INSTITUTO TECNOLOGICO Y DE ESTUDIOS SUPERIORES
DE MONTERREY

Campus Cd. de México

Escuela de Graduados en
Ingeniería y Arquitectura

Seguridad de acceso al código (CAS)

Maestría en Comercio Electrónico

Proyecto presentado por

Luis Felipe Pérez Navarro

[redacted]

México, D. F. Febrero de 2005

Índice

Índice de figuras

Índice de Tablas

Capítulo 1. Introducción

1.1 Antecedentes

Actualmente, las medidas de seguridad para determinar el grado de riesgo que le supone a la computadora de un usuario el instalar alguna aplicación o abrir un archivo que le fue enviado, están basadas en el concepto de advertir al usuario mediante cuadros de diálogo, sobre las posibles consecuencias de instalar dichas aplicaciones o archivos. Los exploradores Web y las aplicaciones de correo electrónico tales como Microsoft Internet Explorer y Microsoft Exchange, han desarrollado algunas medidas de seguridad al ofrecer ciertos mensajes de advertencia cuando se va a abrir un anexo o navegar a una página Web que intente ejecutar código en nuestro sistema de cómputo. En la figura 1.1, se muestra un ejemplo de dichos mensajes de advertencia:

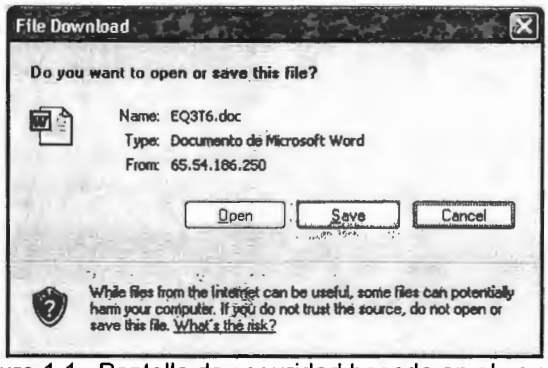

Figura 1.1 . Pantalla de seguridad basada en el usuario

Este tipo de seguridad se conoce en la literatura como seguridad basada en el usuario [Seara, 2003] y resulta claramente ineficaz, porque la intervención humana es necesaria para que logre su objetivo, que es el de prevenir sobre la instalación de software dañino en el equipo del usuario.

En función de las limitantes de la seguridad basada en el usuario, se ha desarrollado un nuevo esquema de protección para limitar las operaciones y acciones de las aplicaciones de software en los equipos de cómputo. Este nuevo esquema de protección para código de software no autorizado, está basado en dotar de ciertos permisos de ejecución y acceso a recursos a las mismas aplicaciones de software. Así como en la mayoría de los sistemas operativos, cada usuario del sistema tiene una serie de permisos que definen su rango de acción y acceso a recursos, de la misma forma, los nuevos esquemas de seguridad proporcionan permisos a las aplicaciones de software [Seara, 2003]. Dichos esquemas se conocen como seguridad de acceso al código (CAS) [Robinson, 2003].

De acuerdo a la visión del director de programas en Microsoft, Ed Robinson, la seguridad de acceso al código es una tecnología que nos permite como humanos, interactuar con aplicaciones de una manera natural, sin tener que tomar decisiones de seguridad en nombre de las aplicaciones que estamos ejecutando. Es decir, podremos abrir con total libertad los anexos de los mensajes de correo electrónico y acceder a sitios Web sin temer que toda la información personal contenida en nuestro equipo va a ser enviada a un atacante, o que nuestro equipo va a comenzar a remitir de forma espontánea un mensaje de correo electrónico a todos nuestros contactos.

1.2 Definición del problema

Hoy en día, la información almacenada en los sistemas informáticos constituye uno de los activos más importantes de una organización. En este sentido, la seguridad en cómputo es un área de las tecnologías de información que ha aumentado su importancia en los últimos años. Una definición formal sobre el significado de la seguridad en cómputo, establece que son *aquellos recursos o actividades destinados a lograr que los activos de una organización sean confidenciales, íntegros, disponibles a sus usuarios, autentificados por mecanismos de control de acceso y sujetos a auditoria* [Seara, 2003]. En términos generales, las tecnologías de información presentan vulnerabilidades en cuatro áreas:

- Ingreso no autorizado a una red.
- Robo de información.
- Modificación de datos.
- Ejecución de acciones no autorizadas y de riesgo por parte de código de software.

El presente trabajo, se enfoca al problema de la ejecución de acciones de riesgo y no autorizadas por parte del código de software, el cual ocurre en el siguiente contexto: Los sistemas de cómputo, ya sean computadoras personales o servidores, ejecutan aplicaciones o programas de diversa índole y origen. En este orden de ideas, el fenómeno del Internet ha acentuado la práctica entre los usuarios de instalar en su equipo mediante un "download", software de todo tipo [Conry, 2002]. Cabe señalar que Internet, representa un nuevo canal de distribución para software que puede acceder a las redes de datos privadas o computadoras personales mediante:

- Archivos adjuntos al correo electrónico.
- Enlaces en sesiones de chat.
- Programas de tipo *cookies* contenidos en páginas Web.
- Fallas de seguridad en los sistemas operativos de los servidores y computadoras personales.

Muchas de estas aplicaciones distribuidas por Internet, tienen la intención de propagar un virus o ejecutar una acción dañina en una determinada red de datos o

equipo de cómputo. Las aplicaciones, están desarrolladas en lenguajes de programación que contienen a métodos y funciones que de ser programados en forma mal intencionada, pueden ejecutar acciones en los subsistemas del sistema operativo del equipo de cómputo o bien, pueden alterar para otros fines, el código de aplicaciones que ya residen en dicho equipo. En este sentido, los subsistemas del sistema operativo, las bibliotecas de clases y las aplicaciones de software, están expuestas a posibles manipulaciones y acciones por parte de aplicaciones de software que de alguna manera lograron instalarse en la red o equipo de cómputo.

1.3 Objetivo

Cabe señalar que las aplicaciones (incluida la generación de virus) y el desarrollo de componentes, se programan mediante plataformas de desarrollo de software (SDK). Las de mayor uso hoy en día son las plataformas .NET y Java, desarrolladas por Microsoft en el primer caso y Sun MicroSystems en el segundo caso.

La plataforma .NET de la empresa Microsoft ha desarrollado un modelo de seguridad basado en seguridad de acceso al código para restringir las acciones y el acceso a recursos por parte de las aplicaciones desarrolladas con lenguajes .NET. Este modelo consiste en prevenir que las bibliotecas de clase u objetos de una aplicación puedan ser llamados y ejecutados sin el conocimiento del programador, usuario o administrador del sistema, por una secuencia de comandos con suficientes privilegios. Las técnicas CAS , desde el punto de vista del esquema de seguridad de la plataforma .NET, parten de la base de analizar la procedencia del código o aplicación que está llamando a una función o clase que reside en las bibliotecas del lenguaje de programación .NET. o que reside en los subsistemas del sistema operativo.

Por otra parte, la plataforma Java también cuenta con una serie de mecanismos de seguridad cuyo fin es el de prevenir el acceso a programas de código malicioso que pudiesen manipular a las bibliotecas de clase del sistema. En este orden de ideas, cada aplicación desarrollada en el lenguaje Java, tiene la capacidad de definir sus propias políticas de seguridad mediante la implantación de un administrador de seguridad "security management", cuya función es proteger al sistema y las bibliotecas de clase, con el fin de definir el ámbito de cada programa Java en cuanto a las capacidades, permisos y privilegios para acceder a ciertos recursos. Este modelo de seguridad original proporcionado por la plataforma Java, es conocido como la "caja de arena" (*sandbox*), y consiste en proporcionar un ambiente de ejecución muy restrictivo para código no confiable que haya sido obtenido de las redes de datos públicas, como es el caso de la Internet [Menchaca, 2000]. El mecanismo principal del modelo de la caja de arena, parte de la base de que el código obtenido del sistema de archivo local es por naturaleza confiable por lo que se le permite el acceso a los recursos del sistema, como el mismo sistema de archivos o a los puertos de comunicación. Por otra parte, el código obtenido de las redes de datos públicas se considera no confiable y por lo tanto, solo tiene acceso ciertos recursos. En este orden de ideas, en la plataforma Java, el acceso a los

recursos importantes del sistema es administrado entre el sistema de tiempo de ejecución y el administrador de seguridad (*Security Manager*), que es implementado por la clase *java.lang.SecurityManager* [Joachim, 1997]. De esta manera, es posible para el sistema determinar si una operación de código de Java es insegura o contraviene las políticas de seguridad, antes de ejecutarla.

En la literatura sobre seguridad para el desarrollo de software .NET, existe documentación sobre diversos aspectos del esquema de seguridad CAS. Sin embargo, en dicha documentación solo se aborda de manera parcial las diversas técnicas que conforman a la seguridad de acceso a código. Esta situación pone de manifiesto la necesidad de implementar una guía o documentación que proporcione a los programadores un proceso sistemático para entender el modelo de seguridad CAS y las distintas estrategias para desarrollar software en un entorno de seguridad de acceso al código. El objetivo del presente trabajo de investigación consiste en describir los lineamientos de seguridad CAS inherentes a la plataforma de desarrollo de software .NET, y proporcionar un documento a los desarrolladores de software y administradores de sistemas para en el primer caso, desarrollar la aplicación y sus acceso a recursos críticos en función del modelo de seguridad CAS, y en el segundo caso, implementar mecanismos que confinen la ejecución de código a un entorno definido por los administradores del sistema.

1.4 Hipótesis

Los sistemas operativos actuales basan su modelo de seguridad, en permitir o denegar ciertas acciones a una aplicación de software, en función de los permisos que tenga o el grupo al que pertenezca el usuario que ejecuta dicha aplicación [O'Reilly, 2003]. Sin embargo la plataforma .NET también proporciona un mecanismo de seguridad de código en donde, además de considerar los permisos que le han sido asignados al usuario por parte del sistema operativo, también se definen permisos al código que compone la aplicación para el acceso de recursos críticos.

El personal de los departamentos de tecnologías de información en las compañías, así como los usuarios de computadoras personales carecen del tiempo y de los recursos en el primer caso, y del conocimiento en el segundo caso para evaluar de forma particular a cada aplicación o código que se instala en el sistema operativo de la computadora del usuario, y por lo tanto es necesario desarrollar una documentación que fomente la cultura del modelo de seguridad basado en CAS para el control de las aplicaciones de software que se ejecutan en las computadoras o redes privadas. La hipótesis que se establece en el presente trabajo, estriba en que los administradores de red y desarrolladores de software, al ejecutar aplicaciones .NET bajo el esquema de seguridad CAS tendrán un entorno de mayor seguridad en sus equipos de cómputo. Con el fin de comprobar la hipótesis anterior, en el presente trabajo de tesis, se desarrolló un método de investigación para conocer en lo general las estrategias de seguridad informáticas que se utilizan en las organizaciones del medio nacional, y en lo particular, para identificar el rol de CAS en dichas estrategias.

1.5 Justificación

En el contexto de las aplicaciones de comercio electrónico, *la seguridad es un componente real del proceso, no un opcional condicionado a tiempos y costos* [Ardita, 2000]. De acuerdo a la liga de consumidores de los Estados Unidos, durante el 2003, los problemas causados por código malicioso o virus tuvieron un impacto económico en las empresas por 320 millones de dólares [Sánchez, 2005]. En el mundo real, los usuarios de equipos de cómputo están expuestos a todo tipo de código malicioso a pesar de contar con las últimas actualizaciones en materia de seguridad para el software de sistema operativo y para el antivirus. De acuerdo a los expertos en seguridad de la empresa *Symantec, la solución antivirus no es suficiente para detectar con un porcentaje infalible de seguridad, a los programas de código malicioso que diariamente surgen en la red.* De tal suerte que en función del problema descrito en la sección de definición del problema, existe la necesidad de establecer mecanismos de seguridad para establecer restricciones a las acciones que las aplicaciones de software pueden ejecutar en los sistemas de cómputo, ya que hoy en día las plataformas de tecnologías de la información (TI), conformadas por servidores, sistemas operativos, aplicaciones y redes, están expuestas a código o programas con origen de diversas fuentes.

