las alas de una mariposa. La necesidad de la seguridad de aplicaciones en forma de arquitectura de seguridad es tan grande como en la construcción de puentes o edificios. "

Los arquitectos de aplicaciones son los responsables de su construcción y diseño para cubrir los típicos riesgos tanto de uso como de ataques extremos. Los diseñadores de puentes necesitan superar cierta cantidad de coches y tráfico a pie, pero también ciclones, terremotos, fuegos, accidentes de tráfico e inundaciones. Los diseñadores de aplicaciones deben superar eventos extremos como fuerza bruta o ataques de inyección y fraude. Los riesgos de los diseñadores de aplicaciones son bien conocidos.

La seguridad ahora es algo esperado, y no un caro complemento o algo dejado de lado. La arquitectura de seguridad se refiere a los pilares fundamentales: la aplicación debe proporcionar controles para proteger la confidencialidad de la información, integridad de los datos, y proporcionar acceso a los datos cuando se requiera (disponibilidad) – y solamente a los usuarios apropiados.

"La arquitectura de seguridad ya no es un conjunto de productos de seguridad lanzados juntos y denominados como "solución".[65]

Cuando se empieza una nueva aplicación o se rediseña una aplicación existente, debe considerar cada característica funcional y tener en cuenta:

¿Son los procesos de alrededor de esta característica lo más seguro posibles? En otras palabras, ¿es este un proceso con defectos?
¿Si fuera malvado, cómo abusaría de esta característica?
¿Se requiere esta característica que este activa por defecto? Si es así, ¿existen límites u opciones que ayuden a reducir el riesgo de esta característica?

[64] Howard and LeBlanc, Writing Secure Code, 2nd Edition, pp 69 – 124, © 2003 MicrosoftPress, ISBN 0-7356-1722-8
[65] Dymaxion, Trike Methodology, http://dymaxion.org/trike/Trike_v1_Methodology_Document-draft.pdf

Andrew van der Stock llamo al proceso anterior “Thinking Evil”,[66] y recomienda ponerse en el lugar de el atacante y pensar en todas las posibles vías en que se puede abusar de cada característica, sin considerar los tres pilares básicos y usando el modelo STRIVE sucesivamente.

La arquitectura de seguridad empieza el día en que se modelan los requisitos del negocio, y no termina nunca hasta que la última copia de su aplicación es retirada. La seguridad es un proceso de larga vida y no un disparo por accidente.

2.6 ARQUITECTURA

La arquitectura de software son las técnicas formas y guías generales que en base a estas se pueden resolver problemas, se define arquitectura porque asemeja planos de una construcción (casa o edificio) y se plasma el funcionamiento la estructura e interacción entre las partes del software.

La siguiente figura muestra una arquitectura muy usada por diferentes lenguajes de programación para aplicaciones Web, que consiste en una arquitectura de 3 capas con el objetivo de separar la lógica de negocio con la lógica de diseño.

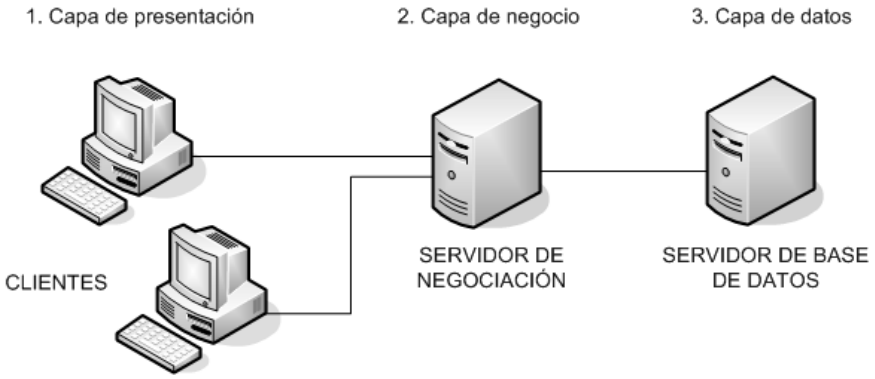

Figura 2.4 Arquitectura de Seguridad[67]

Dependiendo del lenguaje de programación el diseño de la arquitectura varia, a continuación se muestra algunos de las principales arquitecturas de las plataformas con más fuerza en el mercado para aplicaciones empresariales que son J2EE y .NET.

[66] Aspectsecurity,Security, http://www.aspectsecurity.com/press/pr_20061130.htm
[67] Wikipedia, Arquitectura de tres niveles. http://es.wikipedia.org/wiki/Arquitectura_de_tres_niveles

2.6.1 TIPOS DE ARQUITECTURAS EN JAVA J2EE

La figura 2.5 muestra la recopilación de las arquitecturas con mayor implementación siendo la plataforma J2EE la más utilizada.

JSP y Servlets

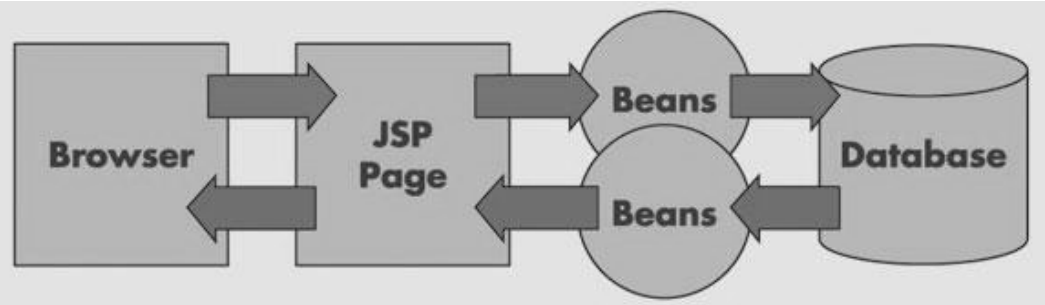

Figura 2.5 [68]

Beans sólo son clases en java que deben cumplir una estructura para la utilización y reutilización de datos. En este tipo de arquitectura los JSP llevaban tanto la lógica de negocio (operaciones a la B.D) como la lógica de presentación (lo que se presenta en el navegador).

MVC (Modelo Vista Controlador)

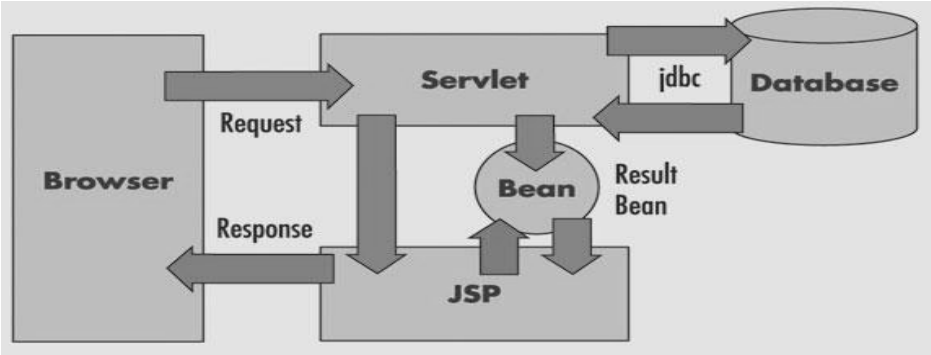

Figura 2.6 [69]

En la figura anterior se muestra como se separa la capa de presentación con la capa de negocio.

[68] Uniovi, Arquitectura Web,
http://www.di.uniovi.es/~dflanvin/docencia/dasdi/teoria/Transparencias/06.%20Arquitectura%20Web.pdf
[69] Uniovi, Arquitectura Web.
http://www.di.uniovi.es/~dflanvin/docencia/dasdi/teoria/Transparencias/06.%20Arquitectura%20Web.pdf

Beans son clases de java encargadas entre la comunicación del Servlet y el JSP. Mientras que Servlet es el encargado de realizar la lógica de negocio (operaciones a la B.D), llenar Beans y enviarlos a la capa de presentación JSP.

MVC con Strusts

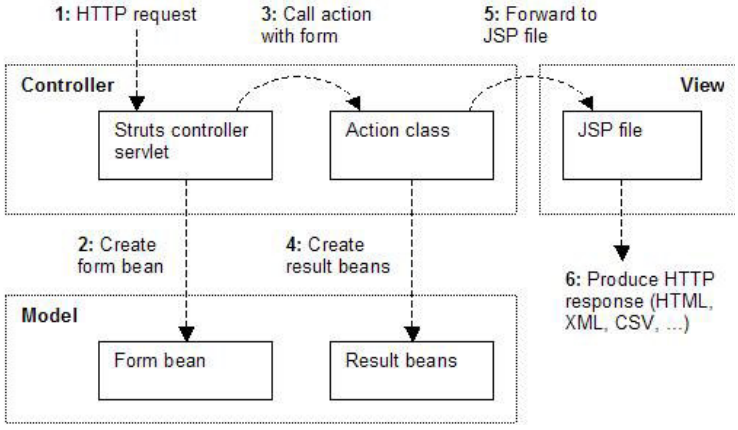

Figura 2.7 [70]

Todos los eventos de la capa de presentación se plasman en la figura 2.7 los cuales pasan por un controlador (STRUTS CONTROLER SERVLET) este se encarga de redireccionar a una clase encargada de realizar la lógica de negocio (Action) una vez que termina sus operaciones le indica al controlador la pagina que se desplegara en el navegador (JSP).

Este modelo es el más usado al realizar una aplicación Web debido a que su implementación es bastante sencilla y mantiene un orden en nuestra aplicación.

2.6.2 TIPOS DE ARQUITECTURAS EN .NET

En la figura 2.8 se muestra la arquitectura de 3 capas de .Net a continuación se describe cada una de las capaz

[70] Uniovi, Arquitectura Web.
http://www.di.uniovi.es/~dflanvin/docencia/dasdi/teoria/Transparencias/06.%20Arquitectura%20Web.pdf

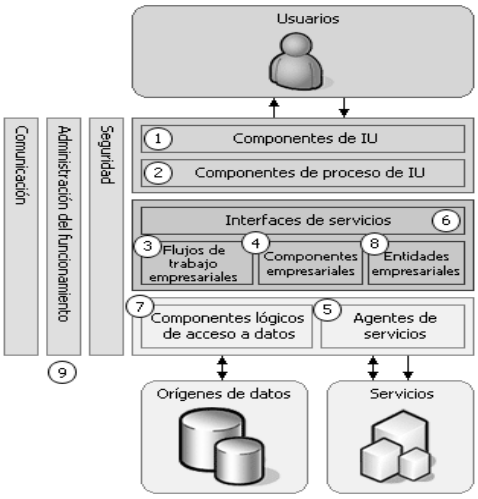

Figura 2.8 Arquitectura .Net[71]

1. Componentes de interfaz de usuario (IU): La mayor parte de las soluciones necesitan ofrecer al usuario un modo de interactuar con la aplicación. En el ejemplo de aplicación comercial, un sitio Web permite al cliente ver productos y realizar pedidos, y una aplicación basada en el entorno operativo Microsoft Windows permite a los representantes de ventas escribir los datos de los pedidos de los clientes que han telefoneado a la empresa. Las interfaces de usuario se implementan utilizando formularios de Windows Forms, páginas Microsoft ASP.NET, controles u otro tipo de tecnología que permita procesar y dar formato a los datos de los usuarios, así como adquirir y validar los datos entrantes procedentes de éstos.
2. Componentes de proceso de usuario: En un gran número de casos, la interactuación del usuario con el sistema se realiza de acuerdo a un proceso predecible. Por ejemplo, en la aplicación comercial, se puede implementar un procedimiento que permita ver los datos del producto. De este modo, el usuario puede seleccionar una categoría de una lista de categorías de productos disponibles y, a continuación, elegir uno de los productos de la categoría seleccionada para ver los detalles correspondientes. Del mismo modo, cuando el usuario realiza una compra, la interactuación sigue un proceso predecible de recolección de datos por parte del usuario, por el cual éste en primer lugar proporciona los detalles de los productos que desea adquirir, a continuación los detalles de pago y, por último, la información para el envío. Para facilitar la sincronización y organización de las interactuaciones con el usuario, resulta útil utilizar componentes de proceso de usuario individuales. De este modo, el flujo del proceso y la lógica de administración de estado no se incluye en el código de los elementos de la interfaz de usuario, por lo que varias interfaces pueden utilizar el mismo "motor" de interacción básica.
3. Flujos de trabajo empresariales: Una vez que el proceso de usuario ha recopilado los datos necesarios, éstos se pueden utilizar para realizar un proceso empresarial. Por ejemplo, tras enviar los detalles del producto, el pago y el envío a la aplicación comercial, puede comenzar

[71] Ciberaula, Arquitectura de Aplicaciones de 3 cap
as. pág. 14 (2008). http://dotnetjunkies.com/WebLog/desarrollonet/archive/2004/06/17/16855.aspx

el proceso de cobro del pago y preparación del envío. Gran parte de los procesos empresariales conllevan la realización de varios pasos, los cuales se deben organizar y llevar a cabo en un orden determinado. Por ejemplo, el sistema empresarial necesita calcular el valor total del pedido, validar la información de la tarjeta de crédito, procesar el pago de la misma y preparar el envío del producto. El tiempo que este proceso puede tardar en completarse es indeterminado, por lo que sería preciso administrar las tareas necesarias, así como los datos requeridos para llevarlas a cabo. Los flujos de trabajo empresariales definen y coordinan los procesos empresariales de varios pasos de ejecución larga y se pueden implementar utilizando herramientas de administración de procesos empresariales, como BizTalk Server Orchestration.

4. Componentes empresariales: Independientemente de si el proceso empresarial consta de un único paso o de un flujo de trabajo organizado, la aplicación requiere probablemente el uso de componentes que implementen reglas empresariales y realicen tareas empresariales. Por ejemplo, en la aplicación comercial, debe implementar una funcionalidad que calcule el precio total del pedido y agregue el costo adicional correspondiente por el envío del mismo. Los componentes empresariales implementan la lógica empresarial de la aplicación.
5. Agentes de servicios: Cuando un componente empresarial requiere el uso de la funcionalidad proporcionada por un servicio externo, tal vez sea necesario hacer uso de código para administrar la semántica de la comunicación con dicho servicio. Por ejemplo, los componentes empresariales de la aplicación comercial descrita anteriormente podría utilizar un agente de servicios para administrar la comunicación con el servicio de autorización de tarjetas de crédito y utilizar un segundo agente de servicios para controlar las conversaciones con el servicio de mensajería. Los agentes de servicios permiten aislar las idiosincrasias de las llamadas a varios servicios desde la aplicación y pueden proporcionar servicios adicionales, como la asignación básica del formato de los datos que expone el servicio al formato que requiere la aplicación.
6. Interfaces de servicios: Para exponer lógica empresarial como un servicio, es necesario crear interfaces de servicios que admitan los contratos de comunicación (comunicación basada en mensajes, formatos, protocolos, seguridad y excepciones, entre otros) que requieren los clientes. Por ejemplo, el servicio de autorización de tarjetas de crédito debe exponer una interfaz de servicios que describa la funcionalidad que ofrece el servicio, así como la semántica de comunicación requerida para llamar al mismo. Las interfaces de servicios también se denominan fachadas empresariales.
7. Componentes lógicos de acceso a datos: La mayoría de las aplicaciones y servicios necesitan obtener acceso a un almacén de datos en un momento determinado del proceso empresarial. Por ejemplo, la aplicación empresarial necesita recuperar los datos de los productos de una base de datos para mostrar al usuario los detalles de los mismos, así como insertar dicha información en la base de datos cuando un usuario realiza un pedido. Por tanto, es razonable abstraer la lógica necesaria para obtener acceso a los datos en una capa independiente de

componentes lógicos de acceso a datos, ya que de este modo se centraliza la funcionalidad de acceso a datos y se facilita la configuración y el mantenimiento de la misma.

8. Componentes de entidad empresarial: La mayoría de las aplicaciones requieren el paso de datos entre distintos componentes. Por ejemplo, en la aplicación comercial es necesario pasar una lista de productos de los componentes lógicos de acceso a datos a los componentes de la interfaz de usuario para que éste pueda visualizar dicha lista. Los datos se utilizan para representar entidades empresariales del mundo real, como productos o pedidos. Las entidades empresariales que se utilizan de forma interna en la aplicación suelen ser estructuras de datos, como conjuntos de datos, DataReader o secuencias de lenguaje de marcado extensible (XML), aunque también se pueden implementar utilizando clases orientadas a objetos personalizadas que representan entidades del mundo real necesarias para la aplicación, como productos o pedidos.
9. Componentes de seguridad, administración operativa y comunicación: La aplicación probablemente utilice también componentes para realizar la administración de excepciones, autorizar a los usuarios a que realicen tareas determinadas y comunicarse con otros servicios y aplicaciones. [72]

2.7 PATRONES DE DISEÑO

Antes de comenzar a diseñar una arquitectura se debe tener en cuenta lo que significan patrones de diseño esto es muy importante debido a que un patrón de diseño es el encargado de fortalecer problemas que se presentaban con frecuencia en situaciones particulares del diseño con el fin de proponer soluciones a estos. Por lo tanto, los patrones de diseño son soluciones exitosas a problemas comunes. Hay una gran variedad de maneras de implementar un patrón de diseño, los detalles de cómo implementar estos se le conoce como estrategia.