1.6 Limitaciones del proyecto

La seguridad informática es un tema amplio, que comprende una metodología en donde es abordan aspectos como: encriptación, autenticación, autorización, seguridad en redes, entre otras. El contexto de la seguridad informática actual, está definido por un modelo formado por la Microsoft Security Task Force (STRIDE) [Robinson, 2003]. Dicho modelo está compuesto por los siguientes puntos:

- Suplantación de la identidad del usuario: El atacante realiza todas las operaciones que pudiera llevar a cabo el usuario auténtico.
- Alteración de los datos: Cualquier dato que pudiera modificar el atacante sin la autorización del usuario legítimo.
- Repudio: Cuando una de las partes niega haber realizado una acción y la otra parte es incapaz de demostrarlo.
- Revelación de información: Cualquier ataque en el que el atacante pueda obtener información sin autorización.
- Denegación al servicio: Afectación de los módulos de operación del sistema operativo.
- Elevación de privilegios. Obtención de más privilegios de los que debiera de tener el usuario.

Cabe señalar que la tecnología de seguridad de acceso a código es una técnica enfocada a daños en el código de los subsistemas del sistema operativo por lo que se enfoca a prevenir la alteración de datos por parte de código dañino. En este orden de ideas el presente trabajo de campo, tiene por limitante proponer una

estrategia o metodología integral de seguridad informática en función de que se deberían tratar otros temas como lo son: autorización, autenticación, servidores "proxy" y "firewall". El enfoque de este trabajo es en específico al problema de la alteración de datos por parte del código dañino.

Capítulo 2. Marco teórico

2.1 Introducción

Tal como se señalo, en la sección de justificación del presente trabajo, los actuales sistemas de equipos conectados suelen estar expuestos a código procedente de orígenes diversos y, posiblemente, desconocidos. El código puede asociarse a mensajes de correo electrónico, incluirse en documentos o descargarse de la red Internet. Dicho código puede contener errores o puntos vulnerables que hacen que pueda ser atacado por código malicioso y, en ocasiones, el código realiza acciones que el usuario desconoce. En consecuencia, los sistemas informáticos pueden resultar dañados y se puede filtrar información confidencial cuando los sistemas de cómputo de los usuarios ejecutan software malicioso o lleno de errores. La mayoría de los mecanismos de seguridad de los sistemas operativos requieren que cada fragmento de código sea de plena confianza para ejecutarse, con excepción quizás de las secuencias de comandos de una página Web. Por consiguiente, sigue habiendo la necesidad de un mecanismo de seguridad de amplia aplicación que permita que el código procedente de un sistema informático se ejecute con protección en otro sistema, incluso cuando no hay ninguna relación de confianza entre ambos. La propuesta de seguridad para abordar el problema de código malicioso en la plataforma .NET se conoce como seguridad de acceso al código.

La seguridad de acceso al código ha sido diseñada para restringir a las aplicaciones de software .NET y componentes (macros, archivos ejecutables, archivos .dll, controles ActiveX) que se encuentran en entornos compartidos tales como las redes de área local o Internet, de ejecutar alguna de las siguientes acciones debido a una manipulación cuyo fin sea el de hacer daño:

- Dañar o destruir de forma inadvertida o intencionada datos sensibles.
- Bloquear el equipo en el que se está ejecutando el código consumiendo todos sus recursos disponibles, tal como los diversos tipos de memoria temporal o el espacio en disco duro. Dicho ataque se conoce como *Denegación de servicio (Dos)* [Seara, 2003].
- Elevar el nivel de privilegio del código de forma intencionada o no que ha sido llamado por el usuario, para llevar a cabo acciones tales como acceder a la información sensible de los usuarios almacenada en el equipo en el que se está ejecutando el código. Dicho ataque se conoce como *ataque de atracción* [Seara, 2003].

2.1.1 Ejemplo

La plataforma de desarrollo de software .NET proporciona un componente llamado *Chart* que puede ser utilizado por los programadores para incorporar funciones tales como mostrar gráficos en distintos formatos. Sin embargo, el componente *Chart* contiene un método llamado *SaveChart* que de ser manipulado en forma mal intencionada, puede pasar información sobre la ruta de acceso de la

aplicación que contiene al componente, o sobre cualquier otro archivo del sistema con el fin de destruirlo o alterarlo. En este sentido, el sistema de seguridad de acceso al código deberá detectar que un llamador que no es de confianza está intentando llamar a un método que carece de autorización y provocar una excepción de seguridad para impedir que la acción se desarrolle.

2.2 La plataforma .NET

2.2.1 Seguridad de acceso al código en el .NET Framework

El .NET Framework es un conjunto de objetos y diseños de la compañía Microsoft para crear aplicaciones de software. Este marco de trabajo está formado por el Common Languaje RunTime (CLR) que funciona como una gran librería de clases unificada, que contiene a todas las clases que funcionan dentro del entorno .NET. Además, el CLR es un entorno que administra la ejecución del código y lo compila al lenguaje intermedio de Microsoft (MSIL) [Payne, 2002]. El Common Languaje Runttime del .NET Framework administra la seguridad de acceso al código de las aplicaciones desarrolladas mediante lenguajes .NET. En este orden de ideas, para determinar si el código de una aplicación .NET tiene autorización para el acceso a un recurso o ejecutar una operación, el CLR recorre la pila de llamadas y compara los permisos concedidos a cada llamador con el permiso que se exige. Si algún llamador de la pila de llamadas no tiene el permiso exigido, se iniciará una excepción de seguridad y se rechazará el acceso. El recorrido de pila está diseñado para ayudar a evitar los ataques trampa, en los cuales código de menor confianza llama a código de gran confianza y lo utiliza para realizar acciones no autorizadas [Panagrasso, 2002]. La solicitud de los permisos de todos los llamadores en tiempo de ejecución afecta al rendimiento, pero es esencial para ayudar a proteger el código de estos ataques por parte de código de menor confianza.

En el .NET Framework, la seguridad de acceso a código realiza las siguientes funciones:

- Define permisos y conjuntos de permisos que representan el derecho de acceso a varios recursos del sistema.
- Permite a los administradores configurar una directiva de seguridad mediante la asociación de conjuntos de permisos a grupos de código.
- Permite que el código solicite los permisos que requiere para ejecutarse así como los permisos que serían útiles tener, además de especificar qué permisos nunca debe tener.
- Concede permisos a cada ensamblado que se carga, basándose en los permisos solicitados por el código y en las operaciones permitidas por la directiva de seguridad.
- Permite que el código exija que sus llamadores tengan permisos específicos.

- Permite al código exigir que sus llamadores posean una firma digital, por lo que sólo los llamadores de una organización o un sitio concretos pueden llamar al código protegido.
- Impone restricciones en el código en tiempo de ejecución mediante la comparación de los permisos concedidos a cada llamador en la pila de llamadas con los permisos que los llamadores deben poseer.

2.2.2 Asignación de permisos y control en la ejecución de código dañino

El sistema de seguridad de acceso al código de .NET asigna a la aplicación o componente permisos tales como acceso a archivos, interfaz de usuario y de red, como base para determinar qué operaciones seguras o inseguras puede realizar la aplicación [Payne, 2002]. Los permisos asignados a una aplicación se establecen en función del nivel de confianza asignado a la misma. Para el caso de las aplicaciones instaladas mediante un programa de instalación y ejecutadas en el propio equipo de cómputo se consideran como aplicaciones de confianza elevada por lo que tendrán todos los permisos disponibles. Por otra parte, los componentes que se cargan y ejecutan desde una red de datos pública, como lo es Internet, tendrán el status de desconfianza elevada y luego entonces tendrán menos permisos. El sistema de seguridad de acceso al código .NET utiliza medios sofisticados, para determinar los permisos que se asignarán a las componentes o aplicaciones basadas en los lenguajes de programación propios de .NET.

En el caso de que una aplicación de software se ejecute directamente desde una red pública, tal como lo es Internet, el sistema de seguridad de acceso al código impide que el código dañino se ejecute al verificar primero si el código tiene permiso para realizar una determinada operación tal como eliminar archivos. Por ejemplo, si al navegar por un portal de Internet que contiene componentes gráficos se ejecuta una instrucción propia del lenguaje Visual Basic. Net, como lo es *Kill* con el fin de eliminar un archivo, dicha instrucción deberá solicitar un permiso para eliminar archivos, y si el sistema CAS no lo autoriza entonces no se le concederá este permiso y se iniciará una excepción de seguridad en el sistema de cómputo. Es decir, la plataforma .NET de manera predeterminada activa la verificación de ciertas funciones y métodos propios de los lenguajes de programación .NET. Por ejemplo, para el caso de Visual Basic.NET, las aplicaciones suelen incluir métodos tales como *FileOpen, Kill, Shell y Show,* los cuales se verifican internamente cuando una aplicación intente ejecutar dichos métodos y en caso de no contar con los permisos necesarios para realizar la acción requerida, se iniciará una excepción de seguridad y la acción no tendrá lugar. La principal característica de la seguridad de acceso al código es que el propio sistema fuerza la seguridad de acceso al código, en concreto, el runtime del .NET. En este sentido CAS determina automáticamente mediante valores predeterminados las operaciones que puede realizar el código.

2.2.3 Zonas de seguridad y niveles de confianza

Tal como se ha señalado en la sección anterior, los permisos son la base de la plataforma .NET. En este orden de ideas, cuando se ejecuta una aplicación que requiere acceder a un archivo en un entorno de red, las acciones permitidas a dicha aplicación, dependerán de la ubicación desde la cual se ejecute. En el momento en que .NET ejecuta a una aplicación, calcula los permisos en función de una serie de factores, tales como la zona desde la que se ejecuta la aplicación. Cada zona tiene asignado un nivel de confianza predeterminado . En la tabla 2.1 se enlistan las zonas disponibles y sus niveles de confianza predeterminados.

Zona	Opción predeterminada para la confianza.	Descripción
Mi Pc	Plena confianza	La aplicación puede hacer lo mismo que el usuario actual del equipo.
Intranet local	Confianza media.	Permisos limitados, tal como el derecho a leer y escribir archivos desde un lugar especial y aislado en disco duro de la PC.
Sitios de confianza	Confianza baja	Permisos limitados para la interfaz de usuario, impresión y ejecución.
Internet	Confianza baja o No se confía	Los permisos dependen de la versión de .NET Framework que se haya instalado.
Sitios que no son de confianza	No se confía	Sin permisos. No se permite la ejecución del código

Tabla 2.1 Zona disponible y valor de confianza predeterminado

La zona *Mi PC*, en la que se encuentran todas las aplicaciones .NET que se ejecutan desde la unidad de disco duro u otro dispositivo similar, es la zona de confianza más elevada. Esta zona suele tener asignado el valor de *Plena confianza*, lo que significa que cualquier aplicación que se ejecute desde dicho sistema de cómputo podrá acceder a todos los recursos que le han sido permitidos al usuario actual.

Al igual que existe un nivel de confianza asociado a cada zona, existe también un conjunto de permisos asociado con cada nivel de confianza. Cuanto mayor sea el nivel de confianza, mayores serán los permisos concedidos. Por el contrario, en cuanto menor sea el nivel de confianza, se tendrán menos permisos y en algunos casos no habrá permiso alguno. En la tabla 2.2 se enlistan los permisos asociados a cada zona.

Nivel de permisos	Mi PC	Intranet local	Internet y Sitios de confianza	Sitios que no son de confianza	Permite a una apli-cación
DnsPermission	Permitido	Permitido	Denegado	Denegado	Realizar operaciones de Sistema de Nombres de Dominio tal como convertir un nombre URL en una dirección IP.
EventLogPermission	Permitido	Denegado	Denegado	Denegado	Leer o escribir de / en el registro de sucesos.
EnvironmentPermission	Permitido	Denegado	Denegado	Denegado	Leer o escribir variables de entorno.
FileDialogPermission	Permitido	Permitido	Denegado	Denegado	Mostrar los cuadros de diálogo Abrir archivo y Guardar.
FileIOPermission	Permitido	Denegado	Denegado	Denegado	Leer, escribir o anexar archivos.
IsolatedStorageFilePermission	Permitido	Denegado	Denegado	Denegado	Leer o escribir datos en un sitio especial y reservado en el disco duro de la PC
PrintingPermission	Permitido	Denegado	Denegado	Denegado	Conectarse con impresoras locales o de red.
ReflectionPermission	Permitido	Denegado	Denegado	Denegado	Consultar las clases, módulos, propiedades , métodos y sucesos que forman la aplicación.
RegistryPermission	Permitido	Denegado	Denegado	Denegado	Leer y escribir en el registro del sistema

Tabla 2.2 . Permisos asociados a cada zona

En la tabla 2.3 se enlistan exclusivamente los permisos otorgados de forma predeterminada a la zona MiPC y que no se asignan a ninguna otra zona.

Nivel de permiso	Permite a una aplicación
DirectoryServicesPermission	Examinar, leer y escribir las entradas de Active Directory.
MessageQueuePermission	Localizar las colas de mensajes disponibles, leer los mensajes contenidos en la cola o enviar y recibir mensajes.
OleDbPermission	Acceder a un proveedor de OleDb o definir una contraseña en blanco en la cadena de conexión
PerformanceCounterPermission	Localizar, modificar o crear categorías de contadores de rendimiento.
ServiceControllerPermission	Localizar, activar o desactivar servicios de Windows tales como el servicio SQL.
SocketPermission	Leer desde o escribir en una determinada red basándose en el nombre del host, puerto y transporte.
SqlClientPermission	Utilizar una contraseña en blanco como parte de la cadena de conexión.
WebPermission	Aceptar datos o transmitir datos desde / hacia un determinado URL.

Tabla 2.3 . Permisos de la zona MiPC

Capítulo 3. Documentación / guía para implementar el modelo de seguridad CAS

3.1 Introducción

En esta sección, se exponen una serie de técnicas y situaciones que un desarrollador de .NET debe de considerar para que su aplicación interactué en forma correcta con el modelo de seguridad CAS. Se parte de la base de que las aplicaciones .NET son administradas por el Common Language Runtime, lo que implica la interacción en tiempo real de dichas aplicaciones con la seguridad administrada del .NET Framework [Lind, 2003]. En este sentido, el CLR evalúa automáticamente a la aplicación y le concede un conjunto de permisos. En función de los permisos que reciba la aplicación, esta se ejecutará correctamente o generará una excepción de seguridad . La configuración de seguridad local de un equipo concreto decide en última instancia los permisos que recibe el código. Dado que esta configuración puede variar de un equipo a otro, nunca se podrá saber con certeza si el código va a recibir los permisos suficientes para ejecutarse. Esta situación no se presenta en la programación no administrada, en donde no es necesario preocuparse por los permisos que requiere el código para ejecutarse. En este capítulo se describen los siguientes puntos como una guía para el programador de aplicaciones .NET y el administrador de estrategias de seguridad para implementar CAS en sus respectivos entornos:

1. Implementar seguridad de tipos.
2. Sintaxis de seguridad.
3. Solicitar permisos.
4. Utilizar bibliotecas de clase seguras.

Además, en este capítulo se exponen los siguientes temas complementarios de la seguridad de acceso al código:

- Determinación de la zona por parte de las aplicaciones .NET
- Distribución de la aplicación .NET
- Modificación de las directivas de seguridad.
- Utilización del almacenamiento aislado.

3.2 Guía para el programador de aplicaciones .NET

3.2.1 Implementar seguridad de tipos

El código con seguridad de tipos es aquel que sólo obtiene acceso a datos siguiendo métodos permitidos y perfectamente definidos. Por ejemplo, dada una referencia a objeto válida, el código con seguridad de tipos solo podrá tener acceso al objeto si utiliza un método definido para ubicar a dicho objeto en la memoria [Cassidy, 2002]. El Common Language Runtime ejecuta un proceso denominado comprobación que examina el código e intenta determinar si tiene seguridad de

tipos. Si se demuestra durante la comprobación que el código tiene seguridad de tipos, se denominará código con seguridad de tipos comprobable. El código puede tener seguridad de tipos pero no ser comprobable debido a las limitaciones del proceso de comprobación o del compilador. No todos los lenguajes tienen seguridad de tipos y algunos compiladores de lenguaje, como Microsoft Visual C++, no pueden generar código administrado con seguridad de tipos comprobable. El administrador de sistema deberá implementar la directiva de seguridad que habilita la seguridad de tipos. De esta forma las aplicaciones desarrolladas en compiladores de código no administrado no se ejecutarán en el sistema. Así mismo, para el caso de código obtenido de las redes de datos públicas, tales como la Internet, siempre se deberá habilitar la seguridad de tipos comprobable como parte de las técnicas CAS.

3.2.2 Sintaxis de seguridad

El desarrollador de aplicaciones .NET al escribir sus clases o componentes podrá exigir que los usuarios o aplicaciones que utilicen sus clases tengan ciertos permisos. De esta forma, el desarrollador puede controlar el alcance de sus componentes para el uso de recursos o acciones en el sistema, en función del tipo de llamador que haga uso de su componente, clase o aplicación. En este orden de ideas, el código administrado del CLR puede interactuar con el sistema de seguridad al solicitar permisos, exigir que los llamadores tengan los permisos especificados y reemplazar determinados valores de seguridad (si dispone de privilegios suficientes). El programador de aplicaciones puede especificar los permisos que su componente requiere desde el mismo código. Para interactuar mediante programación con el sistema de seguridad de .NET Framework, se utilizan dos formas de sintaxis diferentes: la sintaxis declarativa y la sintaxis imperativa [RockWell, 2001]. Algunas operaciones se pueden realizar mediante las dos formas de sintaxis, mientras que otras sólo se pueden llevar a cabo mediante la sintaxis declarativa.

La sintaxis de seguridad declarativa utiliza atributos para colocar la información de seguridad en los metadatos del código. Los atributos se pueden colocar en el nivel de ensamblado, nivel de clase o nivel de miembro para indicar el tipo de solicitud, petición o reemplazo que se desea utilizar. Las solicitudes se utilizan para informar al sistema de seguridad del motor de tiempo de ejecución (CLR) sobre los permisos que la aplicación necesita o no desea. Las peticiones y los reemplazos se utilizan en las bibliotecas para ayudar a proteger los recursos ante los llamadores o para reemplazar el comportamiento de seguridad predeterminado. Por otra parte, la sintaxis de seguridad imperativa emite una llamada de seguridad mediante la creación de una nueva instancia del objeto de permiso que se desea invocar. La sintaxis imperativa se puede utilizar para realizar peticiones y reemplazos, pero no para realizar solicitudes

3.2.3 Solicitar permisos

Tal como se señalo en el apartado anterior, la práctica de incluir permisos en los componentes desarrollados le permiten al programador, asegurarse de que su componente no será manipulado por una aplicación de código malicioso. Por otra parte, el administrador de sistemas, al ser el responsable de establecer las directivas de seguridad, deberá estar relacionado con los permisos que requieren las aplicaciones .NET.

Cuando una aplicación .NET solicita permisos, en realidad se le está comunicando al CLR lo que el código necesita que se le permita hacer. Para solicitar permisos para una aplicación, se colocan atributos (sintaxis declarativa) en el ámbito del ensamblado del código. En tiempo de carga, el CLR del .NET Framework examina las solicitudes de permiso y aplica reglas de directiva de seguridad para determinar qué permisos debe conceder al ensamblado [Fisher, 2002]. Las solicitudes sólo influyen en el motor de tiempo de ejecución para rechazar permisos al código, nunca para concederle más permisos. La directiva de administración local siempre tiene el control final de los permisos máximos que se conceden al código. Aunque el código no tenga que solicitar permisos para la compilación, hay razones importantes por las que el código siempre debería solicitar permisos:

- Al solicitar permisos, aumenta la probabilidad de que el código se ejecute correctamente si se le permite que lo haga. El código que solicita un conjunto mínimo de permisos no se ejecutará, salvo que reciba esos permisos. Si no se identifica un conjunto de permisos mínimos, el código debe controlar discretamente cualquier situación y todas las situaciones en las que, si no se le hubiera concedido ningún permiso, no hubiera podido ejecutarse correctamente.
- Solicitar permisos ayuda a garantizar que se conceden al código sólo los permisos que necesita. Si no se conceden permisos adicionales al código, no puede dañar los recursos protegidos por esos permisos adicionales, aunque sea atacado por código malicioso o tenga errores que puedan aprovecharse para dañar los recursos. Deben solicitarse sólo aquellos permisos que el código necesita y ninguno más.
- Solicitar permisos permite a los administradores conocer los permisos mínimos que necesita la aplicación de modo que puedan ajustar la directiva de seguridad en consecuencia. Los administradores utilizan la herramienta de vista de permisos "PermView.exe" para examinar los ensamblados y configurar la directiva de seguridad de modo que se emitan los permisos necesarios. Si no se solicitan explícitamente los permisos que requiere la aplicación, la herramienta vista de permisos no puede devolver ninguna información sobre dichos permisos. Si un administrador no conoce esta información, la aplicación resulta difícil de administrar.

Al solicitar permisos, se informa al motor de tiempo de ejecución sobre qué permisos necesita la aplicación para funcionar o qué permisos específicos no necesita. Por ejemplo, si la aplicación escribe en el disco duro local sin utilizar el almacenamiento aislado, la aplicación deberá disponer de el permiso "FileIOPermission". Si el código no solicita "FileIOPermission" y la configuración de seguridad local no permite que la aplicación tenga este permiso, se iniciará una excepción de seguridad cuando la aplicación intente escribir en el disco. Incluso si la aplicación puede controlar la excepción, no podrá escribir en el disco. Si el código no tiene acceso a recursos protegidos ni realiza operaciones protegidas, no es necesario solicitar permisos. Por ejemplo, no es necesario realizar una solicitud de permisos si el código simplemente calcula un resultado basándose en las entradas que se le han pasado, sin utilizar ningún recurso. Si el código tiene acceso a recursos protegidos y no solicita los permisos necesarios, quizás aún pueda ejecutarse pero podría generar un error en algún momento de la ejecución si intentase obtener acceso a un recurso para el que no tiene el permiso necesario. Para solicitar permisos, es necesario saber qué recursos y operaciones protegidos utiliza el código, y también qué permisos protegen esos recursos y esas operaciones. Además, se ha de realizar un seguimiento de los recursos a los que tienen acceso los métodos de bibliotecas de clases a los que llaman los componentes.

3.2.4 Utilizar bibliotecas de clase seguras

Una biblioteca segura es una biblioteca de clases que utiliza peticiones de seguridad para garantizar que los llamadores de la biblioteca tienen permiso para obtener acceso a los recursos que ésta expone. Por ejemplo, una biblioteca de clases segura podría tener un método para crear archivos que requiera que los llamadores tengan permisos para crear archivos. El .NET Framework contiene bibliotecas de clases seguras, con lo cual, las aplicaciones o componentes desarrolladas en este plataforma, deberán solicitar los permisos respectivos para el uso de las clases en las bibliotecas.

Si el código solicita los permisos que exige la biblioteca de clases y se le conceden, podrá obtener acceso a la biblioteca y el recurso quedará protegido del acceso no autorizado; si el código no tiene los permisos correctos, no podrá tener acceso a la biblioteca de clases y el código malicioso no podrá utilizarlo para obtener acceso al recurso indirectamente. Incluso si el código tiene permiso de acceso a una biblioteca, no podrá ejecutarse si el código que llama a su código no tiene tampoco permiso para obtener acceso a la biblioteca. La seguridad de acceso a código no elimina la posibilidad de errores humanos al escribir código, sin embargo, si las aplicaciones utilizan bibliotecas de clases seguras para obtener acceso a los recursos protegidos, se reduce el riesgo de infracciones de seguridad en el código de aplicación porque las bibliotecas de clases se examinan detenidamente para detectar posibles problemas de seguridad.

3.3 Determinación de la zona por parte de las aplicaciones .NET

Si la aplicación se ejecuta desde una unidad de disco conectada al equipo local, la plataforma .NET identificará a la aplicación como ejecutada en la zona Mi PC. Para todas las demás zonas (tal como Intranet local, Sitios de confianza, Sitios que no son de confianza e Internet) los permisos que se asignen a la aplicación quedarán determinados por las opciones de zona del navegador Microsoft Internet explorer. De forma predeterminada, la zona Intranet local incluye los siguientes elementos:

- Los sitios de Intranet local no listados en otras zonas, como las zonas de sitios restringidos.
- Aquellos sitios que han sido configurados por el administrador de red para ignorar al servidor proxy.
- Todas las rutas de acceso a la red.

Los componentes de un lenguaje .NET que un programador codifique, tal como las bibliotecas de clase y los controles de usuario de Windows Forms, se encuentran asignados a una zona específica. En el caso de un componente, la zona se determina en función del lugar desde el que se genera el componente. Si el componente se genera directamente desde la Intranet o desde Internet, el componente se asignará a la zona respectiva. Si el componente se encuentra ya instalado en el equipo local, el componente quedará asignado a la zona Mi Pc [Seara, 2003].