"Un patrón de diseño es una abstracción de una solución en un nivel alto. Los patrones solucionan problemas que existen en muchos niveles de abstracción. Hay patrones que abarcan las distintas etapas del desarrollo; desde el análisis hasta el diseño y desde la arquitectura hasta la implementación.

Muchos diseñadores y arquitectos de software han definido el término de patrón de diseño de varias formas que corresponden al ámbito a la cual se aplican los patrones. Luego, se dividió los patrones en diferentes categorías de acuerdo a su uso.

[72] Ciberaula, Arquitectura de Aplicaciones de 3 capas.
http://dotnetjunkies.com/WebLog/desarrollonet/archive/2004/06/17/16855.aspx , pág. 14 (2008).

Los diseñadores de software extendieron la idea de patrones de diseño al proceso de desarrollo de software. Debido a las características que proporcionaron los lenguajes orientados a objetos (como herencia, abstracción y encapsulamiento) les permitieron relacionar entidades de los lenguajes de programación a entidades del mundo real fácilmente, los diseñadores empezaron a aplicar esas características para crear soluciones comunes y reutilizables para problemas frecuentes que exhibían patrones similares.“[73]

Cada lenguaje de programación da a conocer los patrones de diseño así como la implementación de estos.

2.7.1 METODOLOGÍAS DE DESARROLLO DE SOFTWARE.

Al llevar un proyecto como la realización de un software es necesario llevar una metodología para poder llevar el control y tener un orden en la fabricación de un sistema, muchas veces el éxito de una aplicación depende del orden en el que se realizan las cosas, entregar a tiempo y cumplir con los planes de trabajo es una de la cualidades que no se tienen en el desarrollo de software producido en Latinoamérica comúnmente la mayoría de las empresas estiman proyectos en tiempos cortos o prometen cosas fuera del alcance, por consiguiente casi nunca se logra cumplir con los objetivos y por lo tanto se tienen desarrolladores insatisfechos y clientes insatisfechos.

La mayoría de los lideres de proyecto realiza un plan de trabajo el cual cosiste en dividir el sistema en procesos y estos asignarlos a los desarrolladores, la implementación de una metodología que se adecue a la necesidades del equipo de trabajo disminuye muchos de los problemas comunes como cambios que el cliente desea en la etapa final del proyecto, confusión en lo que el cliente quiere y lo que los desarrolladores realizan llegando a un incumplimiento de objetivos.

A continuación se describen brevemente las principales metodologías debido a que son demasiado extensas.

RUP

[73] Ciberaula, Arquitectura de Aplicaciones de 3 capas. http://java.ciberaula.com/articulo/diseno_patrones_j2ee/, pág. 16 (2008),

Rational Unified Process es una metodología para el desarrollo de software junto con UML (Lenguaje de Unificado de Modelado) para definir métodos y procesos mediante la diagramación de estos, comúnmente utilizado en lenguajes de programación orientados a objetos.
RUP básicamente cosiste en un conjunto de metodologías adaptables al desarrollo de sistemas. También se conoce por este nombre al software desarrollado por Rational, hoy propiedad de IBM, el cual incluye información entrelazada de diversos artefactos y descripciones de las diversas actividades. Está incluido en el Rational Method Composer (RMC), que permite la personalización de acuerdo a necesidades del organismo.[74]

EXTREME PROGRAMING (XP)

Metodología de desarrollo de software generalmente aplicada a proyectos de corto plazo. Como su nombre lo indica consiste en una programación rápida donde algunas de las principales características son:

- Pruebas Unitarias: Se basa en las pruebas realizadas a los principales procesos de tal manera que adelantando el resultado, se puedan hacer pruebas de las fallas que pudieran ocurrir. Es como si se pronosticaran los posibles errores.
- Simplicidad en código: Se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- Programación en pares: Una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento.[75]

MICROSOFT SOLUTION FRAMEWORK (MSF)

Es una metodología realizada por Microsoft que puede aplicarse a otros proyectos TI y no solo al desarrollo de software, es flexible, se basa en modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.[76]

SCRUM

[74] Wikipedia, Lenguaje unificado de modelado. http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado.
[75] Wikipedia, Progra,ación extrema. http://es.wikipedia.org/wiki/Programaci%C3%B3n_Extrema.
[76] Wikipedia, Microsoft Framework. http://en.wikipedia.org/wiki/Microsoft_Solutions_Framework.

Es una de las más conocidas metodologías ágiles para la gestión de proyectos. Las metodologías ágiles se centran es aspectos como la flexibilidad en la introducción de cambios y nuevos requisitos durante el proyecto, el factor humano, el producto final, la colaboración con el cliente y el desarrollo incremental como formas de asegurar los buenos resultados en proyectos con requisitos muy cambiantes o cuando se exige, como es habitual, reducir los tiempos de desarrollo manteniendo una alta calidad.

Su principal característica es potenciar la formación de equipos de trabajo autosuficientes y multidisciplinares, reduciendo la carga de gestión y proporcionando a los miembros del equipo un entorno amigable y productivo para desarrollar sus habilidades al máximo.[77]

2.8 MODELO DE RIESGO AMENAZA

Durante el diseño de su aplicación, es esencial el diseño utilizando controles evaluados de riesgo de amenaza, de otra forma se malgastan recursos, tiempo y dinero en controles inútiles y no suficiente en los riesgos reales.

El método utilizado para determinar riesgos no es tan importante como hacer modelado de riesgo de amenaza estructurado. Microsoft[78] señala que la mejora sencilla en su programa de mejora de seguridad fue la adopción universal de modelado de amenaza.

2.8.1 MODELADO DE RIESGO UTILIZANDO EL PROCESO DE MICROSOFT

Modelado de amenaza es “un proceso esencial para el desarrollo de aplicaciones Web seguras. Permite a las organizaciones determinar el control correcto y producir contramedidas efectivas dentro del presupuesto”.[79] Por ejemplo hay poco sentido en agregar un control de $100,000 a un sistema que tiene fraude insignificante.

La figura 2.9 muestra los cinco pasos en el proceso de modelado. Microsoft provee una herramienta de modelado de riesgo escrita en .NET para ayudar con el seguimiento y visualización

[77] Wikipedia, SCRUM. http://es.wikipedia.org/wiki/Scrum.
[78] Microsoft, Threat Modeling Web Applications, © 2005 Microsoft
http://msdn.microsoft.com/security/securecode/threatmodeling/default.aspx?pull=/library/enus/dnpag2/html/tmwa.asp
[79] Microsoft, Threat Modeling Web Applications, © 2005 Microsoft
http://msdn.microsoft.com/security/securecode/threatmodeling/default.aspx?pull=/library/enus/dnpag2/html/tmwa.asp

de árboles de amenazas. El uso de esta herramienta es útil para proyectos más largos y de larga vida.

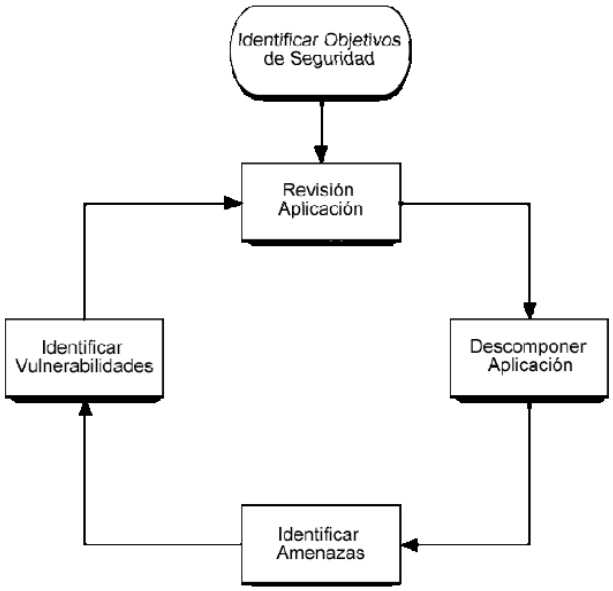

Figura 2.9 Flujo Del Modelo De Amenaza

Identificar Objetivos de Seguridad

El negocio (o líder de la organización) en coordinación con el equipo de desarrollo necesita atender los probables objetivos de seguridad. Los objetivos de seguridad en aplicaciones necesitan ser divididos en:

Identidad: ¿protege está aplicación al usuario de mal uso? ¿Hay controles adecuados para asegurar evidencia de identidad (requerido para muchas aplicaciones bancarias)?
Reputación: la pérdida de reputación derivada de la aplicación siendo mal usada o atacada Exitosamente.
Financiero: el nivel de riesgo que la organización está preparada para tomar en la remediación de potencial pérdida financiera. Un software de foros tendría menor riesgo financiero que la banca por Internet de un corporativo
Privacidad y regulaciones: en qué medida las aplicaciones deben proteger la información del usuario. Software de foros es público por naturaleza, pero un programa de impuestos esta intrínsecamente vinculado a las regulaciones y legislación de privacidad en la mayoría de los países

Disponibilidad de garantías: ¿tiene este software que estar disponible por un SLA o un acuerdo similar? ¿Es infraestructura protegida nacionalmente? ¿A qué nivel tiene que estar disponible la aplicación? Aplicaciones y técnicas altamente disponibles son extraordinariamente caras, así que la fijación de controles correctos puede ahorrar una gran cantidad de recursos y dinero.

Esto de ninguna manera es una lista exhaustiva pero da una idea de algunas de las decisiones de riesgo de negocio que lleva a la construcción de controles técnicos.

Otras fuentes de orientación vienen de:

- Leyes (Como leyes de privacidad o financieras)
- Regulaciones (como regulaciones bancarias o de negocios electrónicos)
- Estándares (como ISO 17799)
- Acuerdos Legales (como acuerdos mercantes)
- Políticas de Seguridad de la información

2.8.2 SISTEMA DE MARCACIÓN DE VULNERABILIDADES COMUNES (CVSS)

El Departamento de EEUU de Seguridad nacional (DHS por sus siglas en inglés) estableció el Grupo de Trabajo de Revelación de Vulnerabilidad NIAC, que incorpora aportaciones de Cisco, Symantec, ISS, Qualys, Microsoft, CERT/CC y eBay. Una de las aportaciones de este grupo es el Sistema de Marcación de Vulnerabilidades Comunes[80] (CVSS por sus siglas en inglés).

2.8.2.1 VENTAJAS AL UTILIZAR CVSS

El documento anexo al Sistema de Marcación contiene el siguiente ejemplo:
"Acaba de recibir una notificación de un investigador de seguridad u otra fuente que su producto tiene una vulnerabilidad, y desea asegurarse que una calificación de seguridad confiable como para alertar a sus usuarios del nivel apropiado de acción requerida cuando libere el parche. Usted es un investigador de seguridad, y ha encontrado varias vulnerabilidades en un programa. Le gustaría usar el sistema de medición CVSS para producir calificaciones de riesgo confiable, para asegurarse que el ISV tomará las vulnerabilidades en serio al ser comparadas a sus clasificaciones."

[80] CVSS, http://www.dhs.gov/interWeb/assetlibrary/NIAC_CyberVulnerabilitiesPaper_Feb05.pdf

El uso de CVSS es recomendado para usarse por departamentos del gobierno de EEUU para el trabajo en grupo – no es claro si esto es una política a la hora de escribir.

2.8.2.2 DESVENTAJAS AL UTILIZAR CVS

CVSS no encuentra o reduce el área de superficie de ataque (por ejemplo, defectos de diseño), tampoco ayuda a enumerar posibles riesgos de una pieza de código arbitrario ya que no está diseñado para ese propósito.

La clasificación de riesgos de CVSS es compleja – una hoja de cálculo es requerida para calcular el riesgo ya que el supuesto detrás de CVSS es que un riesgo simple ha sido anunciado, o un gusano o troyano ha sido liberado orientado a un pequeño número de vectores de ataque.

Los gastos de calcular la clasificación de riesgo de CVSS son bastante alto si es aplicado a una revisión de código minuciosa, que puede tener 250 o más amenazas por clasificar.

CAPÍTULO III PRINCIPALES RIESGOS EN APLICACIONES WEB

Debido a que en la actualidad los Ataques a aplicaciones Web son la principal vertiente de actividad maliciosa sobre Internet, esto debido al continuo crecimiento de este y a que el número de gente continuamente incrementa su uso para todo tipo de actividad, presenta a los atacantes un rango creciente de objetivos además de varias maneras de lanzar actividad maliciosa.

Con base a los conceptos vistos anteriormente en el presente capitulo se mencionan las estadísticas de los ataques más frecuentes que se efectúan a las aplicaciones Web, así como la forma en la que se efectúa el ataque.

Algunas estadísticas provienen de la pagina de Symantec, publicaciones del CERT (Equipo de Respuesta a Incidentes de Seguridad en Cómputo) el cual es un departamento de seguridad de computo de la UNAM que desde 1990 se encarga de proporcionar servicios de respuesta a incidentes de seguridad en computo a aquellos sitios que has sido víctimas de ataques, también se encarga de proporcionar información sobre las vulnerabilidades de seguridad y de esta forma disminuir el nivel de ataques a sitios Web.

CAPÍTULO III PRINCIPALES RIESGOS EN APLICACIONES WEB

3.1 ESTADÍSTICAS DE ATAQUES RELACIONADAS CON DETECCIÓN DE INTRUSOS

Al final del año de 1996 Dan Farmer realizo un estudio comparativo acerca de la detección de intruso, en los sistemas informáticos y Web Sites de comercio haciendo uso de técnicas sencillas, catalogando los tipos de problema de dos grupos:

- Rojo: En este grupo se encuentran los sistemas potencialmente sensibles a ataques, es decir los problemas de seguridad son conocidos por los atacantes. Por ejemplo el servicio FTP mal configurado
- Amarillo: Los ataques de este grupo son menos serios ya que el problema detectado no implica un daño serio e inmediato al sistema sin embargo esto no significa que no exista el riesgo y que no cause daños como la perdida de información, modificación, o bien robo de la misma.

En la siguiente tabla se muestran los sistemas y equipos que fueron evaluados y la categoría en la cual se asignaron, y el porcentaje de vulnerabilidad. Observarse que los sitios de noticias son mucho más vulnerables a los ataques informáticos, seguidos de los sitios gubernamentales de Estados Unidos, encontrándose estos en el grupo rojo.

Tipo de sitio	# Total sitios testeados	% Total Vulnerables	% Yellow	% Red
Bancos	660	68,34	32,73	35,61
Créditos	274	51,1	30,66	20,44
Sitios Federales US	47	61,7	23,4	38,3
News	312	69,55	30,77	38,78
Sexo	451	66,08	40,58	25,5
Totales	**1.734**	**64,93**	**33,85**	**31,08**
Grupo aleatorio	**469**	**33,05**	**15,78**	**17,27**

Figura 3.2 Porcentaje de Vulnerabilidades por tipo de Sitio[81]

3.1.1 DETECCIÓN DE RIESGOS DE LAS APLICACIONES WEB.

Haciendo un análisis profundo, son 3 los principales puntos de ataque hacia una aplicación Web, independientemente de la tecnología aplicada en su desarrollo. Estos son:

- Ataque hacia el Cliente

[81] Porcentaje de Vulnerabilidades por tipo de sitio. http://www.trouble.org/survey

- Ataque hacia el Canal de comunicación
- Ataque hacia Servidores

Estos medios, al ser los principales métodos de comunicación de una aplicación Web, siempre serán los que más sufran ataques.

A continuación se presenta, a manera de Cuadro Sinóptico, los riesgos más significativos a tomar en cuenta en el desarrollo de una aplicación Web, en cada uno de los puntos antes expuestos.

3.1.2 DETECCIÓN DE RIESGOS DEL LADO DEL CLIENTE

Esta es una lista con la clasificación de riesgos encontrados del lado del cliente, con su descripción.