3.4 Distribución de la aplicación .NET

Si la aplicación que un programador desarrolle necesita efectuar una operación crítica, tal como escribir en un archivo que no pertenezca a la zona desde la que se intenta ejecutar la aplicación, existen las siguientes opciones:

- Encontrar una forma alternativa de realizar la operación que funcione en la zona dada.
- Desplegar la aplicación para que se ejecute desde una zona distinta que tenga un mayor nivel de confianza.
- Modificar la directiva de seguridad para asignar a la aplicación el permiso adecuado.

La solución recomendable es cooperar con el sistema de seguridad .NET permitiendo que la aplicación se ejecute en la forma esperada. En este sentido, el diseño de las aplicaciones tiene que considerar la razón del porqué se asignan los permisos predeterminados a cada zona.. Es decir, las restricciones se han definido para impedir que las aplicaciones causen ningún daño de forma intencionada o sin intención.

Si la aplicación necesita más permisos de los que tiene asignados la zona, será necesario modificar la forma en que se distribuye la aplicación. Una posible solución es la de empaquetar la aplicación en un paquete de instalación (como lo hace Microsoft Installer, en un archivo .MSI). De manera que cuando un usuario instale la aplicación en su PC, esta se estará ejecutando desde la zona Mi PC con lo que obtendrá un nivel de plena confianza. Sin embargo, en este caso es recomendable unir una firma digital a la aplicación para que el usuario interesado en instalarla este seguro del origen de dicha aplicación.

3.5 Modificación de las directivas de seguridad

Otra alternativa para los desarrolladores de aplicaciones .NET es la de modificar las reglas o directivas de seguridad utilizadas por el sistema .NET, con el fin de obtener los permisos que la aplicación necesita. Sin embargo, los usuarios de la aplicación deberán aplicar estas mismas modificaciones en las directivas de seguridad de sus respectivos equipos. Los desarrolladores de aplicaciones podrán implementar las modificaciones que necesiten en un paquete que se le proporcionará a los usuarios y estos lo ejecutarán para obtener los permisos necesarios.

3.6 Utilización del almacenamiento aislado

En algunos casos, las aplicaciones que se ejecutan desde un recurso compartido de red, necesitan almacenar ciertas opciones. Estas opciones suelen almacenarse en el registro, pero en .NET una aplicación que se ejecuta desde un recurso compartido de red no tiene los permisos necesarios para leer y escribir las claves del registro. Por esta razón, el programador deberá diseñar su aplicación para que emplee un mecanismo distinto para almacenar opciones como por ejemplo, escribir estas opciones en un archivo contenido en el disco o en una base de datos. Sin embargo, si la aplicación se está ejecutando desde la zona de Intranet local, entonces está carecerá de los permisos para acceder al registro, a un archivo o una base de datos. En estos casos, el código deberá utilizar el almacenamiento aislado. El almacenamiento aislado es un área especial y reservada del disco duro administrado por .NET desde el que la aplicación puede leer o escribir información. En algunos casos, utilizar un almacenamiento aislado resulta una alternativa razonable a leer o escribir información en el disco o en el registro. En el caso de aplicaciones que se ejecuten en la zona Intranet local, se asigna y se mantiene una zona de almacenamiento independiente para cada uno de los usuarios que inicie sesión en el equipo. Esta forma de proceder impide que la aplicación pueda leer información sensible almacenada por otros usuarios y evita que la aplicación pueda sobrescribir archivos importantes que pertenezcan a otros usuarios o al sistema operativo.

Capítulo 4. Método de investigación

4.1 Introducción

El método de investigación tuvo por objetivo recabar las prácticas comunes en materia de seguridad que los programadores consideran para el desarrollo de sus aplicaciones, partiendo de la base, de que el modelo actual de seguridad que los programadores utilizan puede reforzarse y / o complementarse con el modelo de seguridad de acceso al código. En este orden de ideas, el método consistió en la realización de una investigación cualitativa que permitió identificar las estrategias de seguridad que hoy en día aplican los programadores y administradores de red, con el fin de mejorarlas mediante la integración de técnicas de seguridad CAS. Cabe señalar, que CAS es el modelo de seguridad que las tecnologías .NET impulsarán en el futuro, por lo cual es necesario que la comunidad de desarrolladores conozcan sobre esta tecnología.

4.2 Relación del método con las necesidades particulares del proyecto

El método cualitativo en los proyectos de investigación, privilegia los elementos de análisis, lo cual permite estudiar lo que las personas perciben como importante. Para el caso de la presente investigación, se optó por el método cualitativo puesto que la intención es descubrir las distintas interpretaciones que los programadores tienen sobre los modelos de seguridad y la integración de estos con CAS. La implantación exitosa del proyecto descrito en este trabajo, requiere de conocer en forma detallada y concreta los procedimientos específicos de seguridad que hoy en día implementan los programadores y administradores de red, por lo cual, el método cualitativo es idóneo para esta necesidad, puesto que los métodos cuantitativos solo permiten medir la frecuencia de los eventos.

4.3 Definición de variables de investigación

Durante la investigación metodológica se definieron mediante variables las principales amenazas en materia de seguridad informática que existen hoy en día. Enseguida se enlistan las variables que representan a las principales amenazas en lo que respecta a seguridad informática:

1 Virus.
2 Modificación de información.
3 Interrupción de servicios.
4 Software malicioso.
5 Acceso no autorizado a los equipos de la red de área local .
6 Acceso no autorizado a los sistemas y bases de datos.
7 Uso de recursos no autorizados por parte del usuario.
8 Intercepción de información cuando los datos se trasladan por una red pública.

A estas primeras variables se les relacionó con otra variable que representó la técnica específica de prevención y solución para la amenaza representada por la primera variable. Enseguida se enlistan las variables que representan técnicas de prevención y solución en el medio de las tecnologías de información:

1 Encriptación y claves.
2 Autorización integrada en sistemas operativos.
3 Zonas de seguridad.
4 Autenticación.
4. Implementación de "Firewalls" y "Proxy servers."

Finalmente, la última variable representa la categoría a la que pertenecen en materia de seguridad informática las dos primeras variables. Enseguida se enlistan las categorías de campo de estudio que se han definido en el contexto de la seguridad de las tecnologías de información.

1. Cifrado.
2. Autorización basada en roles.
3. Seguridad de acceso al código.
4. Autenticación de seguridad integrada en sistemas operativos.
5. Seguridad de arquitectura.

En este orden de ideas, la definición de variables tiene por objetivo relacionar a las principales amenazas existentes en la seguridad informática con su solución técnica y campo de estudio y de esta forma identificar aquellos problemas de seguridad que pertenecen al campo del control de acceso al código. En la tabla 4.1 se muestra la relación entre las variables definidas anteriormente.

Amenaza en el medio informático	**Técnica de solución**	**Campo de estudio**
Virus	Zonas de seguridad	Seguridad de acceso al código
Modificación de información	Encriptación y claves	Cifrado
Interrupción de servicios	Implementación de Firewalls y Proxy servers	Seguridad en arquitectura
Software malicioso	Zonas de seguridad	Seguridad de acceso al código
Acceso no autorizado a los equipos de la red de área local	Autenticación	Autenticación de seguridad integrada en sistemas operativos
Acceso no autorizado a los sistemas y bases de datos	Autorización integrada en sistemas operativos	Autorización basada en roles
Uso de recursos no autorizados por parte del	Autorización integrada en sistemas operativos	Autorización basada en roles

usuario		
Intercepción de información cuando esta se traslada por una red pública	Encriptación y claves	Cifrado

Tabla 4.1. Relación de amenazas informáticas y su campo de estudio

4.4 Descripción del método

El método de investigación se llevó a cabo mediante la aplicación de una entrevista, a personal que tiene puestos relevantes relacionados con las tecnologías de información. El perfil de los entrevistados, así como el de las compañías para las que laboran es amplio. En algunos casos se acudió a compañías especializadas en ofrecer servicios de consultoría en tecnologías de información y en otros casos a compañías de otro giro pero en donde las tecnologías de información tienen un papel preponderante. Esto con el fin de obtener la mayor diversidad posible en los aspectos que tengan que ver con prácticas de seguridad en tecnologías de información.

Las entrevistas tuvieron el propósito de discernir las amenazas comunes y constantes para las tecnologías de información, así como los controles y procedimientos de prevención que se establecen para el diseño de una arquitectura de seguridad. En este orden de ideas, el método permitió conocer la cultura de seguridad informática que existe en el medio de las compañías establecidas en el país.

El método de investigación que se llevó a cabo para el presente trabajo se desglosa en los siguientes pasos:

1. Definición del problema y establecimiento del objetivo.
2. Establecimiento del marco teórico mediante la consulta de bibliografía y entrevistas a especialistas en materia de seguridad informática.
3. Definir los factores del problema a investigar que en este caso son las distintas amenazas a los sistemas de cómputo y establecer el universo de muestras que comprende a profesionales de las tecnologías de información en todos los ámbitos.
5. Crear los instrumentos para la investigación (guía de entrevista)
6. Aplicar la investigación de campo, mediante la entrevista. La investigación se llevó a cabo, en algunos casos mediante una sesión cara a cara con el especialista, en donde el entrevistador hace uso de una guía de entrevista, y en otros casos, se contactó al especialista vía correo electrónico y se le pidió su aportación mediante la lectura y respuesta a las preguntas de la guía de la entrevista.
7. Analizar los datos y diseñar el esquema tecnológico
8. Validar la propuesta con los especialistas y contra casos documentados.

9. Establecer las conclusiones de la investigación.

4.5 Población y muestreo utilizado

En función de las características de la investigación, no se aplicó una técnica de muestreo específica. La población utilizada consistió en 30 profesionales de las tecnologías de información que ocupan cargos del área de informática en las respectivas empresas en que laboran. Las empresas pertenecen a sectores diversos como lo son: consultoría de servicios de información (EDS), Instituto Mexicano del seguro social (IMSS), Cadenas de autoabastecimiento (Grupo Soriana), PEMEX refinación, empresas privadas de consultaría de servicios de tecnologías de información, Tecnológico de Monterrey, Universidad Nacional Autónoma de México, Academia de Ciencias Mexicana, Oracle S.A de C.V, General Electric plastics, Dupont división agroquímicos, Hospital Médica Sur, Hospital Instituto Nacional de Pediatría, HoneyWell S.A. de C.V. y Banobras. El perfil de los puestos de los entrevistados está enfocado a las áreas de tecnologías de información: ingeniero de servicio, gerente de tecnologías de información, CIO (Chief information officer), soporte técnico, programador analista junior, administrador de red y gerente de producto.

4.6 Instrumentos de investigación

Para el desarrollo de la investigación , se utilizaron como instrumentos la guía de entrevista. Tal como se comentó en las secciones anteriores, a veinte de los profesionales que colaboraron en la investigación se les proporcionó la guía de entrevista mediante correo electrónico. En diez casos, se entrevistó al especialista de forma directa. La guía de entrevista fue diseñada mediante preguntas que le permitieron al entrevistado explayarse mas en sus respuestas con el fin de identificar problemas muy específicos con la seguridad de información en las compañías. La guía de entrevista se diseñó para que fuera contestada en un máximo de 30 minutos con el fin de no distraer a los entrevistados de sus actividades diarias. Sin embargo para aquellos especialistas que aceptaron la entrevista directa, se les aplico la guía de entrevista en un tiempo promedio de una hora, en donde la respuesta a las preguntas del cuestionario fueron escritas por el entrevistador en función de que el entrevistado se explayó en los temas y abordo distintos puntos de vista. En general, el proceso de entrevista directa enriqueció la investigación porque permitió el compartir las experiencias prácticas de los entrevistados en temas como ataque de virus, intrusión de hackers, robo de información y estrategias de seguridad, así como le permitió al entrevistador identificar el grado de conocimiento en las compañías sobra las técnicas de seguridad de acceso al código.

El anexo 1 contiene la guía de entrevista utilizada durante el método de investigación.

4.7 Técnica usada para la recopilación de datos

La recopilación de datos tuvo lugar en el momento de recibir la guía de entrevista mediante correo electrónico y en los casos de las sesiones, durante la entrevista directa. La información obtenida se analizó mediante la comparación y el estudio del material bibliográfico previamente obtenido. Los resultados se recopilaron en documentos Word, así como las conclusiones derivadas de los cuestionarios.

Capítulo 5. Análisis de resultados

5.1 Introducción

En el presente capítulo, se estudia a detalle el resultado de las entrevistas aplicadas con el fin de analizar los diversos aspectos de la problemática de seguridad informática que existe en las organizaciones del medio nacional, así como el grado de avance de los especialistas para enfrentar a las amenazas en materia de tecnologías de información. Cabe señalar, que durante el método de investigación, se puso especial énfasis en los problemas relacionados con la seguridad de acceso al código y mediante el análisis de la información obtenida, se obtuvo el grado de conocimiento y aplicación sobre las técnicas de seguridad de acceso al código para la resolución de problemas de ejecución de software dañino.

5.2 Resultados obtenidos

5.2.1 Resultado para la primera pregunta de la guía de entrevista.

La primera pregunta tuvo por finalidad, conocer el giro de la compañía en la que labora el entrevistado. Cabe señalar, que los entrevistados trabajan para compañías relevantes y con una marca y presencia reconocida en el medio nacional. Los resultados de esta pregunta denotan que los profesionales de las tecnologías de información trabajan en empresas que pertenecen a diversos segmentos industriales, de servicios y académicos, tanto en el sector privado, como en el sector público. En la tabla 5.2.1 se señalan los resultados sobre la pregunta uno de la guía de entrevistas: ¿Cuál es el giro de la compañía en la que labora?.

Giro	Sector	Compañía	Número de entrevistados
Proveedor de servicios de tecnologías de información	Privado	EDS de México	2
Proveedor de servicios de salud	Público	Instituto Mexicano del Seguro Social IMSS	1
Comercio	Privado	Soriana	1
Proveedor de servicios energéticos	Público	PEMEX Refinación	2
Proveedor de servicios de tecnologías de información	Privado	Pc Help, S.A. de C.V.	5
Educación	Civil	Instituto Tecnológico de Monterrey. Campus México	2
Educación	Civil	Academia Mexicana de Ciencias	1
Proveedor de servicios de tecnologías de información	Privado	ORACLE de México	1
Proveedor de servicios y productos para el segmento industrial	Privado	General Electric Plastics S.A. de C.V.	2
Proveedor de servicios y productos para el segmento agrícola	Privado	Dupont México	1
Proveedor de servicios de salud	Privado	Grupo Médica Sur	2

Proveedor de servicios de salud	Público	Instituto Nacional de Pediatría	1
Proveedor de servicios y productos para el segmento de las telecomunicaciones	Privado	HoneyWell México	2
Proveedor de servicios financieros	Privado	Banco Nacional de Obras y Servicios Públicos. (Banobras)	1
Educación	Público	Universidad Nacional Autónoma de México	4
Educación	Civil	Instituto Tecnológico de Monterrey. Campus Estado de México	1
Proveedor de servicios para el segmento inmobiliario	Privado	Regus México, S.A de C.V.	1

Tabla 5.2.1 Giro de las compañías

De acuerdo a los resultados de la tabla 5.2.1, los profesionales de las tecnologías de información laboran en una amplia gama de segmentos: educativos, de servicios e industriales. Hoy en día, las tecnologías de información se han convertido en herramientas necesarias para apoyar los procesos productivos y para delinear la estrategia de negocio en las organizaciones. En este orden de ideas, las organizaciones cuentan con profesionales de las tecnologías de información que laboran de forma directa o mediante un esquema de "outSourcing". El tema de las amenazas de seguridad informática es una de las responsabilidades asignadas a los departamentos de TI de las compañías en las organizaciones.

5.2.2 Resultado para la segunda pregunta de la guía de entrevista

La segunda pregunta tiene por objetivo conocer el puesto que ocupan las personas entrevistadas en la organización, lo que ayudará a los propósitos de esta investigación para saber el grado de responsabilidad del entrevistado en los problemas de amenazas de seguridad informática. La tabla 5.2.2 muestra el puesto de los entrevistados para la organización en la que trabajan.

Compañía	Puesto	Responsabilidad
EDS de México	Ingeniero de sistemas	Análisis, diseño y desarrollo de sistemas de cómputo.
	Ingeniero de soporte	Soporte de sistemas informáticos en sitio.
Instituto Mexicano del Seguro Social IMSS	Coordinador de programas	Análisis, diseño y desarrollo de sistemas de cómputo.
Soriana	Programador Tecnología Web	Desarrollo de aplicaciones Web.
PEMEX Refinación	Gerente de Tecnologías de Información	Responsabilidad directiva.
	Especialista en Comercio Electrónico	Análisis y desarrollo de sistemas empresariales.
Pc Help, S.A. de C.V.	Ingeniero de sistemas	Análisis, diseño y desarrollo de sistemas de cómputo.
	Ingeniero de soporte	Soporte de sistemas informáticos

		en sitio.
	Gerente de Tecnologías de Información	Responsabilidad directiva.
	Programador tecnología Web	Desarrollo de aplicaciones Web.
	Ventas	Desarrollo de cartera de clientes.
Instituto Tecnológico de Monterrey. Campus México	Académico de Licenciatura ITIC	Labor docente.
	Académico en LATI	Labor docente.
Academia Mexicana de Ciencias	Programador Tecnología Web	Desarrollo de aplicaciones Web.
ORACLE de México	Gerente de Soluciones Corporativas	Análisis y desarrollo de sistemas empresariales.
General Electric S.A. de C.V.	Programador Tecnología Web	Desarrollo de aplicaciones Web.
	Ingeniero de soporte	Soporte de sistemas informáticos en sitio.
Dupont México	Gerente de Tecnologías de Información	Responsabilidad directiva.
Grupo Médica Sur	Ingeniero de soporte	Soporte de sistemas informáticos en sitio.
	Programador de tecnología Web	Desarrollo de aplicaciones Web.
Instituto Nacional de Pediatría	Ingeniero de soporte	Soporte de sistemas informáticos en sitio.
HoneyWell México	Ingeniero de soporte	Soporte de sistemas informáticos en sitio.
	Ingeniero de sistemas	Análisis, diseño y desarrollo de sistemas de cómputo.
Banco Nacional de Obras y Servicios Públicos. (Banobras)	Ingeniero de sistemas	Análisis, diseño y desarrollo de sistemas de cómputo.
Universidad Nacional Autónoma de México	Ingeniero de soporte	Soporte de sistemas informáticos en sitio.
	Analista de sistemas	Análisis, diseño y desarrollo de sistemas de cómputo.
	Técnico de infraestructura de redes	Instalación y soporte técnico de redes.
	Académico en Licenciatura de Ingeniería en computación	Labor docente.
Instituto Tecnológico de Monterrey. Campus Estado de México.	Académico de Licenciatura LSCA	Labor docente
Regus México, S.A de C.V.	Ingeniero de soporte	Soporte de sistemas informáticos en sitio.

Tabla 5.2.2 Puesto de los entrevistados

La sección 5.2.2 denota la amplia gama de posiciones y de especialidades en torno a las actividades de tecnologías de información en las compañías. Cabe señalar que de acuerdo al análisis de resultados para esta pregunta, todavía no existe en muchas compañías el puesto específico para aspectos de seguridad informática. De acuerdo a los gerentes entrevistados, la seguridad informática en la práctica, corre a cargo de los ingenieros de soporte técnico y de infraestructura de redes, ya que ellos tienen por responsabilidad la de instalar y actualizar los

paquetes antivirus, así como establecer los servidores Proxy y Firewall que protegerán a las redes de datos de las compañías. En este orden de ideas, los entrevistados de la compañía Pc Help, mencionaron que en los últimos años se han especializado en ofrecer servicios de seguridad basados en la distribución del paquete antivirus "Trend Micro", por lo que en esta compañía si han considerado la creación de un puesto exclusivo para seguridad informática. Sin embargo, de acuerdo a la entrevista realizada al personal de PEMEX, la formalización de un puesto exclusivo para las amenazas en materia de seguridad informática es inminente, puesto que la seguridad hoy en día es una de las principales prioridades en los departamentos de TI. Cabe señalar que de acuerdo a las opiniones recabadas, hoy en día ya existen certificaciones de seguridad dadas por empresas reconocidas en el medio, tal como Microsoft, así como diplomados y especialidades en seguridad informática impartidos por universidades de prestigio y que son aspectos que el personal de recursos humanos considerará para la organización de los departamentos de tecnologías de información. De acuerdo, a la información recabada para la pregunta dos de la guía de entrevista, la seguridad informática será un aspecto cultural en las organizaciones, como lo es la seguridad industrial en las fábricas y existirá un área responsable para este problema.

5.2.3 Resultado para la tercera pregunta de la guía de entrevista

La tercera pregunta de la guía de entrevista analiza el aspecto de seguridad informática que con mayor frecuencia experimentan los entrevistados. Las categorías de amenaza en seguridad informática que se definieron para la pregunta tres son:

1. Virus
2. Modificación de información
3. Interrupción de servicios
4. Software malicioso
5. Acceso no autorizado a los equipos de la red de área local.
6. Acceso no autorizado a los sistemas y bases de datos.
7. Uso de recursos no autorizados por parte del usuario.
8. Intercepción de información cuando esta se traslada por una red pública.

En la figura 5.2.3 se exponen el resultado de las respuestas, en donde el eje X denota el tipo de amenaza y el eje Y representa el total de las selecciones para cada amenaza informática por parte de los entrevistados:

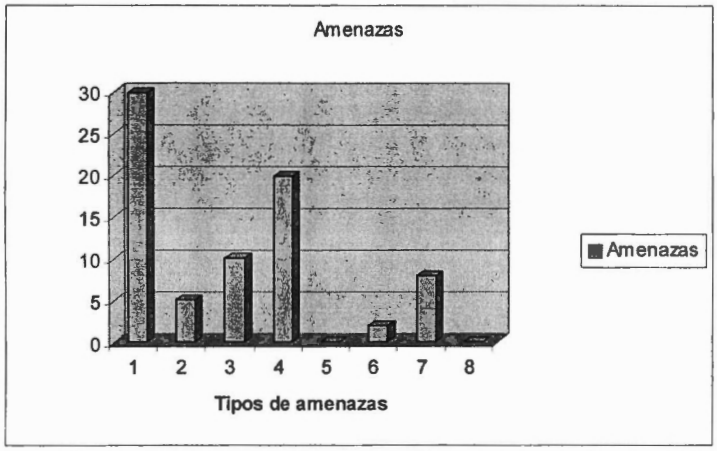

Figura 5.2.3. Frecuencia de amenazas informáticas

La pregunta 3 de la guía de entrevista denota las principales amenazas informáticas a las que se enfrentan las organizaciones en el medio nacional. Cabe señalar que el problema de los virus informáticos es una amenaza que todos los entrevistados dijeron experimentar en sus compañías en algún momento. Por otra parte, la segunda causa de amenaza es la de la ejecución de software no autorizado o malicioso en los equipos de cómputo de los usuarios. Dicho software generalmente se filtra a las redes locales de las compañías mediante mensajes de correo electrónico a los usuarios, en donde se les conmina a abrir un enlace, instalar un archivo o abrir un documento, de acuerdo a la información recabada en las entrevistas directas que se sostuvieron en la UNAM. Tal como se ha mencionado en los capítulos anteriores de este trabajo de investigación, el software de código malicioso, en donde se incluyen a los virus informáticos, tiene como método de prevención a la técnica de acceso de seguridad al código. El hecho de que los virus informáticos y el software malicioso sean los principales problemas en materia de seguridad informática, de acuerdo a los entrevistados, resalta la necesidad de fortalecer y difundir las técnicas de seguridad de acceso al código. Cabe señalar, que para esta pregunta, los entrevistados se explayaron en la descripción de la estrategia antivirus que la mayor parte de las compañías implementan y que consiste básicamente en un servidor de monitoreo en la red de área local para administrar los programas antivirus instalados en las computadoras cliente. La actualización de los antivirus, se realiza de manera automática, pero también desde el administrador de antivirus en el servidor, es posible realizar una instalación manual. A su vez, la utilería de administrador antivirus contiene filtros para detectar la presencia de un programa nocivo en los documentos adjuntos a los correos electrónicos de los usuarios. Sin embargo, el programador de tecnología Web del grupo Médica Sur señaló que las estrategias antivirus no son infalibles y luego entonces es necesario y deseable el complemento de dichas estrategias con técnicas de seguridad de acceso al código. La mayor parte de los entrevistados que desarrollan aplicaciones Web acogió con interés el tema de la seguridad de acceso

al código, puesto que la funcionalidad de sus aplicaciones, dependerá de los permisos de ejecución de código que las organizaciones permitan en sus respectivas infraestructuras.