RIESGO	CAUSAS	IMPACTO
Phishing (Password Harvesting Fishing) Es la capacidad de duplicar una página Web para hacer creer al visitante que se encuentra en la página original en lugar de la apropiada. Normalmente se utiliza con fines delictivos duplicando páginas Web de Bancos conocidos y enviando indiscriminadamente correos para que se acceda a esta página para actualizar las páginas de acceso al banco.	• Falta de Cultura informática • Políticas Deficientes. • Inadecuada Propaganda.	• Desprestigio del sitio Web. • Fraudes
Falta de control en el Numero de "Transacciones" Las transacciones son operaciones entre	• Transacciones limitadas.	• Robo o pérdida de la información. • Acceso a zonas no autorizadas del sistema.

componentes, un ejemplo seria una transacción cuando se modifica el estado de una ID, por esto es que en una aplicación Web se pone empeño en la operación de las transacciones, ya que estas conllevan a la falla de las Aplicaciones.		
Fraudes por ausencia de "Roles" Un rol se define como un perfil que designa el comportamiento de los usuarios que tienen acceso a la aplicación, estos se distinguen entre sí por el nivel de privilegios que poseen respecto a la Aplicación Web.	• Ausencia de Roles	• Difícil administración de la aplicación • • Cambios sin autorización en cuentas de otros usuarios.
Cross Site Scripting (CSS) Una aplicación Web puede ser usada como un mecanismo para transportar un ataque al navegador del usuario final. Un ataque exitoso puede comprometer el Token de sesión del usuario final, atacar la maquina local o enmascarar contenido para engañar al usuario.	• Validadores de entradas de Datos Deficientes	• Desprestigio de la aplicación Web.
Ataques Unicode El ataque Unicode es utilizado para forzar a los Servidores Web a salir de la raíz de la aplicación y acceder a archivos residentes en otras partes de la ruta de la aplicación Esto es	• Validadores de entradas de datos deficientes.	• Divulgación de información privada de Aplicaciones Web tales como rutas, direcciones PI, Puertos.

logrado provocando un error en el cliente.		
Inyección de Código Técnica consistente en aprovechar las debilidades que ofrece la aplicación en sus validaciones e ingresar código malicioso en sus campos de entrada permitiendo que estos se ejecuten en la aplicación Web.	• Validaciones de entradas de datos deficientes.	• Alteración del comportamiento de la aplicación, provocando: • Acceso a zonas no autorizadas del sistema Web. • Alteración del comportamiento de la aplicación

3.1.3 DETECCIÓN DE RIESGOS DEL LADO DEL SERVIDOR.

RIESGO	CAUSAS	IMPACTO
Desbordamiento del Buffer. Debilidad ligada al servidor, ya que esta tiene el control de los procesos que ejecuta la aplicación, esta condición ocurre cuando los datos escritos en la memoria exceden el tamaño reservado en el buffer y direcciones de memoria adyacentes son sobrescritas causando que la aplicación falle o termine de manera inesperada. Los atacantes suelen corromper los buffer de memoria para interrumpir el funcionamiento de la aplicación en el servidor y así denegar los	• Validaciones de entradas de los datos deficientes. • Mala Programación	• La parte que afecta directamente este tipo de ataques es el Back end de la aplicación Por ejemplo puede causar problemas en el funcionamiento de la ID.

servicios. Esto lo logran introduciendo grandes cantidades de información al sistema Web, por ejemplo, en una aplicación de Comercio Electrónico un carrito de compras con un excesivo número de Productos puede ser causa inmediata para un desbordamiento de Memoria.		
Caída del Servidor por mal Manejo de Errores. Condiciones de error que ocurren durante la operación normal que no son manejadas adecuadamente.	• Manejo inadecuado de errores.	• Si un agresor puede causar que ocurran errores que la aplicación Web no maneja, este puede obtener información detallada del sistema, denegar servicios, causar que mecanismos de seguridad fallen.
Administración de Autenticación y Sesión de Servicio Interrumpida	• Mala programación	• Fraudes en la aplicación
Acceso y Modificación a Configuraciones del Servidor.	• Administración de Configuración Insegura.	• Fallas de seguridad en el software del Servidor. • Alteraciones al Listado de Directorio o Acceso no autorizado de Directorio.
Generación de Basura dentro del Servidor.	• Administración de Configuración Insegura.	• Generación de archivos innecesarios por ejemplo, de respaldo o de ejemplo, incluyendo Scripts, Logs, aplicaciones Archivos de configuración y páginas Web.
Vulnerabilidad en los Accesos a las Propiedades	• Administración de Configuración Insegura.	• Permisos no adecuados en archivos y directorios.

de Configuración del Servidor.		
Infiltración de los Usuarios Malintencionados o no Autorizados.	• Cuentas de usuario definidas por defecto en la instalación.	• Acceso no autorizado a propiedades del Servidor final.
Perdida de Información del Servidor Web.	• Funciones administrativas o de depuración que son habilitadas o accesibles.	• Alto costo en recuperación de Información de Sistema.

3.1.4 DETECCIÓN DE RIESGOS EN EL CANAL DE COMUNICACIÓN.

RIESGO	CAUSAS	IMPACTO
Robo de Información en los Canales de Información	• Certificados SSL y opciones de encristalaron mal configurados o no habilitados.	• Alto costo en recuperación de información sensible de la aplicación que corre sobre el servidor Web.
Interceptaron de Información Sensible. Las aplicaciones Web frecuentemente utilizan funciones de criptografía para proteger información y credenciales. Estas funciones y el código que integran a ellas han sido difíciles de codificar adecuadamente, lo cual frecuentemente redunda en una protección débil	• Almacenamiento de información inseguro.	• Robo de información sensible.

3.2 ESTADÍSTICAS DE ATAQUE DDoS

El CERT (Equipo de Respuesta a Incidentes de Seguridad en Cómputo) desde 1990 hasta nuestros días, es el encargado de proveer servicios de respuesta a incidentes de seguridad en cómputo a sitios que han sido víctimas de ataques, así como publicar información respecto a

vulnerabilidades de seguridad, alertas de la misma índole y realizar investigaciones de la amplia área del computo y así ayudar a mejorar la seguridad de los sitios.

Para poder identificar las amenazas que puede correr una aplicación Web es necesario primero conocer los tipos de ataques, las formas de acceso, el objetivo del mismo y la forma en la que opera. La forma en la que afecta el ataque DDoS (Denegación de Servicios) es en bajar los servicios que deberían de estar disponibles; el ataque Corrupción de los datos consiste en modificar la información afectando así la estabilidad del negocio.

La siguiente figura detalla las fases en las que se divide un ataque, la persona que realiza el ataque, la herramienta que utiliza para realizarlo, el método, los resultados y objetivos que se espera obtener el intruso.

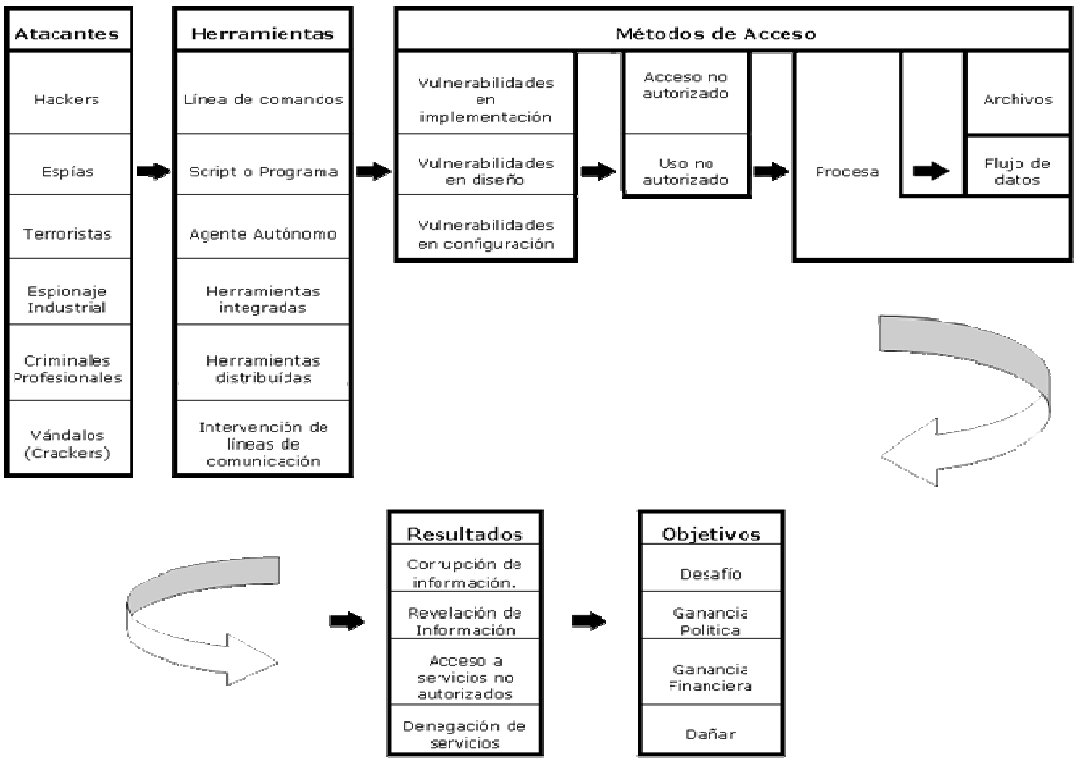

Figura 3.3 Detalles de Ataques[82]

A continuación se ejemplifica un ataque DDoS el consiste en enviar generar un ciclo infinito de peticiones al servidor, se hace más eficiente el ataque ya que cada petición se genera en un nuevo thread esta programación esta realizada en java J2SE.

[82] Detalle de Ataques, Howard Jhon D., Tesis An Analysis Of Security on the Internet 1989-1995

```
public static void main(String args[]) {
      Prueba p = new Prueba();
      String params = " http://localhost:8080/MyApp?user=##&password=$$";
      while (true) {
         params = params.replaceAll("##", "user" + ((int) Math.random()) * (100));
         params = params.replaceAll("$$", "user" + ((int) Math.random()) * (100));
         p.ejecutaPeticiones(params);
      }

   }
   public void ejecutaPeticiones(final String url) {
      Thread run = new Thread() {

         int i = 1000;

         public void run() {
            while (true) {
               try {
                  URL miAplicacion = new URL(url);
                  URLConnection yahooConnection = miAplicacion.openConnection();
                  DataInputStream dis = new
DataInputStream(yahooConnection.getInputStream());
                  String inputLine;
                  while ((inputLine = dis.readUTF()) != null) {
                     System.out.println(inputLine);
                  }
                  dis.close();
               } catch (MalformedURLException me) {
                  System.out.println("MalformedURLException: " + me);
               } catch (IOException ioe) {
                  System.out.println("IOException: " + ioe);
               }

            }
         }
      };
      run.start();
   }
```

Tabla 3.3. Ejemplo de Ataque DDoS.

El ataque DDoS afecta a todo tipo de plataformas, una forma de evitar este tipo de ataques es monitoreando las ips que realizan la petición a nuestra aplicación si estas sobrepasan de un número de conexiones http en rango de tiempo corto es considerado un ataque.

El código malicioso es un ataque informático que requiere de la intervención del usuario para propagas, en el informe de Amenazas de Seguridad en Internet de Symantec publicado en febrero del 2005 latinoamerica es una de las regiones más afectadas con Gaobot, Spybot y Netsky.P. La forma de propagación de este último consiste en un envío masivo de si mismo usando una extensión .zip ya que de esta forma puede evadir los filtros de seguridad ya que archivos con este tipo de extensión generalmente son confiables y por tal motivo el usuario final abre el archivo provocando así la propagación del virus.

El siguiente ejemplo muestra cuando un usuario inserta su usesrname y en vez de esto inserta un comando que originalmente el programador pensó al realizar el programa.

```
<%
   If Not IsEmpty(Request( "username" ) ) Then
      Const ForReading = 1, ForWriting = 2, ForAppending = 8
      Dim fso, f
      Set fso = CreateObject("Scripting.FileSystemObject")
      Set f = fso.OpenTextFile(Server.MapPath( "userlog.txt" ), ForAppending, True)
      f.Write Request("username") & vbCrLf
      f.close
      Set f = nothing
      Set fso = Nothing
      %
       <h1List of logged users:</h1
       <pre
      <%
       Server.Execute( "userlog.txt" )
      %
       </pre
      <%
   Else
      %
       <form
       <input name="username" /<input type="submit" name="submit" /
       </form
      <%
   End If
%
```

Tabla 3.4. Ejemplo de Uso de Code Injection[83]

A continuación se muestra un ejemplo en PHP, en el cual el programador solo pensó en descargar el programa laptop.php o el programa desktop.php, pero cualquiera puede insertar cualquier otro programa:

```
<?php
   $color = 'blue';
   if (__isset( $_GET['SALES'] ) )
      $color = $_GET['SALES'];
   require( $sales . '.php' );
?>
<form method="get">
   <select name="SALES">
      <option value="laptop">laptop</option>
      <option value="desktop">desktop</option>
   </select>
   <input type="submit">
</form>
```

Tabla 3.5. Ejemplo de Inyección de Archivos[84]

Todas las plataformas pueden ser vulnerables de ser atacadas con code injection, desde el HTML, hasta el shell de un sistema operativo. A continuación se describen algunas de las principales plataformas para el desarrollo de aplicaciones Web.

"ASP.NET no contiene ninguna función que para incluir código inyectado, pero puede hacerlo a través del uso de las clases CodeProvider junto con la reflexión. Cualquier código PHP que utilice

[83] OWASP, Code Injection Examples. http://www.owasp.org/index.php/Direct_Dynamic_Code_Evaluation_('Eval_Injection')
[84] OWASP, Una Guía para Construir Aplicaciones y Servicios Web Seguros. 2005, pag 196

la función eval() corre el riesgo de sufrir un ataque de inyección de código. Java generalmente no brinda la habilidad para evaluar JSP's dinámicas"[85].

Sin embargo existen dos excepciones en este rubro:

- Inclusión Dinámica de JSP (<jsp:include …>)
- Utilizar etiquetas de evaluación de JSP's proporcionadas por terceros

Portales y software desarrollado por comunidades a menudo requieren validación de Código dinámico en las plantillas y temas del sitio intercambiables. Si el portal requiere inclusiones dinámica y ejecución de código dinámico, hay un riesgo de inyección de código Java o JSP.
Para combatir estos, la primera línea de defensa la integran:

- Preferir siempre inclusiones estáticas (<%include%>)
- Restringir la inclusión de archivos externos al servidor utilizando las políticas de seguridad de Java 2.
- Establecer reglas de cortafuegos para prevenir conexiones fuera de los límites de Internet.
- Asegurar que el código no interprete información proporcionada sin haberla validado.

En un ejemplo hipotético, el usuario puede seleccionar el uso de "X" como tema inicial. En este ejemplo, el código incluye dinámicamente un archivo llamado "X.tema.jsp" utilizando concatenación simple. Sin embargo, si el usuario ingresa otro tipo de información, puede tener la facultad de obtener código Java interpretado en el servidor. Bajo este escenario, el servidor de aplicaciones ya no es propiedad del usuario. En general, la inclusión dinámica y la evaluación dinámica de código debe ser mal vista dentro de los programas y se debe evitar lo más posible.

3.3 ESTADÍSTICAS ACERCA DE CODE INJECTION

La inyección de datos es común entre los individuos que gustan por atacar las páginas Web, generalmente usan esta técnica para obtener otra información que les ayude a atacar en mayor grado los sistemas.

Algunos de los tipos de inyección de datos buscan modificar los datos de la base de datos, este tipo de inyección se llama SQL Injection. El impacto de este tipo de inyección puede ser desde atacar el sistema Web hasta la perdida de información sensible a la empresa.

3.3.1 INYECCIÓN DE DATOS MALICIOSA

- Instalación de malware en alguna computadora por medio de la inyección de código a un navegador Web o en sus plug ins.
- Instalación de malware inyectando código a sistemas Web desarrollados en cualquiera de las plataformas disponibles.
- La inyección de código puede ser usada por medio del shell del sistema operativo, para obtener mayores privilegios de los permitidos.
- Robo de sesiones desde el navegador Web usando inyección HTML/Script (Cross-site scripting).

3.3.2 INYECCIÓN DE DATOS BENÉFICA

Así como hay inyección de código maliciosa, también hay inyección de código benéfica para el programador, por ejemplo, se puede modificar una tabla de la base de datos de un sistema existente usando la inyección de datos. Básicamente la inyección de código benéfica es útil para modificar el sistema de alguna manera eficiente y con menores costos.

3.3.3 INYECCIÓN DE CÓDIGO INESPERADA

Existe también la inyección de código inesperada, que es cuando el usuario ingresa caracteres inválidos en el sistema lo cual puede ocasionar que este funciona indebidamente con comportamientos inesperados, es por eso que se recomienda que se tenga un control de caracteres en los campos donde se requiere que el usuario ingrese datos al sistema.

Por otro lado de acuerdo a la siguiente figura algunas de las técnicas comunes usadas por atacantes para suplantar un sitio Web incluyen explotar una aplicación Web vulnerable corriendo en el servidor. (Atacando a través de campos de entrada correctamente validados). O explotando alguna vulnerabilidad presente. De acuerdo a la siguiente figura, tan solo en 2008, hubo 12885 vulnerabilidades específicas de sitio.

Dentro de esta actividad, y de acuerdo a un reporte de Symantec,[86] indica que la mayoría de ataques basados en Web son lanzados contra usuarios que visitan sitios Web legítimos, que han sido corrompidos por atacantes para suplantar contenido malicioso.

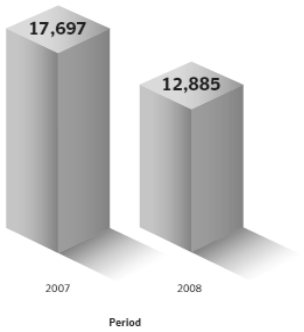

Figura 3.1. Vulnerabilidades especificas en Sitio. [87]

En el caso de sitios populares y confiables, con gran número de visitantes, estos pueden infectar a miles de equipos en un ataque simple. Por ejemplo, un ataque que tenga por destino un sitio Web gubernamental de EEUU o Reino Unido, infectaría a miles de usuarios de esos sitios. Este tipo de ataques proporciona un "océano de posibilidades" para distribuir código malicioso porque su destino está orientado a sitios con una tasa de Trafico Masiva, como lo pueden ser sitios de organizaciones respetables.

Con el propósito de infectar el más grande número posible de sitios Web con un sencillo mecanismo, los atacantes intentarán atacar un tipo específico de clase buscando vulnerabilidades comunes en los sitios y generalmente automatizando el descubrimiento y exploración. Esto permite a los atacantes infectar sitios Web con la eficiencia comúnmente encontrada en Gusanos de Internet.

Conforme la distancia se acorta entre procesos de Negocio y tecnología, el impacto en Seguridad y Riesgos aumenta, no por ello indicando que los nuevos modelos de sistemas presenten mas vulnerabilidades que beneficios, simplemente se debe contemplar que nuevas implicaciones que generan estos nuevos paradigmas, y encontrar soluciones acorde a las nuevas necesidades.

[86] Symantec, Whitepaper Symantec Internet Threat report , http://www.symantec.com/specprog/threatreport/ent-whitepaper_symantec_internet_security_threat_report_x_09_2006.en-us.pdf

[87] Xssed, Xssed Project , http://www.xssed.com/news/2008/

Los riesgos que se presentan en los Sistemas Web vienen de ambos lados involucrados. Cliente - Servidor, es decir, el riesgo puede ser del sistema generado al usuario o del usuario generado al Sistema. El primero consiste en corromper el servicio que se le está prestando al usuario, esto con fines fraudulentos, para poder obtener alguna ventaja sobre él. El segundo es algún tipo de vulnerabilidad que presenta el sistema y que el usuario o intruso aprovecha para corromper al sistema. Estos riesgos siempre representaran un peligro potente sobre las partes involucradas y conllevan siempre una potencial pérdida económica para las Empresas.

CAPÍTULO IV LEGISLACIÓN INFORMÁTICA APLICADA AL DESARROLLO WEB

En México existe una legislación informática que comparada con otros países resulta muy pobre para la problemática actual. El incremento de los delitos informáticos se debe en gran medida a que no hay una legislación adecuada y existen muchos huecos en ella; México tiene como reto crear leyes para los delitos informáticos actuales. En este capítulo se mencionan las leyes mexicanas existentes relacionadas con la informática así como el uso de terceros.

Existen mejores prácticas para llevar a cabo las actividades relacionadas con TI, en este capítulo se mencionan algunas de las relacionadas con las aplicaciones Web y el uso de terceros.

CAPÍTULO IV LEGISLACIÓN INFORMÁTICA APLICADA AL DESARROLLO WEB

4.1 LEGISLACIÓN INFORMÁTICA EN MÉXICO

Lo que se conoce como derecho nace como un medio para regular la conducta del hombre en sociedad. La conducta ha evolucionando de la mano con las tecnologías por lo que las leyes deben adaptarse para regular también lo concerniente a la informática.

El derecho regula la conducta y a la sociedad a través de leyes. El proceso de creación e inserción de estas leyes a la vida cotidiana es un proceso largo y lento porque hay ignorancia sobre el tema en los encargados de crear las leyes así como una falta de interés en el tema, lo que ha llevado al país a permanecer rezagados en la legislación informática.

Antes de proseguir con este capítulo se explica brevemente el marco jurídico de México.

Artículo 40 de la CONSTITUCIÓN POLÍTICA DE LOS ESTADOS UNIDOS MEXICANOS:
"Es voluntad del pueblo mexicano constituirse en una República representativa, democrática, federal, compuesta de Estados libres y soberanos en todo lo concerniente a su régimen interior; pero unidos en una federación establecida según los principios de esta ley fundamental."[88]

El Poder legislativo, se deposita en un Congreso Federal, el cual tiene facultades exclusivas para legislar sobre: hidrocarburos, minería, industria cinematográfica, comercio, juegos con apuestas y sorteos, intermediación y servicios financieros, energía eléctrica y nuclear, derecho marítimo, ciudadanía, migración, vías generales de comunicación, correos, aguas, moneda, delitos federales, coordinación en materia de seguridad pública, fiscalización superior de la federación, leyes del trabajo reglamentarias del artículo 123 Constitucional, entre otras.

Lamentablemente es en el congreso donde no se ha profundizado en legislar sobre informática lo que genera que huecos legales en cuestiones de delitos informáticos.

4.2 DELITOS INFORMÁTICOS

[88] CONSTITUCIÓN POLÍTICA DE LOS ESTADOS UNIDOS MEXICANOS, Constitución publicada en el Diario Oficial de la Federación el 5 de febrero de 1917

La conducta que tiene como instrumento o fin computadoras u otros bienes informáticos y que lesionan o dañan bienes, intereses o derechos de personas físicas o morales.
Los principales delitos informáticos son:

- Intervención de correo electrónico
- Acceso no autorizado a sistemas o servicios
- Reproducción no autorizada de programas informáticos.
- Uso no autorizado de programas y de datos.

4.2.1 INTERVENCIÓN DE CORREO ELECTRÓNICO

Este delito atenta contra la privacidad como derecho fundamental de las personas y es penalizado según el Código Penal Federal (Publicado en el Diario Oficial de la Federación el 14 de agosto de 1931), por medio de aplicaciones Web sin los controles mínimos se puede intervenir la información, a continuación los detalles.

TITULO QUINTO **Delitos en Materia de Vías de Comunicación y Correspondencia**
CAPÍTULO II Violación de correspondencia
Artículo 173
Se aplicarán de tres a ciento ochenta jornadas de trabajo en favor de la comunidad:
I.- Al que abra indebidamente una comunicación escrita que no esté dirigida a él, y
II.- Al que indebidamente intercepte una comunicación escrita que no esté dirigida a él, aunque la conserve cerrada y no se imponga de su contenido.
Los delitos previstos en este artículo se perseguirán por querella.

Artículo 174
No se considera que obren delictuosamente los padres que abran o intercepten las comunicaciones escritas dirigidas a sus hijos menores de edad, y los tutores respecto de las personas que se hallen bajo su dependencia, y los cónyuges entre sí.

TITULO NOVENO
Revelación de secretos y acceso ilícito a **sistemas y equipos de informática**

4.2.2 ACCESO NO AUTORIZADO A SISTEMAS O SERVICIOS

Según el Código Penal Federal (Publicado en el Diario Oficial de la Federación el 14) se penaliza al o los individuos que acceden sin autorización a servicios o sistemas, en este punto se incluyen las aplicaciones Web que es el tema en el cual se enfoca este trabajo.
En la legislación existente sobre el acceso no autorizado no especifica a detalle la amplia gama de servicios informáticos que pueden ser víctimas de accesos no autorizados y la interpretación de dicha ley puede ser interpretada erróneamente si no se conocen los conceptos básicos de informática.

Artículo 211 bis 1

Al que sin autorización modifique, destruya o provoque pérdida de información contenida en sistemas o equipos de informática protegidos por algún mecanismo de seguridad, se le impondrán de seis meses a dos años de prisión y de cien a trescientos días multa.

Al que sin autorización conozca o copie información contenida en sistemas o equipos de informática protegidos por algún mecanismo de seguridad, se le impondrán de tres meses a un año de prisión y de cincuenta a ciento cincuenta días multa.

Artículo 211 bis 2

Al que sin autorización modifique, destruya o provoque pérdida de información contenida en sistemas o equipos de informática del Estado, protegidos por algún mecanismo de seguridad, se le impondrán de uno a cuatro años de prisión y de doscientos a seiscientos días multa.

Al que sin autorización conozca o copie información contenida en sistemas o equipos de informática del Estado, protegidos por algún mecanismo de seguridad, se le impondrán de seis meses a dos años de prisión y de cien a trescientos días multa.

Artículo 211 bis 3

Al que estando autorizado para acceder a sistemas y equipos de informática del Estado, indebidamente modifique, destruya o provoque pérdida de información que contengan, se le impondrán de dos a ocho años de prisión y de trescientos a novecientos días multa.

Al que estando autorizado para acceder a sistemas y equipos de informática del Estado, indebidamente copie información que contengan, se le impondrán de uno a cuatro años de prisión y de ciento cincuenta a cuatrocientos cincuenta días multa.

Artículo 211 bis 4

Al que sin autorización modifique, destruya o provoque pérdida de información contenida en sistemas o equipos de informática de las instituciones que integran el sistema financiero, protegidos por algún mecanismo de seguridad, se le impondrán de seis meses a cuatro años de prisión y de cien a seiscientos días multa.

Al que sin autorización conozca o copie información contenida en sistemas o equipos de informática de las instituciones que integran el sistema financiero, protegidos por algún mecanismo de seguridad, se le impondrán de tres meses a dos años de prisión y de cincuenta a trescientos días multa.

Artículo 211 bis 5

Al que estando autorizado para acceder a sistemas y equipos de informática de las instituciones que integran el sistema financiero, indebidamente modifique, destruya o provoque pérdida de información que contengan, se le impondrán de seis meses a cuatro años de prisión y de cien a seiscientos días multa.

Al que estando autorizado para acceder a sistemas y equipos de informática de las instituciones que integran el sistema financiero, indebidamente copie información que contengan, se le impondrán de tres meses a dos años de prisión y de cincuenta a trescientos días multa.
Las penas previstas en este artículo se incrementarán en una mitad cuando las conductas sean cometidas por funcionarios o empleados de las instituciones que integran el sistema financiero.

Artículo 211 bis 6

Para los efectos de los artículos 211 Bis 4 y 211 Bis 5 anteriores, se entiende por instituciones que integran el sistema financiero, las señaladas en el artículo 400 Bis de este Código.

Artículo 211 bis 7

Las penas previstas en este capítulo se aumentarán hasta en una mitad cuando la información obtenida se utilice en provecho propio o ajeno.

4.2.3 REPRODUCCIÓN NO AUTORIZADA DE PROGRAMAS INFORMÁTICOS.

El uso no autorizado de programas afecta a las aplicaciones WEB cuando un trabajador hace uso de un sistema desarrollado en una empresa sin autorización y pretende venderlo a otra compañía,

si no se protege el sistema en base a los derechos de autor se puede lucrar con el software que es propiedad de alguna empresa, es importante que cuando se hagan contratos con terceros se especifiquen cláusulas que hablen sobre la propiedad del software desarrollado para una empresa o persona, así mismo algunas empresas incluyen este tipo de leyes en los contratos de su personal; es decir que los sistemas desarrollados por un empleado para alguna empresa son propiedad de la empresa y no del empleado.

LEY FEDERAL DEL DERECHO DE AUTOR

(Publicada en el Diario Oficial de la Federación el 26 de diciembre de 1996)

TITULO SEGUNDO Del Derecho de Autor

CAPÍTULO I Reglas generales

Artículo 13

Los derechos de autor a que se refiere esta Ley se reconocen respecto de las obras de las siguientes ramas:

XI. Programas de cómputo;

XIV. De compilación, integrada por las colecciones de obras, tales como las enciclopedias, las antologías, y de obras u otros elementos como las bases de datos, siempre que dichas colecciones, por su selección o la disposición de su contenido o materias, constituyan una creación intelectual. Las demás obras que por analogía puedan considerarse obras literarias o artísticas se incluirán en la rama que les sea más afín a su naturaleza.

TITULO QUINTO De los Derechos Conexos

CAPÍTULO IV **De los Programas de Computación y las Bases de Datos**

Artículo 101

Se entiende por programa de computación la expresión original en cualquier forma, lenguaje o código, de un conjunto de instrucciones que, con una secuencia, estructura y organización determinada, tiene como propósito que una computadora o dispositivo realice una tarea o función específica.

Artículo 101

Se entiende por programa de computación la expresión original en cualquier forma, lenguaje o código, de un conjunto de instrucciones que, con una secuencia, estructura y organización determinada, tiene como propósito que una computadora o dispositivo realice una tarea o función específica.

Artículo 102

Los programas de computación se protegen en los mismos términos que las obras literarias. Dicha protección se extiende tanto a los programas operativos como a los programas aplicativos, ya sea en forma de código fuente o de código objeto. Se exceptúan aquellos programas de cómputo que tengan por objeto causar efectos nocivos a otros programas o equipos.

Artículo 103

Salvo pacto en contrario, los derechos patrimoniales sobre un programa de computación y su documentación, cuando hayan sido creados por uno o varios empleados en el ejercicio de sus funciones o siguiendo las instrucciones del empleador, corresponden a éste.

Como excepción a lo previsto por el artículo 33 de la presente Ley, el plazo de la cesión de derechos en materia de programas de computación no está sujeto a limitación alguna.

Artículo 104

Como excepción a lo previsto en el artículo 27 fracción IV, el titular de los derechos de autor sobre un programa de computación o sobre una base de datos conservará, aún después de la venta de ejemplares de los mismos, el derecho de autorizar o prohibir el arrendamiento de dichos ejemplares. Este precepto no se aplicará cuando el ejemplar del programa de computación no constituya en sí mismo un objeto esencial de la licencia de uso.

Artículo 105

El usuario legítimo de un programa de computación podrá realizar el número de copias que le autorice la licencia concedida por el titular de los derechos de autor, o una sola copia de dicho programa siempre y cuando:

I. Sea indispensable para la utilización del programa, o

II. Sea destinada exclusivamente como resguardo para sustituir la copia legítimamente adquirida, cuando ésta no pueda utilizarse por daño o pérdida. La copia de respaldo deberá ser destruida cuando cese el derecho del usuario para utilizar el programa de computación.

Artículo 106

El derecho patrimonial sobre un programa de computación comprende la facultad de autorizar o prohibir:

I. La reproducción permanente o provisional del programa en todo o en parte, por cualquier medio y forma;

II. La traducción, la adaptación, el arreglo o cualquier otra modificación de un programa y la reproducción del programa resultante;
III. Cualquier forma de distribución del programa o de una copia del mismo, incluido el alquiler, y
IV. La descompilación, los procesos para revertir la ingeniería de un programa de computación y el desensamblaje.

Artículo 107
Las bases de datos o de otros materiales legibles por medio de máquinas o en otra forma, que por razones de selección y disposición de su contenido constituyan creaciones intelectuales, quedarán protegidas como compilaciones. Dicha protección no se extenderá a los datos y materiales en sí mismos.

Artículo 108
Las bases de datos que no sean originales quedan, sin embargo, protegidas en su uso exclusivo por quien las haya elaborado, durante un lapso de 5 años.

Artículo 109
El acceso a información de carácter privado relativo a las personas, contenida en las bases de datos a las que se refiere el artículo anterior, así como la publicación, reproducción, divulgación, comunicación pública y transmisión de dicha información, requerirá la autorización previa de las personas de que se trate.

Quedan exceptuados de lo anterior, las investigaciones de las autoridades encargadas de la procuración e impartición de justicia, de acuerdo con la legislación respectiva, así como el acceso a archivos públicos por las personas autorizadas por la ley, siempre que la consulta sea realizada conforme a los procedimientos respectivos.

Artículo 110
El titular del derecho patrimonial sobre una base de datos tendrá el derecho exclusivo, respecto de la forma de expresión de la estructura de dicha base, de autorizar o prohibir:
I. Su reproducción permanente o temporal, total o parcial, por cualquier medio y de cualquier forma;
II. Su traducción, adaptación, reordenación y cualquier otra modificación;
III. La distribución del original o copias de la base de datos;
IV. La comunicación al público, y

V. La reproducción, distribución o comunicación pública de los resultados de las operaciones mencionadas en la fracción II del presente artículo.

Artículo 111

Los programas efectuados electrónicamente que contengan elementos visuales, sonoros, tridimensionales o animados quedan protegidos por esta Ley en los elementos primigenios que contengan.

Artículo 112

Queda prohibida la importación, fabricación, distribución y utilización de aparatos o la prestación de servicios destinados a eliminar la protección técnica de los programas de cómputo, de las transmisiones a través del espectro electromagnético y de redes de telecomunicaciones y de los programas de elementos electrónicos señalados en el artículo anterior.

Artículo 113

Las obras e interpretaciones o ejecuciones transmitidas por medios electrónicos a través del espectro electromagnético y de redes de telecomunicaciones y el resultado que se obtenga de esta transmisión estarán protegidas por esta Ley.

4.2.4 USO DE PROGRAMAS Y DE DATOS CON O SIN AUTORIZACIÓN.

Se puede penalizar a personas que sin autorización acceden a sistemas o datos, pero también se puede penalizar a personas que acceden a sistemas excediendo los privilegios que se le dieron. Esto afecta a las aplicaciones WEB cuando un atacante vulnera la seguridad de la misma accediendo a los sistemas e incluso a información sensible de la empresa ya que cada vez más las empresas usan aplicaciones WEB por su portabilidad, sistemas heterogéneos y por la distribución geográfica de sus instalaciones.

CÓDIGO PENAL FEDERAL

(Publicado en el Diario Oficial de la Federación el 14 de agosto de 1931)

TITULO NOVENO **Revelación de secretos y acceso ilícito a sistemas y equipos de informática**

CAPÍTULO I Revelación de secretos

Artículo 210

Se impondrán de treinta a doscientas jornadas de trabajo en favor de la comunidad, al que sin justa causa, con perjuicio de alguien y sin consentimiento del que pueda resultar perjudicado, revele algún secreto o comunicación reservada que conoce o ha recibido con motivo de su empleo, cargo o puesto.