5.2.4 Resultado para la cuarta pregunta de la guía de entrevista

La pregunta cuatro de la guía de entrevista consistió en preguntar a los entrevistados las principales afectaciones que la compañía en donde laboran ha experimentado debido a ataques informáticos. Los entrevistados señalaron que la principal consecuencia de un ataque informático es en tiempo de productividad y horas de trabajo útil. En este orden de ideas, un ataque informático en una organización supone la cancelación temporal de servicios como: correo electrónico, aplicaciones de trabajo colaborativo, bases de datos y aplicaciones financieras. El caso de los sistemas en los hospitales, de acuerdo a los entrevistados del IMSS y del Instituto Nacional de Pediatría, es particularmente sensible puesto que algunas áreas de terapia intensiva, son monitoreadas por aparatos que tienen una interfase y aplicación instalada en los equipos de cómputo.

De la sección 5.2.4 se infiere que el principal daño en materia de ataques informáticos para una organización es en términos de productividad. Por una parte, los usuarios ven limitado su campo de acción puesto que en un ataque informático, la red o su equipo se da de baja temporalmente para resolver el problema. Sin embargo, el personal de los departamentos de tecnología de información, también sufre una merma en su productividad puesto que todos los recursos del área, durante una crisis, están enfocados a resolver el problema de seguridad. De acuerdo al gerente de tecnologías de información Dupont, cuando un virus paraliza a la red, todo el personal del departamento de TI, desde el gerente hasta los ingenieros de servicios, enfocan sus esfuerzos en resolver la crisis, con lo cual, otros proyectos y asignaciones del personal de TI se ven retrasados. De acuerdo al personal de Pc Help, las compañías sufren también como consecuencia de un ataque, una merma económica, puesto que se requieren de mas horas de ingenieros de servicio en sitio, para resolver la crisis. PC Help, especifica en sus contratos, mediante una cláusula, el número de horas por mes destinadas a resolver un problema de virus, sin embargo, en ocasiones no es posible resolver el problema en las horas especificadas en el contrato, por lo que las compañías incurren en un gasto económico. Los entrevistados no han experimentado daños por amenazas de otra índole como son, robo de información encriptada o acceso no autorizado a la red de área local y a las bases de datos respectivas. En este sentido, la respuesta a la pregunta cuatro de la guía de entrevista, hace lógica con el análisis de resultados de la respuesta tres, en el sentido de que los programas de software malicioso, como los virus, son las principales amenazas en materia de seguridad informática para las compañías y es necesario implementar permisos para la ejecución de código, es decir técnicas CAS.

5.2.5 Resultado para la quinta pregunta de la guía de entrevista

La quinta pregunta de la guía de entrevista, tiene por objetivo conocer sobre las estrategias integrales de seguridad informática en las compañías del medio nacional. La mayor parte de los entrevistados reconoce a la información como uno de los activos mas valiosos de la compañía para la cual labora. En algunos casos, como lo es el de la compañía Soriana, el programador Web entrevistado señaló que las tecnologías de información son un factor estratégico, mediante el cual, la compañía ha decidido distinguirse de sus competidores. Soriana, ha implementado un modelo de negocio basado en hacer mas productivas sus relaciones con los proveedores y con los clientes mediante aplicaciones Web. Es en este contexto, que la mayor parte de los entrevistados señalaron la importancia de contar con una estrategia integral de seguridad. Sin embargo solo en el caso de PEMEX y Grupo Soriana, se ha considerado con formalidad, una serie de políticas y procedimientos que forman una estrategia integral de seguridad informática. Las características de la estrategia integral de seguridad para el Grupo Soriana son:

- Programas antivirus actualizados
- Aplicación automática de parches en las aplicaciones para remover vulnerabilidades.
- Autenticación estricta de usuarios mediante nombre de usuario y contraseña.
- Seguridad en enlaces inalámbricos mediante el uso de redes virtuales VPN.

El análisis de resultados de esta sección establece que la seguridad informática integrada ha surgido para proteger los activos de información de las compañías, así como sus infraestructuras de tecnologías de información. Sin embargo, el concepto de seguridad informática integrada todavía no es una práctica común en los planes de los departamentos de TI. En este orden de ideas, la seguridad integrada combina múltiples tecnologías de seguridad junto con el diseño y cumplimiento de políticas y procedimientos. El tema de este trabajo de investigación, seguridad de acceso al código, de acuerdo a los entrevistados, es solo una parte de la política de seguridad integral que debería de existir en la mayor parte de las organizaciones. De acuerdo a la información recabada mediante la pregunta cinco, los entrevistados señalaron los siguientes puntos y componentes de las tecnologías de información como parte de una política de seguridad informática:

- Firewalls: Control de tráficos de las redes locales mediante la selección de información que entra y sale para garantizar que no ocurran accesos no autorizados.
- Detección de intrusos: Detección de accesos no autorizados y emisión de alertas y reportes para darle seguimiento al acceso ilegal.
- Filtrado de contenidos en aplicaciones de correo electrónico: Detección y eliminación de archivos con código de software potencialmente maligno.

- Redes Privadas Virtuales: Protección de las conexiones fuera del perímetro de la red de área local, lo que le permite a los usuarios de las organizaciones comunicarse desde las redes de datos públicas, de manera segura.
- Manejo de la vulnerabilidad: Introspección de fisuras de las aplicaciones y sistemas operativos.
- Seguridad de acceso al código: Establecimiento de permisos en las librerías del sistema operativo y de las máquinas virtuales de ejecución de aplicaciones .NET con el fin de controlar las acciones de las aplicaciones desarrolladas mediante tecnología .NET.

5.2.6 Resultado para la sexta pregunta de la guía de entrevista

La pregunta seis de la guía de entrevista tuvo por finalidad conocer sobre si las empresas para las que trabajan los entrevistados cuentan con alguna aplicación en específico de seguridad informática. Los resultados de la pregunta seis se presentan en la tabla 5.2.6:

Compañía	Herramienta	Aplicación
Eds de México	Planeación y administración de la seguridad informática. Diagnóstico de vulnerabilidades y riesgos. Diagnóstico de seguridad	Net Protect Enterprise Mcafee ThreatScan NetIntelligence
Instituto Mexicano del Seguro Social IMSS	Planeación y administración de la seguridad informática.	Norton Antvirus Corporate Edition
Soriana	Planeación y administración de la seguridad informática Diagnóstico de vulnerabilidades y riesgos	Mcafee VirusScan DeLuxe Mcafee ThreatScan
PEMEX Refinación	Planeación y administración de la seguridad informática. Diagnóstico de vulnerabilidades y riesgos. Diagnóstico de seguridad	Mcafee VirusScan Deluxe Mcafee ThreatScan PestPatrol
Pc Help, S.A. de C.V.	Planeación y administración de la seguridad informática. Diagnóstico de vulnerabilidades y riesgos. Diagnóstico de seguridad	Norton Antivirus Corporate Edition PatchLink update ETrust Intrusión Detección
Instituto Tecnológico de Monterrey. Campus México	El entrevistado no contó con información al respecto	
Academia Mexicana de Ciencias	El entrevistado no contó con información al respecto	
ORACLE de México	Planeación y administración de la seguridad informática. Diagnóstico de vulnerabilidades y riesgos. Diagnóstico de seguridad	Mcafee VirusScan Deluxe Mcafee ThreatScan Mcafee VirusScan Deluxe
General Electric	Planeación y administración de la seguridad	

S.A. de C.V.	informática. Diagnóstico de vulnerabilidades y riesgos.	 Microsoft Windows Patch update
Dupont México	Planeación y administración de la seguridad informática. Diagnóstico de seguridad	Norton Antivirus Corporate Edition Norton Antivirus Corporate Edition
Grupo Médica Sur	Diagnóstico de vulnerabilidades y riesgos.	Microsoft Windows Patch update
Instituto Nacional de Pediatría	Diagnóstico de vulnerabilidades y riesgos.	Microsoft Windows Patch update
HoneyWell México	Planeación y administración de la seguridad informática. Diagnóstico de vulnerabilidades y riesgos. Diagnóstico de seguridad	Norton Antivirus Corporate Edition Microsoft Windows Patch update Norton Antivirus Corporate Edition
Banco Nacional de Obras y Servicios Públicos. (Banobras)	Planeación y administración de la seguridad informática. Diagnóstico de vulnerabilidades y riesgos. Diagnóstico de seguridad	Mcafee VirusScan Deluxe Microsoft Windows Patch update Mcafee VirusScan Deluxe
Universidad Nacional Autónoma de México	El entrevistado no contó con información al respecto	
Instituto Tecnológico de Monterrey. Campus Estado de México.	El entrevistado no contó con información al respecto	
Regus México, S.A de C.V.	Diagnóstico de vulnerabilidades y riesgos.	Microsoft Windows Patch update

Tabla 5.2.6. Aplicación de herramientas de seguridad informática

En esta sección, se establece que la mayor parte de las compañías tienen conciencia sobre la necesidad de contar con herramientas para controlar y prevenir las amenazas de índole informática. Cabe señalar que de acuerdo a la tabla 5.2.6, la herramienta de mayor uso en las organizaciones es la de actualización automática de vulnerabilidades en las aplicaciones y / o sistema operativo. Esto en función de que la utilería de aplicación de parches para vulnerabilidades en el software viene integrada en la mayor parte de los sistemas operativos, como es el caso del sistema Windows XP. Otra conclusión derivada de la pregunta seis es que solo los corporativos o empresas especializadas en servicios de tecnologías de información cuentan con una utilería de administración de planeación y administración de la seguridad informática debido a la mayor conciencia entre el personal de TI sobre los daños de las amenazas informáticas. Cabe señalar, que los sistemas operativos actuales contienen una utilería de administración de seguridad de acceso al código, sin embargo ninguno de los entrevistados mención su uso, lo que denota todavía una falta de conocimiento hacia el tipo de problemas que la seguridad CAS puede resolver.

5.2.7 Resultado para la séptima pregunta de la guía de entrevista

La aplicación de la pregunta número siete de la guía de entrevista tiene por objetivo recabar información sobre los recursos y presupuestos que las organizaciones asignan para el control y prevención de amenazas de índole informática. Cabe señalar que en el plan de negocios de las compañías que participaron en esta investigación, se realiza un presupuesto general para las tecnologías de información, en donde se consideran:

- Equipo de hardware.
- Recursos humanos para el área de tecnología de información.
- Licencias de software.
- Contratos de "outsourcing" para mantenimiento.

De acuerdo a la información recabada en la sección 5.2.7, las compañías no tienen en sus presupuestos un rubro específico para los problemas de seguridad informática. Para la mayor parte de los gerentes de tecnologías de información entrevistados, las aplicaciones de software antivirus se cargan al centro de costos que representa el rubro de licencias de software. Respecto a los recursos empleados, los ingenieros de servicio en sitio, generalmente, además de las tareas de mantenimiento o soporte a los sistemas o infraestructuras de redes, proveen los servicios de actualización de antivirus. De acuerdo al personal entrevistado de la empresa PcHelp y EDS de México, cuando se vende una solución integral de infraestructura a una compañía, las licencias de los antivirus en las computadoras clientes, así como el costo de las aplicaciones de firewall y proxy servers forman parte de la cotización en el rubro de aplicaciones o licencias, sin existir una categoría específica para problemas de seguridad.

5.2.8 Resultado para la octava pregunta de la guía de entrevista

La octava pregunta de la guía de entrevista tiene por objetivo, conocer si las compañías cuentan entre su personal, con algún especialista en técnicas de seguridad informática. Enseguida se enlistan las técnicas de seguridad tomadas en cuenta para esta pregunta:

1. Cifrado
2. Autorización basada en roles
3. Seguridad de acceso al código.
4. Autenticación de seguridad integrada en sistemas operativos
5. Seguridad de arquitectura.

Los resultados se muestran en la figura 5.2.8 en donde el eje de las X representa el tipo de técnicas de seguridad y el eje de las Y representa el número de empresas que si cuenta con alguna de las técnicas mencionadas anteriormente:

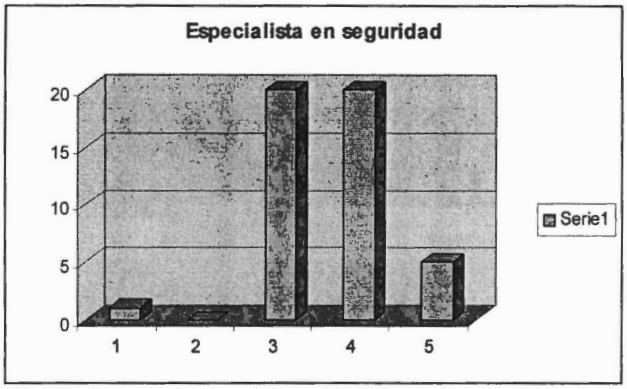

Figura 5.2.8. Especialista en seguridad informática

5.2.9 Resultado para la novena pregunta de la guía de entrevista

Respecto a la pregunta nueve de la guía de entrevista, la mayor parte de los entrevistados desconoce las técnicas de seguridad de acceso al código. Del total de los entrevistados, solamente los programadores de aplicaciones Web están familiarizados con el concepto de seguridad de acceso al código. Cabe señalar que solamente el programador de aplicaciones Web de la compañía PcHelp, ha desarrollado aplicaciones tomando en cuenta el concepto de zona que se explico en el marco teórico del presente trabajo de investigación.

La pregunta nueve de la guía de entrevista es la que mayor valor aporta a este trabajo de investigación. De acuerdo a los resultados de la sección 5.2.9, la mayor parte de los entrevistados desconoce las técnicas de seguridad basadas en dotar de permisos de ejecución a las aplicaciones que residan en los equipos de cómputo. Cabe señalar, que los entrevistados están relacionados con las técnicas de seguridad al usuario, explicadas en el marco teórico, en donde, al usuario se le asignan permisos para delimitar sus acciones y el acceso a recursos. Sin embargo, la seguridad de acceso al código representa una técnica en donde el concepto de asignar permisos se traslada a las aplicaciones de software. Dicho concepto fue entendido fácilmente por el personal entrevistado y establecieron un paralelismo entre la seguridad basada en el usuario y la seguridad de acceso al código. Tal como se señaló en la sección 5.2.3, la mayor parte de las amenazas informáticas en las compañías están en función de la ejecución de software de código malicioso o dañino, lo que incluye desde virus a programas o documentos que se adjuntan en los correos electrónicos de los usuarios. En este contexto, los entrevistados demostraron un gran interés por el concepto de clasificar las áreas de ejecución de software mediante zonas, en donde cada zona tiene un perímetro de seguridad basado en candados y restricciones por parte del motor de ejecución de las aplicaciones .net .

5.2.10 Resultado para la décima pregunta de la guía de entrevista

La pregunta diez de la guía de entrevista tiene por objetivo conocer si las gerencias de las compañías están conscientes e involucradas en el diseño de las estrategias de seguridad informática. Los resultados obtenidos demuestran que la mayor parte de los directivos saben del problema de seguridad informática y sus consecuencias, sin embargo delega el diseño de la estrategia en los responsables del departamento de tecnologías de información. Los directivos de empresas especializadas en servicios de tecnologías de información si asumen una participación activa y directa en los procedimientos y políticas para definir la estrategia de seguridad informática.

La sección 5.2.10 establece que los directivos de las organizaciones conocen sobre las consecuencias de los ataques informáticos, sin embargo, todavía no participan directamente en fomentar una cultura de seguridad informática para toda la empresa. De acuerdo al gerente de tecnologías de información de PcHelp, la seguridad informática es un concepto que en un futuro cercano se integrará a la cultura de seguridad en las organizaciones. Así como actualmente, las compañías han desarrollado sistemas de seguridad física y de seguridad industrial, la protección a las amenazas informáticas deberá ser parte de las políticas y procedimientos comunes a todos los empleados de una compañía.

5.3 Análisis de resultados en general

De acuerdo a la investigación llevado a cabo, se establece que para el grupo de entrevistados los mecanismos de seguridad más comunes conceden derechos a los usuarios basándose en las credenciales de inicio de sesión (normalmente, una contraseña) y limitan los recursos (a menudo, directorios y archivos) a los que puede obtener acceso el usuario. Este tipo de mecanismo de seguridad se conoce como seguridad basada en el usuario. Sin embargo, tal como se ha señalado en el capítulo de introducción y corroborado en las sesiones de entrevista, actualmente los usuarios obtienen código de muchos orígenes, que pueden no ser fiables, el código puede contener errores o puntos vulnerables que hacen que pueda ser atacado por código malicioso y, en ocasiones, el código realiza acciones que el usuario desconoce. En este sentido, la seguridad de usuario no es suficiente ya que los sistemas informáticos pueden resultar dañados y se puede filtrar información confidencial cuando los usuarios prudentes y de confianza ejecutan software malicioso o lleno de errores. El método de investigación ha puesto de manifiesto la necesidad de complementar las estrategias integrales de seguridad en las compañías mediante un mecanismo que proporcione al código varios grados de confianza, dependiendo de su procedencia y de otros aspectos de la identidad del código. En este orden de ideas, la seguridad de acceso a código impondrá distintos niveles de confianza en el código, lo que permitirá un mayor control sobre las acciones y recursos que las aplicaciones puedan ejecutar.

Capítulo 6. Conclusiones

6.1 Introducción

En el presente capítulo se establecen conclusiones en función de los resultados obtenidos mediante el método de investigación llevado a cabo. Las conclusiones se han planteado de acuerdo a los objetivos e hipótesis sobre los modelos de seguridad informática, definidos en la sección de "Introducción" del presente trabajo de investigación.

6.2 Respuesta global al problema

A partir de la información recabada en la sección de análisis de resultados, se pone de manifiesto que las organizaciones están expuestas en forma constante a una gran diversidad de amenazas en materia de seguridad informática. De acuerdo a los resultados de las entrevistas llevadas a cabo en la metodología de investigación, el software de código malicioso, en donde se incluyen a los virus computacionales, es el principal problema en materia de seguridad informática que existe en las organizaciones del país. El resultado obtenido, en el sentido de que los problemas causados por código malicioso ocupan el primer lugar en materia de amenazas informáticas en las organizaciones, hace sentido con las aseveraciones de la sección de "justificación" del presente trabajo, en donde se señala que el daño causado por programas de tipo virus tiene un impacto económico anual en las empresas de los Estados Unidos por una cantidad de 320 millones de dólares [Sánchez, 2005]. Cabe señalar que en función del trabajo de investigación llevado a cabo, se sabe que la mayor parte de los responsables de seguridad informática en las organizaciones abordan el problema del código malicioso desde la perspectiva de una estrategia antivirus. En este sentido, la respuesta global actual al problema de la ejecución de código dañino, se basa en la implementación de una estrategia antivirus que consiste básicamente en instalar un servidor de monitoreo en la red de área local para administrar los programas antivirus instalados en las computadoras clientes. La mayor dificultad en implementar una solución eficiente de detección de virus en una corporación, radica en las dificultades en transferir las nuevas versiones del antivirus a todos los equipos de la organización y luego entonces, el éxito de las soluciones antivirus estriba en la selección de una herramienta de distribución que permita a los administradores de red configurar y actualizar las nuevas versiones.

6.3 Respuestas a preguntas secundarias planteadas en la introducción

En la sección de "introducción" del presente trabajo de investigación, se menciona la importancia actual de las tecnologías de información en los procesos de producción y modelos de negocio en las organizaciones. Aunado al tema de seguridad de acceso al código que es el tema principal del presente trabajo, el método de investigación llevado a cabo da respuesta a aspectos secundarios planteados en la sección de "introducción" y que son inherentes a la seguridad informática, como son: el valor de la información como activo en las organizaciones

y el papel de la seguridad informática como especialidad de las tecnologías de información.

Derivado del análisis general de las entrevistas llevadas a cabo a profesionistas de las tecnologías de información, se concluye que el activo mas importante hoy en día en las organizaciones es la información. Actualmente, los procesos informáticos son un factor estratégico para la consecución de los objetivos de las compañías. Cabe señalar que hace algunos años, las tecnologías de información, se ubicaban como excelentes herramientas para mejorar la productividad, al automatizar y hacer mas confiables aquellos procesos que requieren de muchos cálculos, tales como: cálculos de nóminas, administración de materiales y aspectos contables. Sin embargo, el comentario general de los entrevistados, es en el sentido de que los modelos de negocios actuales en muchas organizaciones están íntimamente ligados a los aspectos de los negocios electrónicos, lo que convierte a los diversos elementos de las tecnologías de información en factores estratégicos que marcarán o diferenciarán a las compañías de su competencia. Ejemplos que se mencionaron fueron los de las compañías Wall Mart, y Dell en donde en el primer caso, la automatización de los diversos procesos que involucran a la cadena de valor, tales como la distribución, logística y almacenamiento le ha permitido a Wall Mart posicionarse como una empresa líder en su ramo, y por otra parte, la compañía Dell basa su modelo de negocio en la distribución de su producto a través de la red Internet y en la automatización mediante tecnologías de información de los procesos relacionados con las órdenes de compras.

De esta forma, se concluye, que la respuesta general al problema de la seguridad informática, es en el sentido de que las tecnologías de información son elementos críticos para la supervivencia y el éxito de las organizaciones, así como para cambiar las prácticas de negocio actuales, crear nuevas oportunidades y reducir costos. Luego entonces, la seguridad informática es un área que adquiere mayor importancia día con día puesto que protege al principal activo de las organizaciones, que es la información y las tecnologías que la soportan. El punto de vista de los entrevistados, coincide con las aseveraciones expuestas en la sección de "definición del problema" del presente trabajo, en donde se señala que la seguridad en cómputo es un área estratégica y de importancia vital para el desarrollo de los procesos de tecnologías de información que tienen lugar en las sociedades y en las organizaciones.

6.4 Logro de objetivos planteados al inicio

El análisis de resultados permite corroborar los objetivos planteados en la sección correspondiente en el capítulo de introducción del presente trabajo, en el sentido de que los profesionales de las tecnologías de información, no incorporan de manera sistemática las funciones del modelo de seguridad de acceso al código para proteger los recursos críticos de las organizaciones, y por lo tanto es necesario proporcionar un documento a los desarrolladores de aplicaciones y administradores

de sistema con el fin de que implementen mecanismos que confinen la ejecución de código a un entorno seguro. Tal como se señala en la sección de "respuesta al problema global" en el presente capítulo, los problemas de software malicioso son resueltos hoy en día mediante estrategias antivirus, sin complementar dichas estrategias con funciones de seguridad de acceso a código. En este sentido, se concluye que tal como se planteó en los objetivos iniciales, es necesario desarrollar una metodología que sirva de guía para los responsables de desarrollar aplicaciones y administradores de sistemas, en la cual se explique la incorporación de permisos específicos al código de las aplicaciones con el fin de controlar las operaciones y accesos a los recursos críticos del equipo de cómputo, por parte de dichas aplicaciones.

De acuerdo a la sección de "antecedentes" del capítulo de introducción en el presente trabajo, los modelos de seguridad informática clásicos están basados en la identidad del usuario que intenta ejecutar el código. Sin embargo, el modelo de seguridad de acceso al código está basado en la identidad del propio código que se ejecutará. En este orden de ideas, uno de los objetivos planteados en el capítulo de introducción señala la necesidad de fomentar la cultura de la seguridad de acceso al código mediante la definición de guías y documentos que apoyen a los programadores de aplicaciones .NET y administradores de sistema.

6.5 Los resultados de las hipótesis

La hipótesis definida en el capítulo de introducción del presente trabajo de investigación, establece al modelo de seguridad de acceso al código como un complemento a las estrategias de seguridad informática actuales, con el fin de proporcionar un entorno de mayor seguridad para aquellas organizaciones que están expuestas a la ejecución de código dañino. A lo largo de las entrevistas aplicadas a profesionistas de las tecnologías de información y que son responsables de la seguridad informática en sus compañías respectivas, quedo de manifiesto que la seguridad tradicional de los sistemas operativos es insuficiente ya que solo toma en cuenta la identidad del usuario del código. El análisis de resultados, en específico de las preguntas de la entrevista número cinco y nueve, reafirma la hipótesis del presente trabajo de investigación en el sentido de que es necesario implementar mecanismos de seguridad para limitar las acciones del código mediante permisos de ejecución. En este orden de ideas, los profesionistas entrevistados subrayan la necesidad de incorporar a sus estrategias integrales de seguridad informáticas, los siguientes conceptos inherentes a la seguridad de acceso al código:

- Clasificación y ubicación de diferentes códigos y distintos niveles de confianza, no sólo de acuerdo con los derechos del usuario que ejecuta el código sino también tomando en cuenta quién autorizó el código y las políticas del sistema.

- Todas las aplicaciones que tengan como objetivo ejecutarse en tiempo real deberán interactuar con el sistema de seguridad que se incorpora en la plataforma de ejecución .NET .
- Cuando se ejecute una aplicación, se evaluará automáticamente y el sistema de seguridad CAS proporcionará un conjunto de permisos. Dependiendo de los permisos que reciba la aplicación, se ejecutará adecuadamente o generará una excepción de seguridad. Las configuraciones locales de seguridad en una máquina en particular deciden en última instancia los permisos que recibe el código.