Artículo 211

La sanción será de uno a cinco años, multa de cincuenta a quinientos pesos y suspensión de profesión en su caso, de dos meses a un año, cuando la revelación punible sea hecha por persona que presta servicios profesionales o técnicos o por funcionario o empleado público o cuando el secreto revelado o publicado sea de carácter industrial.

Artículo 211 Bis

A quien revele, divulgue o utilice indebidamente o en perjuicio de otro, información o imágenes obtenidas en una intervención de comunicación privada, se le aplicarán sanciones de seis a doce años de prisión y de trescientos a seiscientos días multa.

Capítulo II **Acceso ilícito a sistemas y equipos de informática**

Artículo 211 bis 1

Al que sin autorización modifique, destruya o provoque pérdida de información contenida en sistemas o equipos de informática protegidos por algún mecanismo de seguridad, se le impondrán de seis meses a dos años de prisión y de cien a trescientos días multa.

Al que sin autorización conozca o copie información contenida en sistemas o equipos de informática protegidos por algún mecanismo de seguridad, se le impondrán de tres meses a un año de prisión y de cincuenta a ciento cincuenta días multa.

Artículo 211 bis 2

Al que sin autorización modifique, destruya o provoque pérdida de información contenida en sistemas o equipos de informática del Estado, protegidos por algún mecanismo de seguridad, se le impondrán de uno a cuatro años de prisión y de doscientos a seiscientos días multa.

Al que sin autorización conozca o copie información contenida en sistemas o equipos de informática del Estado, protegidos por algún mecanismo de seguridad, se le impondrán de seis meses a dos años de prisión y de cien a trescientos días multa.

Artículo 211 bis 3

Al que estando autorizado para acceder a sistemas y equipos de informática del Estado, indebidamente modifique, destruya o provoque pérdida de información que contengan, se le impondrán de dos a ocho años de prisión y de trescientos a novecientos días multa. Al que estando autorizado para acceder a sistemas y equipos de informática del Estado, indebidamente copie información que contengan, se le impondrán de uno a cuatro años de prisión y de ciento cincuenta a cuatrocientos cincuenta días multa.

Artículo 211 bis 4

Al que sin autorización modifique, destruya o provoque pérdida de información contenida en sistemas o equipos de informática de las instituciones que integran el sistema financiero, protegidos por algún mecanismo de seguridad, se le impondrán de seis meses a cuatro años de prisión y de cien a seiscientos días multa.

Al que sin autorización conozca o copie información contenida en sistemas o equipos de informática de las instituciones que integran el sistema financiero, protegidos por algún mecanismo de seguridad, se le impondrán de tres meses a dos años de prisión y de cincuenta a trescientos días multa.

Artículo 211 bis 5

Al que estando autorizado para acceder a sistemas y equipos de informática de las instituciones que integran el sistema financiero, indebidamente modifique, destruya o provoque pérdida de información que contengan, se le impondrán de seis meses a cuatro años de prisión y de cien a seiscientos días multa.

Al que estando autorizado para acceder a sistemas y equipos de informática de las instituciones que integran el sistema financiero, indebidamente copie información que contengan, se le impondrán de tres meses a dos años de prisión y de cincuenta a trescientos días multa.

Las penas previstas en este artículo se incrementarán en una mitad cuando las conductas sean cometidas por funcionarios o empleados de las instituciones que integran el sistema financiero.

Artículo 211 bis 6

Para los efectos de los artículos 211 Bis 4 y 211 Bis 5 anteriores, se entiende por instituciones que integran el sistema financiero, las señaladas en el artículo 400 Bis de este Código.

Artículo 211 bis 7

Las penas previstas en este capítulo se aumentarán hasta en una mitad cuando la información obtenida se utilice en provecho propio o ajeno.

4.3 LEGISLACIÓN APLICADA AL OUTSOURCING

En México el uso de terceros (Outsourcing) se ha venido dando desde hace varios años, las empresas contratan a terceros por medio de Outsourcing para agilizar sus procesos y delegar ciertas tareas a empresas especializadas en realizar dicha actividad.

Este capítulo expone temas relacionados con la contratación de terceros que realizan aplicaciones Web. Las empresas tiene que tener cuidado al hacer uso de dichas empresas ya que pueden exponer información sensible u obtener como resultado una aplicación poco que segura que de igual forma exponga la operación del negocio. Es importante que se tenga clara la legislación existen respecto al uso de Outsourcing cuando se haga uso de ellos. A continuación se explican detalles sobre la legislación que aplica al uso de terceros o intermediarios, como en la LEY FEDERAL DEL TRABAJO.

Según la LEY FEDERAL DEL TRABAJO, Nueva Ley publicada en el Diario Oficial de la Federación el 1º de abril de 1970, TEXTO VIGENTE Última reforma publicada DOF 17-01-2006[89] un intermediario es la persona que contrata o interviene en la contratación de otra u otras para que presten servicios a un patrón, según su **artículo 12.**

Artículo 13.- No serán considerados intermediarios, sino patrones, las empresas establecidas que contraten trabajos para ejecutarlos con elementos propios suficientes para cumplir las obligaciones que deriven de las relaciones con sus trabajadores. En caso contrario serán solidariamente responsables con los beneficiarios directos de las obras o servicios, por las obligaciones contraídas con los trabajadores

Artículo 14.- Las personas que utilicen intermediarios para la contratación de trabajadores serán responsables de las obligaciones que deriven de esta Ley y de los servicios prestados.
Los trabajadores tendrán los derechos siguientes:
I. Prestarán sus servicios en las mismas condiciones de trabajo y tendrán los mismos derechos que correspondan a los trabajadores que ejecuten trabajos similares en la empresa o establecimiento; y

[89] LEY FEDERAL DEL TRABAJO, Última Reforma DOF 17-01-2006.

II. Los intermediarios no podrán recibir ninguna retribución o comisión con cargo a los salarios de los trabajadores.

Artículo 15.- En las empresas que ejecuten obras o servicios en forma exclusiva o principal para otra, y que no dispongan de elementos propios suficientes de conformidad con lo dispuesto en el Artículo 13, se observarán las normas siguientes:

I. La empresa beneficiaria será solidariamente responsable de las obligaciones contraídas con los trabajadores; y II. Los trabajadores empleados en la ejecución de las obras o servicios tendrán derecho a disfrutar de condiciones de trabajo proporcionadas a las que disfruten los trabajadores que ejecuten trabajos similares en la empresa beneficiaria. Para determinar la proporción, se tomarán en consideración las diferencias que existan en los salarios mínimos que rijan en el área geográfica de aplicación en que se encuentren instaladas las empresas y las demás circunstancias que puedan influir en las condiciones de trabajo.

Artículo 20.- Se entiende por relación de trabajo, cualquiera que sea el acto que le dé origen, la prestación de un trabajo personal subordinado a una persona, mediante el pago de un salario. Contrato individual de trabajo, cualquiera que sea su forma o denominación, es aquel por virtud del cual una persona se obliga a prestar a otra un trabajo personal subordinado, mediante el pago de un salario. La prestación de un trabajo a que se refiere el párrafo primero y el contrato celebrado producen los mismos efectos.

Artículo 24.- Las condiciones de trabajo deben hacerse constar por escrito cuando no existan contratos colectivos aplicables. Se harán dos ejemplares, por lo menos, de los cuales quedará uno en poder de cada parte.

Artículo 25.- El escrito en que consten las condiciones de trabajo deberá contener:
I. Nombre, nacionalidad, edad, sexo, estado civil y domicilio del trabajador y del patrón;
II. Si la relación de trabajo es por obra o tiempo determinado o tiempo indeterminado;
III. El servicio o servicios que deban prestarse, los que se determinarán con la mayor precisión posible;
IV. El lugar o los lugares donde debe prestarse el trabajo;
V. La duración de la jornada;
VI. La forma y el monto del salario;
VII. El día y el lugar de pago del salario;

VIII. La indicación de que el trabajador será capacitado o adiestrado en los términos de los planes y programas establecidos o que se establezcan en la empresa, conforme a lo dispuesto en esta Ley; y IX. Otras condiciones de trabajo, tales como días de descanso, vacaciones y demás que convengan el trabajador y el patrón.

Se tiene que tener cuidado al momento de realizar un contrato con un Outsourcing, especificar todos los detalles tomando como ejemplo el artículo 25 de la LEY FEDERAL DEL TRABAJO. Se tiene que tener cuidado de especificar que es lo que va a desarrollar el Outsourcing, los tiempos de entrega, fechas de revisión, especificar quien proveerá la herramienta con la que se elaboraran las aplicaciones WEB (hardware y software), la plataforma a usar, los detalles de la licencias de dicho software. Así mismo en el contrato con el Outsourcing se deben detallar puntos como los derechos de autor sobre la aplicación a desarrollarse y sobre la confidencialidad de la información que se brinda al tercero para desarrollar la aplicación WEB, así mismo se debe especificar los permisos y el nivel de acceso que se tendrá a la información de la empresa ya que si no se tiene un contrato donde se especifiquen estos detalles los Outsourcing pueden no tener un buen control sobre si personal.

4.4 MEJORES PRACTICAS APLICADAS A TERCEROS

Existen varias mejores prácticas que se pueden implementar para las aplicaciones WEB desarrolladas por terceros, dichas prácticas pueden ser ITIL, CMMi, ISO27001, Six Sigma, entre otros. A continuación se presenta una breve descripción de cómo apoyarse en ITIL y COBIT.

4.4.1 ITIL

Las empresas de TI necesitan concentrarse en la calidad de los servicios que brindan, y asegurarse que los mismos estén alineados a los objetivos de la organización que las contrata.

Cuando los servicios de TI son críticos, cada una de las actividades que se realizan deben de estar ejecutadas con un orden determinado para asegurar que el grupo de TI proporciona valor y entrega los servicios de forma consistente.

ITIL tiene que ver con todos aquellos procesos que se requieren ejecutar dentro de las organizaciones para la administración y operación de la infraestructura de TI, de tal forma que se

tenga una óptima provisión de servicios a los clientes bajo un esquema de costos congruentes con las estrategias del negocio.

Desarrollada su primera versión a finales de 1980, la Biblioteca de Infraestructura de Tecnologías de la Información (ITIL) se ha convertido en el estándar mundial de facto en la Gestión de Servicios Informáticos. Uno de los conceptos esenciales de ITIL es que establece que para una adecuada Gestión de Servicios en las Tecnologías de Información es necesaria una mezcla entre tres factores: Personas, Procesos y Tecnología.

4.4.1.1 CARACTERÍSTICAS DE ITIL

- Es un Framework de procesos de IT no propietario.
- Es independiente de los proveedores.
- Es independiente de la tecnología.
- Está basado en "Best Practices".

Provee:

- Una terminología estándar.
- Las interdependencias entre los procesos.
- Los lineamientos para la implementación.
- Los lineamientos para la definición de roles y responsabilidades de los procesos
- Las bases para comparar la situación de la empresa frente a las “mejores prácticas”.
-

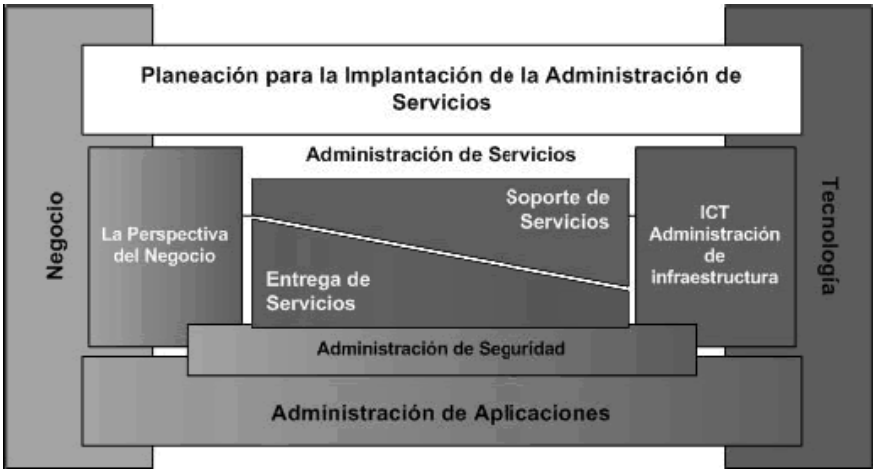

Figura 4.1 Planeación para la Implementación de ITIL[90]

[90] Centro de Coordinación de ITIL UTN FRBA, ITIL - Mejores Prácticas en la Gestión de Servicios de TI.

4.4.1.2 ADMINISTRACIÓN O GESTIÓN DE SERVICIOS DE TI

La gestión de Servicios Informáticos es abarcada por dos publicaciones: Entrega de Servicios y Soporte de Servicios. **Entrega de Servicios**: Cubre los procesos necesarios para la planeación y entrega de la calidad de los servicios de TI. Estos procesos son:

- Administración de Niveles de Servicio
- Administración Financiera
- Administración de Capacidad
- Administración de la Continuidad de Servicios de TI
- Administración de la Disponibilidad

Soporte de Servicios: Proporciona los detalles de la función de mesa de servicio y los procesos necesarios para el soporte y mantenimiento de los servicios de TI. Estos procesos son:

- Administración de Incidentes
- Administración de Problemas
- Administración de Configuraciones
- Administración de Cambios
- Administración de Releases

4.4.1.3 PROCESOS DE GESTIÓN DE SERVICIOS

La Gestión de Servicios de TI organiza las actividades necesarias para administrar la entrega y soporte de servicios en procesos.

Un proceso es una serie de actividades que a partir de una entrada obtienen una salida. El flujo de la información dentro y fuera de cada área de proceso indicará la calidad del proceso en particular.

Existen puntos de monitoreo en el proceso para medir la calidad de los productos y provisión de los servicios. Los procesos pueden ser medidos por su efectividad y eficiencia, es decir, si el proceso alcanzó su objetivo y si se hizo un óptimo uso de los recursos para lograr ese objetivo.

Por lo que si el resultado de un proceso cumple con el estándar definido, entonces el proceso es efectivo, y si las actividades en el proceso están cumpliendo con el mínimos requerido esfuerzo y costo, entonces el proceso es eficiente.

4.4.2 COBIT

El estándar COBIT (Control Objectives for Information and related Technology) ofrece un conjunto de mejores prácticas para la gestión de los Sistemas de Información de las organizaciones.

El objetivo principal de COBIT consiste en proporcionar una guía a alto nivel sobre puntos en los que establecer controles internos con tal de:

- Asegurar el buen gobierno, protegiendo los intereses de los clientes, accionistas, empleados.
- Garantizar el cumplimiento normativo del sector al que pertenezca la organización
- Mejorar la eficacia y eficiencia de los procesos y actividades de la organización
- Garantizar la confidencialidad, integridad y disponibilidad de la información

El estándar define el término control como: “Políticas, procedimientos, prácticas y estructuras organizacionales diseñadas para proveer aseguramiento razonable de que se lograrán los objetivos del negocio y se prevendrán, detectarán y corregirán los eventos no deseables”
Por tanto, la definición abarca desde aspectos organizativos (flujo para pedir autorización a determinada información, procedimiento para reportar incidencias, selección de proveedores) hasta aspectos más tecnológicos y automáticos (control de acceso a los sistemas, monitorización de los sistemas mediante herramientas automatizadas).

Por otra parte, todo control tiene por naturaleza un objetivo. Es decir, un objetivo de control es un propósito o resultado deseable como por ejemplo: garantizar la continuidad de las operaciones ante situaciones de contingencias.

En consecuencia, para cada objetivo de control de nuestra organización puede implementar uno o varios controles (ejecución de copias de seguridad periódicas, traslado de copias de seguridad a otras instalaciones) que nos garanticen la obtención del resultado deseable (continuidad de las operaciones en caso de contingencias).

COBIT clasifica los procesos de negocio relacionados con las Tecnologías de la Información en 4 dominios:

- Planificación y Organización
- Adquisición e Implementación
- Entrega y Soporte
- Supervisión y Evaluación

4.4.2.1 PLANIFICACIÓN Y ORGANIZACIÓN

La dirección de la organización debe implicarse en la definición de la estrategia a seguir en el ámbito de los sistemas de información, de forma que sea posible proporcionar los servicios que requieran las diferentes áreas de negocio. Para ello, COBIT presenta 10 procesos:

- PO1: Definición de un plan estratégico: gestión del valor, alineación con las necesidades del negocio, planes estratégicos y tácticos.
- P02: Definición de la arquitectura de información: modelo de arquitectura, diccionario de datos, clasificación de la información, gestión de la integridad.
- P03: Determinar las directrices tecnológicas: análisis de tecnologías emergentes, monitorizar tendencias y regulaciones.
- P04: Definición de procesos IT, organización y relaciones: análisis de los procesos, comités, estructura organizativa, responsabilidades, propietarios de la información, supervisión, segregación de funciones, políticas de contratación.
- P05: Gestión de la inversión en tecnología: gestión financiera, priorización de proyectos, presupuestos, gestión de los costos y beneficios.
- P06: Gestión de la comunicación: políticas y procedimientos, concienciación de usuarios.
- P07: Gestión de los recursos humanos de las tecnologías de la información: contratación, competencias del personal, roles, planes de formación, evaluación del desempeño de los empleados.
- P08: Gestión de la calidad: mejora continua, orientación al cliente, sistemas de medición y monitorización de la calidad, estándares de desarrollo y adquisición.
- P09: Validación y gestión del riesgo de las tecnologías de la información
- P10: Gestión de proyectos: planificación, definición de alcance, asignación de recursos.

4.4.2.2 ADQUISICIÓN E IMPLEMENTACIÓN

Con el objeto de garantizar que las adquisiciones de aplicaciones comerciales, el desarrollo de herramientas a medida y su posterior mantenimiento se encuentre alineado con las necesidades del negocio, el estándar Cobit define los siguientes 7 procesos:

AI1: Identificación de soluciones: análisis funcional y técnico, análisis del riesgo, estudio de la viabilidad.
AI2: Adquisición y mantenimiento de aplicaciones: Diseño, controles sobre la seguridad, desarrollo, configuración, verificación de la calidad, mantenimiento.
Objetivo de control: Adquisición y mantenimiento de aplicaciones software.
Requisito de negocio: Suministrar funciones automáticas que soporten de forma efectiva los procesos de negocio.

Se debe comprobar si existe:

- La definición de estados específicos de los requisitos funcionales y operativos.
- Una implementación estructurada con dictámenes claros.
- Las actividades de desarrollo y pruebas están separadas.

Indicadores:

- Cantidad de aplicaciones entregadas puntualmente de acuerdo al plan.
- Cantidad de pedidos de cambios relacionados con defecto del sistema.
- Tiempo en que tardan en analizar y resolver problemas.

AI3: Adquisición y mantenimiento de la infraestructura tecnológica: Plan de infraestructuras, controles de protección y disponibilidad, mantenimiento.
AI4: Facilidad de uso: Formación a gerencia, usuarios, operadores y personal de soporte.
AI5: Obtención de recursos tecnológicos: control y asignación los recursos disponibles, gestión de contratos con proveedores, procedimientos de selección de proveedores.
AI6: Gestión de cambios: Procedimientos de solicitud/autorización de cambios, verificación del impacto y priorización, cambios de emergencia, seguimiento de los cambios, actualización de documentos.

AI7: Instalación y acreditación de soluciones y cambios: Formación, pruebas técnicas y de usuario, conversiones de datos, test de aceptación por el cliente, traspaso a producción.

4.4.2.3 ENTREGA Y SOPORTE

La entrega y soporte de servicios se encuentran constituidos por diversos procesos orientados a asegurar la eficacia y eficiencia de los sistemas de información.

DS1: Definición y gestión de los niveles de servicio: SLA con usuarios/clientes
DS2: Gestión de servicios de terceros: gestión de las relaciones con proveedores, valoración del riesgo (non-disclousure agreements NDA), monitorización del servicio.
Objetivo de control: Gestionar los servicios prestados por terceros.
Requisito de negocio: Asegurar que las reglas y las responsabilidades de terceras partes están definidas de forma clara, adheridas y continuar satisfaciendo los requisitos.

Se tomará en consideración:

- Monitorización de contratos existentes.
- Acuerdos de servicio con terceras partes.
- Requisitos legales y regulados.

Indicadores:

- Cantidad de contratos de servicio actualizados.

DS3: Gestión del rendimiento y la capacidad: planes de capacidad, monitorización del rendimiento, disponibilidad de recursos.
DS4: Asegurar la continuidad del servicio: plan de continuidad, recursos críticos, recuperación de servicios, copias de seguridad.
DS5: Garantizar la seguridad de los sistemas: gestión de identidades, gestión de usuarios, monitorización y tests de seguridad, protecciones de seguridad, prevención y corrección de software malicioso, seguridad de la red, intercambio de datos sensibles.
DS6: Identificar y asignar costos
DS7: Formación a usuarios: identificar necesidades, planes de formación.
DS8: Gestión de incidentes y Help Desk: registro y escalado de incidencias, análisis de tendencias.

DS9: Gestión de configuraciones: definición de configuraciones base, análisis de integridad de configuraciones.
DS10: Gestión de problemas: identificación y clasificación, seguimiento, integración con la gestión de incidentes y configuraciones.
DS11: Gestión de los datos: acuerdos para la retención y almacenaje de los datos, copias de seguridad, pruebas de recuperación.
DS12: Gestión del entorno físico: acceso físico, medidas de seguridad, medidas de protección medioambientales.
DS13: Gestión de las operaciones: planificación de tareas, mantenimiento preventivo.

4.4.2.4 SUPERVISIÓN Y EVALUACIÓN

- Garantizar la alineación con la estratégica del negocio
- Verificar las desviaciones en base a los acuerdos del nivel de servicio
- Validar el cumplimiento regulatorio
- Esta supervisión implica paralelamente la verificación de los controles por parte de auditores (internos o externos), ofreciendo una visión objetiva de la situación y con independencia del responsable del proceso.

ME1: Monitorización y evaluación del rendimiento
ME2: Monitorización y evaluación del control interno
ME3: Asegurar el cumplimiento con requerimientos externos
ME4: Buen gobierno

4.5 ISO 27001

La información tiene una importancia fundamental para el funcionamiento y quizá incluso sea decisiva para la supervivencia de la organización. El hecho de disponer de la certificación según ISO 27001 ayuda a gestionar y proteger valiosos activos de información.

La norma ISO 27001 es un estándar internacional para la seguridad de la información publicado en Octubre de 2005. Dedicado a la organización de la seguridad de las tecnologías de la información.
Establece un sistema gerencial que permite minimizar el riesgo y proteger la información de amenazas externas o internas.

Esta norma, no está orientada a despliegues tecnológicos o de infraestructura, sino a aspectos netamente organizativos, es decir, la frase que podría definir su propósito es Organizar la seguridad de la información.

Actualmente es el único estándar aceptado internacionalmente para la administración de la Seguridad de la Información y se aplica a todo tipo de organizaciones, independientemente de su tamaño o actividad.

Su objetivo principal es el establecimiento e implementación de un Sistema de Gestión de la Seguridad de la Información.

La seguridad de la información debe preservar la:

- Confidencialidad: Aseguramiento de que la información es accesible sólo para aquellos autorizados a tener acceso.
- Integridad: Garantía de la exactitud y completitud de la información y de los métodos de su procesamiento.
- Disponibilidad: Aseguramiento de que los usuarios autorizados tienen acceso cuando lo requieran a la información y sus activos asociados.

Entre otros objetivos de la norma son los siguientes:

- La definición clara y transmitida a toda la organización de los objetivos y directrices de seguridad.
- La sistematización, objetividad y consistencia a lo largo del tiempo en las actuaciones de seguridad.
- El análisis y prevención de los riesgos en los Sistemas de Información.
- La mejora de los procesos y procedimientos de gestión de la información.
- La motivación del personal en cuanto a valoración de la información.
- El cumplimiento con la legislación vigente.
- Una imagen de calidad frente a clientes y proveedores.

Propone secuencias de acciones tendientes al:

1 Establecimiento-Implementación
2 Operación
3 Monitorización
4 Revisión-Mantenimiento
5 Mejora SGSI Sistema de Gestión de la Seguridad de la Información.

Estas son las normas Publicadas referidas a ISO 27001

2005 - ISO-27001 Certificable.

2006 - ISO 27006 regula los organismos de certificación, alineada con 17021.

2007 - ISO-27002 guía de controles, Ex 17799.

Sin publicar aún.

27003 Ayuda para la implantación SGSI

27004 Métricas

27005 Riesgos

27007 requisitos de auditoría de un SGSI

27011 Sector TICs

27031 Plan de Continuidad de Negocio.

27032 Cyber seguridad

27033 Seguridad en redes, sobre la base de 18028

27034 Seguridad en las aplicaciones.

4.5.1 BENEFICIOS AL IMPLANTAR LA NORMA

El principal beneficio de la norma ISO 27001 es que aumenta la credibilidad de cualquier organización. La norma claramente demuestra la validez de su información y un compromiso real de mantener la seguridad de la información. El establecimiento y certificación de un SGSI puede así mismo transformar la cultura corporativa tanto interna como externa de la empresa al implantar la Norma, abriendo nuevas oportunidades de negocio con clientes conscientes de la importancia de la seguridad, además de mejorar el nivel ético y profesional de los empleados y la noción de la confidencialidad en el puesto de trabajo. Aún más, permite reforzar la seguridad de la información y

reducir el posible riesgo de fraude, pérdida de información y revelación. Además de un alto nivel de transparencia y confianza para los empleados de la empresa, de sus clientes y proveedores.

4.5.2 NORMA ISO 27001

ISO 27001 es la única norma internacional auditable que define los requisitos para un sistema de gestión de la seguridad de la información (SGSI). Donde un SGCI es una parte del sistema de gestión de una organización, basado en una aproximación de los riesgos del negocio (actividad) para establecer, implementar, operar, monitorizar, revisar, mantener y mejorar la seguridad de la información. La creación de un SGSI es una decisión estratégica en una organización y como tal, debe ser apoyada y supervisada por la dirección. El hecho de certificar un SGSI según la norma ISO 27001 puede aportar las siguientes ventajas a la empresa que siga un Modelo de Implementación:

- Demuestra la garantía independiente de los controles internos y cumple los requisitos de gestión corporativa y de continuidad de la actividad comercial.
- Demuestra independientemente que se respetan las leyes y normativas que sean de aplicación.
- Proporciona una ventaja competitiva al cumplir los requisitos contractuales y demostrar a los clientes que la seguridad de su información es primordial.
- Verifica independientemente que los riesgos de la organización estén correctamente identificados, evaluados y gestionados al tiempo que formaliza los procesos. procedimientos y documentación de protección de la información.
- Demuestra el compromiso de la cúpula directiva de su organización con la seguridad de la información.
- El proceso de evaluaciones periódicas ayudan a supervisar continuamente el rendimiento y la mejora.

En la norma ISO 27001, se menciona lo siguiente:

"La organización, establecerá, implementará, operará, monitorizará, revisará, mantendrá y mejorará un documentado SGSI en su contexto para las actividades globales de su negocio y de cara a los riesgos".

El mismo documento debe contener:

- La política de seguridad
- Las normas o estándares de funcionamiento
- Los procedimientos detallados
- Guías y recomendaciones

Para implementar el SGSI se debe utilizar el ciclo continuo PDCA10 cuyo objetivo final es asegurar la Integridad, Confidencialidad y Disponibilidad de la Información.

La metodología PDCA es la médula de la instrumentación básica de la Gestión de la Calidad Total, pero vale la pena insistir en que su extraordinaria potencialidad técnica solo podrá ser bien aprovechada si hay adecuada motivación, participación y valorización de los técnicos y funcionarios. La metodología PDCA está integrada por cuatro pasos.

- **PLANEAR- PLAN (ESTABLECER EL SGSI):** Establecer política, objetivos, procesos y procedimientos SGSI relevantes para manejar el riesgo y mejorar la seguridad de la información para entregar resultados en concordancia con las políticas y objetivos generales de la organización.
- **HACER-DO (IMPLEMENTAR Y OPERAR EL SGSI):** Implementar y operar la política, controles, procesos y procedimientos SGSI.
- **CHEQUEAR- CHECK (MONITOREAR Y REVISAR EL SGSI):** Evaluar y, donde sea aplicable, medir el desempeño del proceso en comparación con la política, objetivos y experiencias prácticas SGSI y reportar los resultados a la gerencia para su revisión.
- **ACTUAR-ACT (MANTENER Y MEJORAR EL SGSI):** Tomar acciones correctivas y preventivas, basadas en los resultados de la auditoría interna SGSI y la revisión gerencial u otra información relevante, para lograr el mejoramiento continuo del SGSI.

El uso eficiente de estos recursos, aplicando rigurosamente los procedimientos correspondientes, permitirá obtener resultados cada vez mejores, alcanzándose las metas establecidas, o por lo menos, acercándose cada vez más a ellas. Se trata de una metodología relativamente simple, pero que, si bien aplicada gerencialmente y con funcionarios motivados, será extraordinariamente efectiva.

CAPÍTULO V MODELO DE SEGURIDAD EN LAS APLICACIONES WEB DESARROLLADAS POR UN TERCERO

En este capítulo se da a conocer el modelo de seguridad el cual describe el uso de técnicas y herramientas para minimizar los ataques que se vieron en los capítulos anteriores, se adapta a cualquier tecnología que nos permita desarrollar una aplicación Web. Este modelo proporciona información general acerca de las características y servicios de seguridad principalmente en 4 niveles Capa de presentación, Capa de Negocio, Capa de Servicios y Capa de arquitectura.

CAPÍTULO V MODELO DE SEGURIDAD EN LAS APLICACIONES WEB DESARROLLADAS POR UN TERCERO

5.1 MODELO DE SEGURIDAD EN APLICACIONES WEB.

El modelo que se representa con la figura 5.1 muestra las cuatro capas propuestas por el modelo de seguridad, durante este capítulo se explica cada una de las capas y los subcomponentes dentro de cada una de ellas.

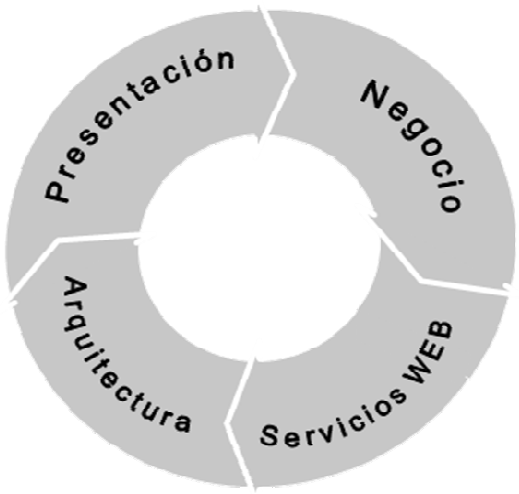

Figura 5.1Modelo de Seguridad

5.1.1 CAPA DE PRESENTACIÓN

Es la capa que ve el usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio. También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario.

5.1.1.1 TÉCNICAS Y HERRAMIENTAS DE SEGURIDAD

5.1.1.1.1 USO ADECUADO DE FRAMEWORKS DE JavaScript(AJAX)

- EL JavaScript Ajax solo debe ser usado para renderizar componentes de la capa de presentación.
- Nunca debe contener lógica de negocio en el Java script.
- Validaciones necesarias para disminuir las peticiones a nuestro servidor capa de negocio.

- No usar Proxy para llamar a otros sistemas como por ejemplo Web Services.
- Usar algún Framework estable de JS Query, Prototipo o bien Mootools.

5.1.1.1.2 VALIDACIÓN ANTISAMY

Antisamy es un conjunto de librerías para validar la capa de presentación, nos ayuda a reforzar los puntos débiles en un ataque de inyección de JavaScripty CSS basándose en el uso de expresiones regulares. Este proyecto se encuentra disponible para diferentes lenguajes de programación.

5.1.1.1.3 COMPONENTE DE ENCRIPTACIÓN CRYTTR

Componente encargado de encriptar y desincriptar datos en la aplicación Web, utiliza un algoritmo de encriptación y desincriptación donde se debe de conocer la llave publica tanto en la capa de negocio como en la capa de presentación.

5.1.1.1.4 ANTI TAMPERING

Técnica que consiste en generar un hash del lado del servidor para identificar al usuario que se logueo en la aplicación. Cada acción que el usuario realiza se valida del lado del servidor comparando el hash generado con anterioridad si es incorrecto se invalida la petición.
Algoritmos comunes MD5, SHA, SHA1.

5.1.2 CAPA DE NEGOCIO

Es donde se ejecuta las operaciones referentes a lo que debe de realizar el sistema según los requerimientos establecidos, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos, para almacenar o recuperar datos de él.

5.1.3 TÉCNICAS Y HERRAMIENTAS DE SEGURIDAD

5.1.3.1.1 LIMITACIÓN DE INTENTOS DE LOGUEO

Técnica que consiste en disminuir un ataque de fuerza bruta al querer entrar al sistema, limitar el número de intentos deshabilitar al usuario o forzar un tiempo de espera disminuirá las probabilidad de que este ataque tenga éxito, es necesario tener una bitácora para analizar los intentos fallidos así como el nombre de usuario y la maquina que intenta entrar al sistema.

5.1.3.1.2 COMPONENTES DE VALIDACIÓN APLICADOS AL NEGOCIO.

Utilizar un conjunto de librerías para validar los datos que llegan de la capa de negocio, siempre será necesario realizar las validaciones de las entradas de formulario. Verificar números, cadenas expresiones regulares. Estas validaciones también deben de existir en la capa de presentación con el objetivo de disminuir las peticiones.

5.1.3.1.3 ELIMINACIÓN DE CICLOS INNECESARIOS.

Verificar que el código no tenga errores lógicos de programación el tener un ciclo infinito dentro de la aplicación gastaría los recursos del CPU ineficientemente provocando un DOS, es necesario verificar la partes del negocio que sean más solicitadas estén desarrolladas o programadas eficientemente.

5.1.3.1.4 COMPONENTE DE LOG DE SUCESOS

Utilizar alguna librería que nos permita identificar los pasos realizados e información de lo que está sucediendo dentro del sistema únicamente información necesaria, el enviar mensajes al estándar output (salida a consola) gasta demasiados recursos del procesador pudiendo provocar un DOS.
Ejemplo de librería Log4j

5.1.3.1.5 COMPONENTES DE TRANSACCIONABILIDAD

Uso de librerías para manejar la transaccionabilidad a la base de datos. Permitir integridad de los datos. Verificar que las transacciones se ejecuten en puntos específicos y no corromper nuestra base de datos.

5.1.3.1.6 COMPONENTES SQL DE TIPO "BIND VARIABLE"

Conjunto de librerías correspondiente al lenguaje de programación en el cual se utiliza binding para construir la sentencia SQL, estos componentes nos ayudan a evitar la inyección de sql.

5.1.4 CAPA DE DATOS

Es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

5.1.5 CAPA DE SERVICIOS WEB.

Capa encargada de exponer el negocio a consumidores de servicios “aplicaciones cliente” básicamente sirve para intercambiar información entre dos aplicaciones no importa el lenguaje en el que se encuentran programadas, estos servicios se encuentran desarrollados en un estándar llamado WSDL y utilizan SOAP como protocolo .

5.1.5.1 TÉCNICAS Y HERRAMIENTAS DE SEGURIDAD

5.1.5.1.1 CREDENCIALES DE AUTENTICACIÓN

Proveer a las aplicaciones consumidoras de servicio (aplicaciones cliente) un usuario y contraseña esto es para limitar a las aplicaciones cliente y dar un servicio de respuesta más rápido. Con esto puede disminuir que un usuario ajeno estrese la aplicación
Relacionar un usuario a una IP especifica podríamos disminuir la suplantación de identidad como consecuencia evitar el servicio a cualquier usuario mal intencionado que podría tener las herramientas necesarios para generar un ataque DDOS.

5.1.5.1.2 LIMITACIÓN DE CONSUMO

Limitar al cliente a consumir a un numero de real de peticiones por hora es decir si se tiene un numero exagerado de peticiones podríamos catalogarlo con un ataque de tipo DOS provocando que tire todos nuestros servicios, este procedimiento mejora el rendimiento y estabilidad de nuestra aplicación.

5.1.6 CAPA DE ARQUITECTURA

Esta capa es la encargada de cumplir con las características necesarias para un buen funcionamiento de la aplicación, es el entorno donde va a vivir el sistema.

5.1.6.1 TÉCNICAS Y HERRAMIENTAS DE SEGURIDAD

5.1.6.1.1 ESTRUCTURA DE BALANCEO DE CARGA

En una aplicación con un número alto de peticiones es ideal esta arquitectura. Consiste realizar un cluster ya sea de tipo servidor de aplicaciones o base de datos. Como su nombre lo indica se reparten las peticiones entre servidores, si un servidor se cae los demás siguen trabajando. Esta estructura no permite disminuir un ataque de tipo DOS, DDOS.

5.1.6.1.2 COMPONENTES ORM

Utilización de componentes ORM permite disminuir el estrés a la base de datos debido a que se almacenan los registros de las últimas consultas. Las nuevas peticiones se resuelven del cache generado si es que se encuentra de lo contrario se tendrán que pedir de nuevo a la base de datos y almacenarlo en dicho cache, con esto se obtiene una mayor repuesta por parte del servidor.

5.1.6.1.3 SERVIDOR DE CACHE APACHE

Contar con un servidor de cache este servidor almacenará las ultimas peticiones o URL estáticas como imágenes scripts archivos flash y otros, este servidor nos ayuda a disminuir la carga a la aplicación de estar generando una nueva respuesta por cada petición.

5.1.6.1.4 COMPONENTES FIREWALL DE APLICACIÓN MODSECURITY

Firewall que se ocupa dentro de un servidor Web el principal objetivo es proveer protección contra diversos ataques y permite monitorear tráfico HTTP, así como realizar análisis en tiempo real sin necesidad de hacer cambios a la infraestructura existente.

5.1.6.1.5 IMPLEMENTACIÓN DE SSL(HTTPS)

Permite tener confidencialidad en las transmisiones de nuestra capa de negocio a la capa de presentación y viceversa.

Proporciona sus servicios de seguridad cifrando los datos intercambiados entre el servidor y el cliente con un algoritmo de cifrado simétrico, típicamente el RC4 o IDEA, y cifrando la clave de sesión de RC4 o IDEA mediante un algoritmo de cifrado de clave pública, típicamente el RSA.

Un componente muy común para la generación de los certificados digitales es la utilización de OpenSSL.

CAPÍTULO VI APLICACIÓN DEL MODELO DE SEGURIDAD

En este capítulo se aplica a un caso práctico el modelo de seguridad; las técnicas y herramientas para minimizar los ataques explicados durante el desarrollo de este trabajo. Este modelo fue desarrollado para que se adapte a cualquier tecnología que nos permita desarrollar una aplicación Web. Este modelo proporciona información general acerca de las características y servicios de seguridad principalmente en 4 niveles Capa de presentación, Capa de Negocio, Capa de Servicios y Capa de arquitectura, finalmente se explican las ventajas y desventajas del uso del modelo así como la conclusión a la que se llego.

CAPÍTULO VI APLICACIÓN DEL MODELO DE SEGURIDAD

GENERALIDADES DE LA EMPRESA

Desde 1999, un significativo crecimiento ha convertido a Dominion en uno de los grupos tecnológicos de referencia en el mercado de la nueva economía.

Gracias al desarrollo sus actividades en cuatro grandes divisiones de negocio: logística, ingeniería, tecnologías de la información y biotecnología. Dominion ha crecido constantemente en los últimos años. Dominion cuenta con una capacidad extraordinaria para abordar proyectos en cualquier ámbito tecnológico, cubriendo, a través de sus diferentes divisiones, desde grandes proyectos de infraestructuras en el ámbito de la Ingeniería de Telecomunicaciones, hasta proyectos de desarrollo e implantación de soluciones en el ámbito de las Tecnologías de la Información, ofreciendo una visión 360º de las necesidades tecnológicas de sus clientes.

Dentro del área de tecnologías de la información Dominion cuenta con soluciones para ERP's, CRM's, BI, asistencia técnica, consultoría tecnológica y desarrollo de soluciones.

Dominion cuenta con una gran experiencia en el desarrollo de Soluciones a Medida, además del desarrollo de proyectos tanto en Inet como en plataformas OpenSource, y soluciones de movilidad. La empresa ha podido desarrollar conocimientos y soluciones específicas para determinadas necesidades verticales.

6.1 APLICACIÓN DEL MODELO

A continuación se muestra una comparativa del modelo de seguridad a un sistema en desarrollo con tecnología J2EE java.

Como características generales el sistema en desarrollo es el control y administración de los puntos de venta como los afiliados de los productos de la empresa de T-Regalo. http://www.t-regalo.com.mx

Este sistema se está desarrollando con la siguiente tecnología.

- Sistema gestor de datos Oracle.
- Plataforma Java J2EE Struts 1.2.

- Framework Extjs.
- JPA Toplink essentials.
- Framework Spring.

CAPA DE PRESENTACIÓN

Procedimiento	Aplicación dentro del desarrollo	Descripción	Resultado
Uso adecuando de JavaScript (sin lógica de negocio y servicios Web).	Correcto.	El uso de JavaScript únicamente para la construcción de todos los componentes visuales HTML, se realizan validación necesarias de campos obligatorios nunca se llaman a Web Services dentro de esta capa	Disminuye un ataque tampering, ataque SQL inyección si se maneja lógica de negocio
Herramientas para evitar la inyección de CSS y Java script.	Correcto	Utilización de un Framework Extjs el cual por default no permite la visualización del java script o CSS ingresado por el usuario	Disminuye la inyección de JavaScript y CSS
Uso de herramientas de encriptación de datos en el navegador	Incorrecto	No se usa ninguna herramienta para encriptar los datos que se envían desde el cliente hacia el servidor y viceversa	Cualquier persona mal intencionada puede obtener la información de la transacción por medio de un sniffer. Datos de envío como

			datos de respuesta.
Técnica Anti-Tampering	Incorrecto	No se usa ninguna técnica disminuir la manipulación de los campos ocultos	Cualquier persona mal intencionada puede modificar los parámetros ocultos y alterar la información para su conveniencia.

CAPA DE NEGOCIO

Procedimiento	Aplicación dentro del desarrollo	Descripción	Resultado
Limitación de número de intentos de logueo.	Incorrecto.	No se limitan los intentos de logueo.	Cualquier usuario mal intencionado puede recurrir a un ataque de fuerza bruta y entrar al sistema.
Componentes de validación de formulario.	Incorrecto.	Se toman en cuenta que la validación en la capa de presentación funciona correctamente.	Excepciones no esperadas dentro del sistema.
Código eficiente, disminución de ciclos innecesarios	Correcto	Se revisa la eficiencia de la aplicación estresándola con JMETER	Disminución de un ataque DDoS

Componente de Log de sucesos	Correcto	Utilización de librería log4j cuando se libera a nivel productivo únicamente se registran sucesos de vital importancia como son las excepciones de la aplicación.	Disminución de un ataque DDoS.
Componentes de transaccionabilidad.	Correcto	En las aplicaciones se utiliza Spring como contenedor Web el cual se encarga de realizar las transacciones correspondientes.	Proteger la autenticación de los datos.
Componentes SQL de tipo "bindy variable"	Correcto	En las aplicaciones desarrolladas por Dominion se utiliza toplink essential con JPATempleate de Spring los dos utilizan binding.	Disminuye un ataque de inyección de SQL

CAPA DE SERVICIOS WEB

Procedimiento	Aplicación dentro del desarrollo	Descripción	Resultado
Autenticación de usuarios.	Correcto.	Al realizar Web servicies siempre se valida a la aplicación	Disminución de un ataque DDoS

		cliente que realiza la transacción.	
Identificación por IP de usuario	Incorrecto	Debido al negocio no se puede amarrar una IP específica a un cliente.	Suplantación de identidad.
Limitación de consumo	Incorrecto	Nunca se toma en cuenta cuantas peticiones puede realizar la aplicación cliente.	Ataque tipo Dos o DDoS

CAPA DE ARQUITECTURA

Procedimiento	Aplicación dentro del desarrollo	Descripción	Resultado
Estructura del balanceo de carga	Incorrecto.	Hasta ahora no se ha llevado una arquitectura de ese tipo debido a las dimensiones de la aplicación.	Ataque de tipo DOS y DDoS
Componentes ORM	Correcto	Implementación de ORM JAP Toplink essential	Ataque de tipo DOS y DDoS
Servidor de cache	Incorrecto	No se ha implementado un	Ataque de tipo DOS

		servidor de cache	
Componentes Firewall de aplicación	Incorrecto	No se ha implantado un firewall de ampliación	Ataque DOS
Implementación de SSL Http	Correcto	En el desarrollo siempre se revisa que la implementación funcione correctamente con Http.	

6.2 RESULTADOS

Los resultados arrojados después de aplicar el modelo de seguridad a la aplicación Web nos muestra que el sistema no cumple 100% con las propuestas sugerida en el modelo por lo que se pueden analizar como implementar las mejores prácticas sugeridas.

En la capa de presentación no se cuenta con la encriptación para el navegador o Tampering por lo que la aplicación puede ser susceptible de ser atacada por medio de sniffer o manipulación de datos. El diseñador de la aplicación debe tomar en cuenta estas vulnerabilidades y mejorar el diseño de la aplicación.

La capa de negocio es una de las más importantes, pero esta aplicación no limita el número de logeo o cuenta con una validación de datos. Estas vulnerabilidades hacen débil la aplicación de ataques internos o externos pero si cuenta con transaccionabilidad lo que hace que la aplicación cuenta con consistencia de los datos así como un código eficiente que evita grandes consultas innecesarias que consumen recursos y hacen menos eficiente la aplicación.

Los servicios Web no están aplicados completamente ya que no cumple con la asociación de usuarios con una IP pero esto es una justificación puesto que al ser una aplicación Web está puede ser acezada desde diferentes ubicaciones, tampoco limitan el consumo de cada usuario en la aplicación, esto si puede ser eficientado ya que un usuario puede consumir grandes cantidades de recursos de la aplicación con procesos innecesarios. El punto que cumple la aplicación es la

autenticación de usuarios para accesar la aplicaciones, esto brinda cierta seguridad ya que el sistema no es de acceso libre a cualquier usuario.

Finalmente, la capa de arquitectura tampoco cumple al 100% con el modelo de seguridad puesto que no tiene un servidor cache para eficientar los accesos a base de datos o un firewall de aplicación. El diseño puede ser eficientado o analizar los puntos que no cumplen tienen una justificación dentro de la lógica del negocio o de la aplicación.

6.3 VENTAJAS

Con base a las capas establecidas en el modelo de seguridad se detectaron diversas ventajas; a continuación se mencionan cada una de estas:

CAPA DE PRESENTACIÓN

- Cuando no se tiene contemplada una lógica de negocios, el uso adecuado de Java Script ayuda a disminuir los ataques de tipo tampering, ya que sólo se realiza la validación de los campos obligatorios y no se realiza un llamado a los Web Services.
- La utilización de la herramienta Framework Extjs no permite que el usuario realice la inyección de código malicioso.
- El uso de la encriptación de datos en el navegador permite disminuir los ataques de personas mal intencionadas que deseen obtener información durante la transacción por medio de sniffer.
- Al utilizar las técnicas Anti-Tampering se evita que cualquier persona mal intencionada puede modificar los parámetros ocultos y alterar la información a su conveniencia.

CAPA DE NEGOCIO

- Cuando se establece un límite de intentos de logueo se previenen los ataques de fuerza bruta para entrar a la aplicación.
- Al establecer componentes para validación de formularios se disminuyen las excepciones no esperadas dentro del sistema.
- El realizar una lógica de programación adecuada permite no tener ciclos innecesarios dentro del sistema, evitando los ataques Dos.
- El registro de sucesos de vital importancia como las excepciones de la aplicación se atribuye al uso de la librería Log4j de Java Script, lo cual contribuye a la baja de ataques de tipo Dos.

- Al aplicar los componentes de transaccionabilidad como lo es la herramienta Spring se encargan de proteger la autenticación de los datos.
- La utilización de componentes SQL de tipo "bind variable" contribuye a la disminución de los ataques de inyección de SQL.

CAPA DE SERVICIOS WEB

- Cuando en el sistema se aplica la autenticación de usuarios se disminuye los ataques del tipo DDoS.
- Al realizar la identificación por IP del usuario se disminuye o puede evitar la suplantación de identidad.
- La limitación de las peticiones del cliente hacia el servidor permite la disminución de los ataques Dos o DDoS.

CAPA DE ARQUITECTURA

- Al estructurar la carga de solicitudes se efectúa un balance de las peticiones, el cual se encarga de otorgar los recursos necesarios para la ejecución de estas, disminuyendo los ataques de tipo Dos y DDoS.
- El uso de los componentes ORM permite la integración completa de los elementos de la base de datos con los componentes de Java.
- La implementación de un Servidor de Cache evita los ataques del tipo Dos, ya que disminuye el número de consultas recurrentes a la base de datos, almacenando la información constantemente solicitada.
- La aplicación de los Componentes Firewall reduce la probabilidad de sufrir ataques tipo Dos.
- Al implementar los SSL HTTPS se aumenta la seguridad de la aplicación.

6.4 DESVENTAJAS

Entre las desventajas encontradas al implementar este modelo se seguridad se encuentran:

Costo Económico

- Se necesita invertir dinero en recursos externos capacitado en aplicar modelos de seguridad en aplicaciones Web.

- De otro modo, el equipo desarrollador interno debe recibir una capacitación para poder aplicar este Modelo.

Costo en Horas Hombre.

- Se debe establecer un tiempo determinado en el Desarrollo del proyecto para capacitar al personal en implementar el modelo de seguridad.
- Este costo en tiempo debe estar contemplado desde la planeación del proyecto, para evitar posibles limitaciones de tiempo y establecer un periodo predefinido para esta tarea.

Resistencia del personal al cambio de modelo.

- Debe existir una cultura para adaptarse a las nuevas tecnologías de seguridad entre los usuarios, de otro modo, estos se mostrarán renuentes al momento de aplicarlas en el desarrollo de las aplicaciones.

6.5 CONCLUSIONES

A lo largo de la aplicación del modelo en el desarrollo que está siendo realizado por Dominion se mencionaron procedimientos planteados en el modelo se seguridad Web, estos deben ser utilizados para reducir las vulnerabilidades de una aplicación. Se demostró que la aplicación que se examino no cumple con varios puntos del modelo, lo que la hace vulnerable a varios tipos de ataques, si se hubiese implementado el modelo desde el inicio de habrían ahorrado tiempo en re-trabajo; como no fue así el líderes del proyecto están consiente de los puntos que pueden ser aplicados y ellos deben evaluar cómo aplicarlos a esta aplicación, lo que tomará tiempo y por lo tanto costos para la empresa.

El modelo puede ser aplicado en cualquier plataforma de desarrollo por lo que no esta casada con alguna tecnología en especial, en el caso práctico de este trabajo se uso una aplicación Java, pero el modelo puede ser aplicado a otras tecnologías.

CONCLUSIONES

El presente trabajo ha contribuido a visualizar un ambiente general de la situación de la Auditoría en Informática, muchas veces se habla de controles y de seguridad pero pocas veces se toma conciencia de que estos son de poca utilidad si no son evaluados objetivamente. El mundo hoy en día avanza tan rápidamente que la tecnología se vuelve anticuada u obsoleta con gran facilidad y las empresas buscan estar a la vanguardia tecnológicamente para así ser más competitivas, no obstante, el tener la mejor tecnología no significa que esta sea utilizada correctamente o que sea absolutamente necesaria en nuestro entorno.

Un primer paso para una adecuada implementación de controles y de TI, consiste en Hacer conciencia en las empresas sobre la importancia de la Auditoría en Informática, por lo que se debe tener en cuenta que en la actualidad, una empresa no sobrevive si no es implementando la tecnología de información en sus áreas estratégicas, y si esta es poco eficaz, o bien insuficiente, las vuelve menos competentes.

Asimismo, y sin importar que tan grande sea el avance tecnológico en una empresa, sus flujos operativos no dejan de ser operados por un ser humano directa o indirectamente, por lo que una correcta evaluación nos ayuda a explotar mejor cada herramienta tecnológica que se utilice.

Hoy en día, la auditoría en informática tiene un papel fundamental dentro de las organizaciones, papel que muchas veces no se valora o se subestima, ya que los resultados que ésta da y sus beneficios, no se ven reflejados inmediatamente o bien, las sugerencias que nos arroja pueden llegar a ser consideradas como innecesarias o carentes de un valor real por la persona encargada de tomar las decisiones sobre estas.

Se debe de dar la importancia a implementar un modelo de seguridad en las aplicaciones Web dentro de las empresas, ya que existe dentro de ellas una dependencia hacia la tecnología cada vez mayor, si bien las amenazas, en cuanto a TI, siempre van un paso más adelante que las soluciones, entonces se debe procurar que estas soluciones sean implementadas adecuadamente y aprovechadas al máximo. Por ende, existe la necesidad de crear una cultura sobre ello y así mismo, volverlo un habito dentro de las empresas por medio de la auditoría en informática y la implementación de modelos de seguridad para todas las áreas, con ello se puede tener un mayor avance dentro de la organización así como un surgimiento gradual de mejores prácticas.

La existencia de una modelo de seguridad para evaluar las aplicaciones Web desarrolladas por terceros es muy importante, ya que por lo general, estos desempeñan un papel fundamental dentro

de las empresas privadas al contribuir en la mejora continua e implementación de controles sobre los procesos y operaciones necesarios para el cumplimiento de los objetivos de la empresa como organización, de manera tal que se adquiera un crecimiento que vaya de la mano con los expertos en la materia con el fin de detectar sus debilidades y factores de riesgo, disminuyéndolos.

Para llevar a cabo una exitosa implementación de un modelo de seguridad debe existir armonía entre los elementos involucrados como los organismos reguladores, la legislación informática y el fin que la organización en cuestión pretende alcanzar; manteniendo unificación y coherencia entre los objetivos que persigue cada uno de ellos.

Actualmente las empresas están invirtiendo fuertes cantidades en adquirir herramientas que controlen la seguridad, pero no están invirtiendo suficiente en funciones que evalúen que estas herramientas están correctamente implementadas y administradas.

En la mayoría de las empresas tanto públicas como privadas la función de auditoría en informática es impuesta y no implementada por iniciativa propia, porque consideran que solo representa un gasto extra, debido al desconocimiento que se tiene de los beneficios que se obtienen al contar con esta función.

La aplicación del modelo de seguridad propuesto en este trabajo ayuda a las empresas a eficientar la seguridad de las aplicaciones Web y como consecuencia a disminuir los costos relacionados con una aplicación son controles de seguridad.

BIBLIOGRAFÍA

HOWARD, Michael y LeBLANC, David. Writing Secure Code, 2nd Edition, Microsoft Press, EEUU 2003, Recuperado Mayo 2009

The Open Web Aplication Security Project. Una guía para construir aplicaciones y servicios Web seguros. Segunda edición. Editorial Black Hat, EEUU 2005, Recuperado Mayo 2009

MUÑOZ TORRES, Ivonne Valeria. Conservación de correos electrónicos en el Gobierno Federal. VLEX NETWORKS, S. L 2000, Recuperado Julio 2009

Código Penal Federal, Publicado en el Diario Oficial de la Federación en México el 14 de agosto de 1991, Recuperado Julio 2009

Unión Europea. Diario Oficial de las Comunidades Europeas, DIRECTIVA 2002/58/CE DEL PARLAMENTO EUROPEO Y DEL CONSEJO. 2002, Recuperado Agosto 2009

Unión Europea. Diario Oficial de las Comunidades Europeas, Decisión del Consejo relativa a la seguridad de los sistemas de información. 1992, Recuperado Agosto 2009

CONSULTAS DE INTERNET

PC Magazine [en línea]. Web Application Definition. Disponible en Web: http://www.pcmag.com/encyclopedia_term/0,2542,t=Web+application&i=54272,00.asp, Recuperado Marzo 2009

Wikipedia [en línea]. Definition of Web Application. Disponible en Web: http://en.wikipedia.org/wiki/Web_application, Recuperado Marzo 2009

Wikipedia [en línea]. History of Web Applications. Disponible en Web: http://en.wikipedia.org/wiki/Web_application/History, Recuperado Marzo 2009

Ferran Barba [en línea]. Aplicaciones Web a Medida. Disponible en Web: http://www.ferranbarba.com/aplicaciones-Web/, Recuperado Marzo 2009

Wikipedia [en línea]. Internet Applications. Disponible en Web: http://es.wikipedia.org/wiki/Aplicaciones_de_Internet_Ricas, Recuperado Marzo 2009

Wikipedia [en línea]. Framework para aplicaciones WEB. Disponible en Web: http://es.wikipedia.org/wiki/Framework_para_aplicaciones_Web, Recuperado Marzo 2009

Wikipedia [en línea]. OpenSSL. Disponible en Web: http://es.wikipedia.org/wiki/OpenSSL, Recuperado Marzo 2009

ROMERO, Antonio [en línea]. Administración y gerencia. Disponible en Web: http://www.gestiopolis.com/recursos/documentos/fulldocs/ger/outsourcingantonio.htm, Recuperado Abril 2009

Wikipedia [en línea]. Seguridad en Informática, 2009. Disponible en Web: http://es.wikipedia.org/wiki/Seguridad_inform%C3%A1tica, Recuperado Abril 2009

ALEGSA [en línea]. Definición de ataque informático. Disponible en Web: http://www.alegsa.com.ar/Dic/ataque%20informatico.php, Recuperado Abril 2009

KIOSKEA [en línea]. Introducción a los ataques. Disponible en Web: http://es.kioskea.net/contents/attaques/attaques.php, Recuperado Abril 2009

OWASP [en línea]. Direct Dynamic Code Evaluation. Disponible en Web: http://www.owasp.org/index.php/Direct_Dynamic_Code_Evaluation_('Eval_Injection') , Recuperado Abril 2009

Honey Net [en línea]. Definición de Remote Code-Inclusion. Disponible en Web: http://www.honeynet.org/node/6, Recuperado Abril 2009

Instituto de seguridad de Internet [en línea]. XSS. Disponible en Web: http://www.instisec.com/publico/xss.asp, Recuperado Abril 2009

FELTEN, Edward. Web Spoofing: an Internet Con Game [en línea]. Princeton University Department of Computer Science, 1996. Disponible en Web: http://www.cs.princeton.edu/sip/pub/spoofing.html, Recuperado Abril 2009

IMCP. Criptología y Seguridad en Internet [en línea], 2009. Disponible en Web: http://www.imcp.org.mx/spip.php?article108, Recuperado Abril 2009

Wikipedia [en línea]. Ataque de fuerza bruta. Disponible en Web: http://es.wikipedia.org/wiki/Ataque_de_fuerza_bruta, Recuperado Abril 2009

Gestiopolis [en línea]. Outsourcing. Disponible en Web: http://www.gestiopolis.com/recursos/documentos/fulldocs/ger/outsourcingantonio.htm, Recuperado Abril 2009

RadioTrece [en línea]. Robo de información y datos en las empresas. Disponible en Web: http://www.radiotrece.com.mx/2007/10/29/robo-de-informacion-y-datos-en-las-empresas/, Recuperado Abril 2009

ACUÑA, Andrés. Economía y negocios [en línea]. Disponible en Web: http://www.economiaynegocios.cl/noticias/noticias.asp?id=45823, Recuperado Abril 2009

Noticias [en línea]. Ciberdelito lo nuevo informática. Disponible en Web: http://www.noticias.com/articulo/18-08-2006/emil-domec/ciberdelito-lo-nuevo-informatica-como-robar-base-datos-570j.html, Recuperado Abril 2009

Seguridad información [en línea]. La otra cara del Robo de Información. Disponible en Web: http://seguridad-informacion.blogspot.com/2007/11/la-otra-cara-del-robo-de-informacin.html, Recuperado Abril 2009

Inteco [en línea]. Seguridad de información en equipos directivos. Disponible en Web: http://www.inteco.es/, Recuperado Abril 2009

TECHWEEK, Marie-Claire. Riesgo de fugas de información. [en línea]. Disponible en Web: http://www.techweek.es/autores/aut112, Recuperado Abril 2009

Wikipedia [en línea]. Code Injection. Disponible en Web: http://www.en.wikipedia.org/wiki/Code_injection, Recuperado Mayo 2009

Universidad Autónoma de Baja California [en línea]. La Productividad en la Informática, 2004. Disponible en Web: http://yaqui.mxl.uabc.mx/~halbarran/control.doc, Recuperado Mayo 2009

SAITTA, Larcom y EDDINGTON, Michael. A conceptual model for threat modeling Applications, 2005, Disponible en Web:
http://dymaxion.org/trike/Trike_v1_Methodology_Document-draft.pdf, Recuperado Mayo 2009

ISO27000 [en línea]. Sistema de Gestión de la Seguridad de la Información. Disponible en Web:
http://www.iso27000.es/doc_sgsi_all.htm, Recuperado Mayo 2009

Aspectsecurity,Security [en línea]. Verification Services. Disponible en Web:
http://www.aspectsecurity.com/press/pr_20061130.htm, Recuperado Mayo 2009

Wikipedia [en línea]. Arquitectura de tres niveles. Disponible en Web:
http://es.wikipedia.org/wiki/Arquitectura_de_tres_niveles, Recuperado Mayo 2009

Universidad de Oviedo [en línea]. Arquitectura Web. Disponible en Web:
http://www.di.uniovi.es/~dflanvin/docencia/dasdi/teoria/Transparencias/06.%20Arquitectura%20Web.pdf, Recuperado Mayo 2009

Ciberaula [en línea]. Arquitectura de Aplicaciones de 3 capas, 2008. Disponible en Web:
http://dotnetjunkies.com/WebLog/desarrollonet/archive/2004/06/17/16855.aspx, Recuperado Mayo 2009

Wikipedia [en línea]. Lenguaje unificado de modelado. Disponible en Web:
http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado, Recuperado Mayo 2009

Wikipedia [en línea]. Programación extrema. Disponible en Web:
http://es.wikipedia.org/wiki/Programaci%C3%B3n_Extrema, Recuperado Mayo 2009

Wikipedia [en línea]. Microsoft Framework. Disponible en Web:
http://en.wikipedia.org/wiki/Microsoft_Solutions_Framework, Recuperado Mayo 2009

Wikipedia [en línea]. SCRUM. Disponible en Web: http://es.wikipedia.org/wiki/Scrum, Recuperado Mayo 2009, Recuperado Mayo 2009

Microsoft [en línea]. Threat Modeling Web Applications, 2005. Disponible en Web:
http://msdn.microsoft.com/security/securecode/threatmodeling/default.aspx?pull=/library/enus/dnpag2/html/tmwa.asp, Recuperado Mayo 2009

Wikipedia [en línea]. OpenSSL. Disponible en Web: http://es.wikipedia.org/wiki/OpenSSL, Recuperado Abril 2009

Symantec [en línea]. Whitepaper Symantec Internet Threat report. Disponible en Web: http://www.symantec.com/specprog/threatreport/ent-whitepaper_symantec_internet_security_threat_report_x_09_2006.en-us.pdf, Recuperado Junio 2009

HOWARD Jhon D. “An Analysis Of Security on the Internet 1989-1995”. Carnegie Institute of Technology. Carnegie Mellon University. 1995, Recuperado Junio 2009

OWASP [en línea]. Code Injection Examples. Disponible en Web: http://www.owasp.org/index.php/Direct_Dynamic_Code_Evaluation_('Eval_Injection'), Recuperado Junio 2009

OWASP. Una Guía para Construir Aplicaciones y Servicios Web Seguros. 2005, pag 196, Recuperado Junio 2009

México. Constitución Política de los Estados Unidos Mexicanos. Disponible en Web: http://www.diputados.gob.mx/LeyesBiblio/pdf/1.pdf, Recuperado Julio 2009

México. Ley Federal de Derechos de Autor. Disponible en http://www.diputados.gob.mx/LeyesBiblio/regley/Reg_LFDA.pdf, Recuperado Julio 2009

Estados Unidos. United States Code Annotated, Crimes and Criminal Procedures. Disponible en Web: http://www.usdoj.gov/criminal/cybercrime/1030NEW.htm, Recuperado Septiembre 2009

GLOSARIO

.NET: Contrato: Un contrato, en términos generales, es definido como un acuerdo privado, oral o escrito, entre partes que se obligan sobre materia o cosa determinada, y a cuyo cumplimiento pueden ser exigidas. Es un acuerdo de voluntades que genera derechos y obligaciones para las partes. Por ello se señala que habrá contrato cuando varias partes se ponen de acuerdo sobre una manifestación de voluntad destinada a reglar sus derechos.

Auditoría: La palabra Auditoría viene del latín AUDITORIUS, y de esta proviene auditor, que tiene la virtud de oír.

Auditoría informática: Proceso de recoger, agrupar y evaluar evidencias para determinar si un Sistema de Información salvaguarda el activo empresarial, mantiene la integridad de los datos, lleva a cabo eficazmente los fines de la organización y utiliza eficientemente los recursos.

Code Inyection: Seguridad informática, código malicioso es un término que hace referencia a bhcualquier conjunto de códigos, especialmente sentencias de programación, que tiene un fin malicioso. Esta definición incluye tanto programas malignos compilados, como macros y códigos que se ejecutan directamente, como los que suelen emplearse en las páginas Web (scripts).

Contrato: Un contrato, en términos generales, es definido como un acuerdo privado, oral o escrito, entre partes que se obligan sobre materia o cosa determinada, y a cuyo cumplimiento pueden ser exigidas. Es un acuerdo de voluntades que genera derechos y obligaciones para las partes. Por ello se señala que habrá contrato cuando varias partes se ponen de acuerdo sobre una manifestación de voluntad destinada a reglar sus derechos.

Cookie: Es un fragmento de información que se almacena en el disco duro del visitante de una página Web a través de su navegador, a petición del servidor de la página. Esta información puede ser luego recuperada por el servidor en posteriores visitas.

Cross Site Scripting: XSS, del inglés Cross-site scripting es un tipo de inseguridad informática o agujero de seguridad basado en la explotación de vulnerabilidades del sistema de validación de HTML incrustado.

Framework: Un Framework, en el desarrollo de software, es una estructura de soporte definida, mediante la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente,

puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

ISO27001: El estándar para la seguridad de la información ISO/IEC 27001 (Information technology - Security techniques - Information security management systems - Requirements) fue aprobado y publicado como estándar internacional en Octubre de 2005 por International Organization for Standardization y por la comisión International Electrotechnical Commission.

ITIL: La Biblioteca de Infraestructura de Tecnologías de Información, frecuentemente abreviada ITIL (del inglés Information Technology Infrastructure Library), es un marco de trabajo de las buenas prácticas destinadas a facilitar la entrega de servicios de tecnologías de la información (TI). ITIL resume un extenso conjunto de procedimientos de gestión ideados para ayudar a las organizaciones a lograr calidad y eficiencia en las operaciones de TI. Estos procedimientos son independientes del proveedor y han sido desarrollados para servir como guía que abarque toda infraestructura, desarrollo y operaciones de TI.

Java: Java es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria.

Mejores prácticas: Por mejores prácticas se entiende un conjunto coherente de acciones que han rendido buen o incluso excelente servicio en un determinado contexto y que se espera que, en contextos similares, rindan similares resultados. Las mejores prácticas (best practices, en inglés) dependen de las épocas, de las modas y hasta de la empresa consultora o del autor que las preconiza. No es de extrañar que algunas sean incluso contradictorias entre ellas.

Outsourcing: Outsourcing o Tercerización (también llamada subcontratación) es una técnica innovadora de administración, que consiste en la transferencia a terceros de ciertos procesos complementarios que no forman parte del giro principal del negocio, permitiendo la concentración de los esfuerzos en las actividades esenciales a fin de obtener competitividad y resultados tangibles.

DDoS: En seguridad informática, un ataque de denegación de servicio, también llamado ataque DoS (de las siglas en inglés Denial of Service), es un ataque a un sistema de computadoras o red que causa que un servicio o recurso sea inaccesible a los usuarios legítimos. Normalmente provoca la pérdida de la conectividad de la red por el consumo del ancho de banda de la red de la víctima o sobrecarga de los recursos computacionales del sistema de la víctima.