6.6 Contraste entre fundamentos y resultados obtenidos

La sección de fundamentos del presente trabajo de investigación describe una metodología específica para el desarrollador de aplicaciones .NET y administrador de sistema que le permite utilizar mecanismos de seguridad de acceso al código para fortalecer las estrategias integrales de seguridad informática. De manera, que la sección de fundamentos está orientada a técnicas basadas en las bibliotecas de clase de la plataforma .NET para limitar las acciones de las aplicaciones .NET y el acceso a recursos determinados, dichas técnicas se describen en el capítulo de análisis de fundamentos. Sin embargo, a raíz de los resultados obtenidos mediante el método de investigación, quedo de manifiesto que la seguridad CAS es parte de una estrategia integral de seguridad proporcionada por la plataforma .NET . La mayor parte de los profesionistas entrevistados implementan otras técnicas de seguridad propias de la plataforma .NET como son la autenticación y la autorización y que junto con la seguridad de acceso a código, componen el modelo integral de seguridad propuesto por la plataforma .NET. Este resultado contrasta con las aseveraciones del capítulo de "análisis de fundamentos" en donde se señala que la seguridad de acceso al código es la base del modelo de seguridad en la plataforma .NET .

En función de lo expresado por los profesionistas de las tecnologías de información entrevistados, se define a la autenticación como el proceso de solicitar a los usuarios del sistema una cuenta y contraseña válida. Actualmente, se utilizan una gran variedad de mecanismos de autenticación, tales como: autenticación básica, implícita, Passport y de sistema operativo [McDonald, 2004]. Por otra parte, la autorización es el proceso de determinar si se permite a un usuario realizar una acción solicitada. La autorización tiene lugar después de la autenticación y utiliza información relativa a la identidad y rol del usuario para determinar a que recursos puede tener acceso dicho usuario.

Se concluye que la seguridad en la plataforma .NET proporciona el concepto de ubicar diferentes códigos y distintos niveles de confianza, no sólo de acuerdo con los derechos del usuario que ejecuta el código sino también tomando en cuenta quien autorizó el código y las políticas del sistema. En este orden de ideas, la plataforma .NET proporciona tres mecanismos fundamentales para proteger los recursos y el código de usuarios no autorizados:

- Seguridad de la aplicación Web mediante autenticación del usuario.
- Seguridad de acceso a código.
- Seguridad basada en roles.

6.7 Limitaciones o condiciones específicas

El modelo de seguridad propuesto en el presente trabajo de investigación está basado en las características de seguridad de acceso al código en la plataforma .NET desarrollada por la empresa Microsoft. De acuerdo a las opiniones expresadas durante la etapa de investigación queda de manifiesto que las herramientas de desarrollo de aplicaciones que se utilizan hoy en día en las empresas son de diversa índole. En este orden de ideas, el presente trabajo tiene la condición específica de proporcionar una metodología para programadores y administradores de sistemas bajo entornos .NET, en donde el código tiene las siguientes características:

- Código con seguridad de tipos: Para que el código pueda beneficiarse de la seguridad de acceso a código, es preciso utilizar un compilador que genere código con seguridad de tipos comprobable.
- Sintaxis imperativa y declarativa: La interacción con el sistema de seguridad del motor de tiempo de ejecución se realiza mediante llamadas de seguridad imperativas y declarativas. Las llamadas declarativas se realizan mediante atributos, mientras que las llamadas imperativas se realizan con nuevas instancias de clases del código.
- Permisos para el código: Las solicitudes se aplican al ámbito del ensamblado, en donde el código informa al motor de tiempo de ejecución sobre los permisos que requiere para ejecutarse o específicamente no desea. El motor de tiempo de ejecución evalúa las solicitudes de seguridad cuando el código se carga en memoria.
- Bibliotecas de clase seguras: Las bibliotecas de clases utilizan la seguridad de acceso a código para especificar los permisos que requieren para que se obtenga acceso a ellos. Se debe de conocer los permisos necesarios para obtener acceso a cualquier biblioteca que utilice el código y realizar las solicitudes correspondientes en el código.

6.8 Otras conclusiones

En función de los resultados del método de investigación se concluye que la seguridad de la información en las empresas consiste en proteger aquellos activos que poseen valor para las empresas. De acuerdo, a las opiniones recabadas durante la metodología de investigación, son tres elementos los que se consideran como activos, desde el punto de vista de la seguridad informática:

- La información: son elementos que contienen información registrada, en medio electrónico o físico.

- Los infraestructura de la información. Consiste en las aplicaciones de software que se utilizan para la lectura, tránsito y almacenamiento de la información, tales como: aplicaciones comerciales, sistemas operativos, y navegadores de Internet. Así mismo, la infraestructura de la información también se compone de los dispositivos en los que se almacene, procese o transmita la información de la empresa: servidores, enrutadores y equipos portátiles.
- Las personas que las utilizan. Son los individuos que utilizan la estructura tecnológica y de comunicación de la empresa y que manejan a la información.

A raíz del análisis de resultados se concluye que el auge de Internet ha llevado a muchas organizaciones nacionales a utilizar esta tecnología dentro de sus procesos y actividades cotidianas. De hecho, algunas empresas, han establecido procesos en donde Internet es el elemento primordial de su operación. Sin embargo, de acuerdo a los profesionistas de TI entrevistados, al mismo tiempo que se incrementa el uso de redes de datos públicas, los incidentes de seguridad informática se han incrementado de manera dramática. En este orden de ideas, se concluye que cuanto más trascendentes son las tecnologías de información para los procesos de las organizaciones, más importante es la necesidad de diseñar estrategias de seguridad informática que proporcionen altos niveles de disponibilidad y confianza en dichos procesos.

6.9 Recomendaciones

Durante la realización del presente trabajo se encontró que un número significativo de organizaciones de diferentes sectores de la vida pública nacional, están expuestas a incidentes de seguridad en función de vulnerabilidades informáticas existentes en sus sistemas operativos. Un estudio de la empresa de seguridad SekurelT, Consultores en Seguridad Informática, revela que de un total de 242 entidades del sector nacional público, más del 50% de estas contienen vulnerabilidades de alto riesgo en sus aplicaciones de software [Ardita,2000]. A raíz de estos resultados, se sugiere ahondar en el estudio de las vulnerabilidades desde el punto de vista de la seguridad de software, en donde se definen a estas como la deficiencia de un sistema operativo o aplicación, la cual puede ser explotada por usuarios no autorizados con fines no genuinos. Así mismo, se recomienda establecer una metodología para clasificar a las vulnerabilidades de software y que complemente los procedimientos de la seguridad de acceso al código. Dicha metodología servirá como marco de referencia para identificar las vulnerabilidades de las aplicaciones presentes en la organización y tomar la acción preventiva adecuada. Cada organización desarrollará su metodología de acuerdo a las características específicas de su arquitectura de software, sin embargo se recomienda partir de la siguiente clasificación básica de vulnerabilidades:

- De bajo riesgo: Son aquellas que permiten la obtención de información secundaria, tal como sistema operativo, versiones de aplicaciones utilizadas, entre otras.
- De medio riesgo: Aquellas que permiten extraer información más relevante acerca del objetivo, como pueden ser nombres de cuentas de usuario, así como las que permiten ataques que pueden comprometer parcialmente el desempeño y buen funcionamiento de los sitios atacados.
- De alto riesgo: Son aquellas que pueden representar el compromiso total del sistema, permitiendo al atacante el control total del objetivo o, al menos, impedir completamente su funcionamiento.

Para futuras investigaciones que vayan en la línea de este estudio se recomienda el estudio de las características del código administrado y su integración con las siguientes evoluciones de los sistemas operativos "Windows" .La metodología CAS pertenece al área de compiladores que se dicen de código administrado porque dicho código es supervisado durante su ejecución. En caso de errores, como podrían ser excepciones de seguridad, estos se capturan en cuanto se provocan. Es decir mediante el código administrado, el motor de ejecución de la plataforma .NET realiza en tiempo real un control automático del código para que este sea seguro. Sin embargo, el código administrado y las técnicas CAS introducen factores de sobrecarga que repercuten en la demanda de más requisitos del sistema. La anterior a partir del hecho de que el consumo de recursos del procesador y memoria para supervisar el código es mucho mayor durante la ejecución. En este orden de ideas se recomienda a los interesados en la implementación de técnicas CAS, ahondar en el desarrollo de los sistemas automáticos de administración de memoria y recursos de procesador.

Obras consultadas

Bibliográficas

Seara, Daniel, "Seguridad en aplicaciones desarrolladas con Microsoft.net Framework", Universidad.Net. 2003.

Robinson, Ed, "Seguridad de acceso al código", Primera edición, Mc Graw Hill 2003.

Ardita, C., Julio. "Seguridad en el comercio electrónico".Primera edición, CYBSEC, 2000.

Payne, Chris. "Aprendiendo ASP.NET". Primera Edición, Pearson Educación, 2002.

RockWell, Michael. "Programación con Microsoft Windows DNA". Primera Edición, Pearson Educación, 2001.

Cassidy Wayne. "Visual Basic .NET Web Applications". Primera Edición, Mc Graw Hill, 2002.

Panagrosso, David. "Visual Basic.NET Windows Applications". Primera Edición, Mc Graw Hill, 2002.

Lind S. Keneth. "XML Web Services and Server Componentes Development with Visual Basic .NET". Primera Edición, Mc Graw Hill, 2003.

Electrónicas

Sánchez F. Albert. "Security response". 2003.
http://www.telecable.es/personales/fontecha/seguridad.doc#_Toc18070369
Enero 20, 2005.

Carmona, David. "Seguridad en .NET". 2004.
http://www.mslatam.com/latam/technet/cso/HTMLes/FrontEnd/Users/micuenta/micuenta.asp
Enero 15, 2005.

Aparicio, Jorge E. "Un estudio sobre el estado de la seguridad informática en México". 2001.
http://www.jmgarcia.com.mx/articulos/estudio.pdf#search='vulnerabilidad%20inform%C3%A1tica'
Febrero 28, 2005.

Menchaca, M. Rolando. "Arquitectura de la máquina virtual Java". 2000.
http://www.revista.unam.mx/vol.1/num2/art4/
Marzo 5, 2005.

Biblioteca digital

MacDonald, Mathew. (2004). Safeguard VB.NET applications with Windows-Type Security [Versión electrónica], *Inside Microsoft Visual Basic, Proquest Computing, 5.*

Conry, M., Andrew. (2002). Behavior-blocking stops unknown malicious code [Versión electrónica], *Network Magazine, Proquest Computing, 50.*

Fisher, Jason. (2002). Exploring the new frontiers of .NET [Versión electrónica], *Active Server Developer's Journal, Proquest Computing, 4.*

O'reilly, Associates (2003). Addressing Windows vulnerabilities at the application level [Versión electrónica], *Programming .NET Security, Proquest computing.*

Joachim, David (1997). Sun changes security framework for Java Development Kit [Versión electrónica], *Business Information's Internet Week, Proquest computing, 9.*

Anexo 1

Guía de entrevista

1. ¿Cuál es el giro de la compañía en la que labora?
2. ¿Cuál es el puesto que ocupa en la compañía y las principales características de este?
3. Señale cuales de las siguientes amenazas en materia de seguridad informática ha experimentado su compañía:
    - Virus.
    - Modificación de información.
    - Interrupción de servicios.
    - Software malicioso.
    - Acceso no autorizado a los equipos de la red de área local.
    - Acceso no autorizado a los sistemas y bases de datos.
    - Uso de recursos no autorizados por parte del usuario.
    - Intercepción de información cuando esta se traslada por una red pública.

4. ¿Alguna vez la compañía ha experimentado consecuencias negativas en función de ataques de índole informática (virus, hackers, intrusión) ?
5. ¿La compañía tiene alguna estrategia de seguridad de información , y si la tiene, cual es su característica?
6. ¿La compañía cuenta con alguna de las siguientes herramientas: ?
    - Diagnóstico de seguridad
    - Diagnóstico de vulnerabilidades y riesgos
    - Planeación y administración de la seguridad informática
7. ¿Cuáles son los recursos y presupuesto utilizado para los problemas de seguridad en la información?
8. ¿La compañía cuenta con un experto interno o mediante un esquema de "outsourcing" para alguna de las siguientes áreas: ?
    - Cifrado.
    - Autorización basada en roles.
    - Seguridad de acceso al código.
    - Autenticación de seguridad integrada en sistemas operativos.
    - Seguridad de arquitectura.
9. ¿Conoce los conceptos y la teoría de seguridad basada en dotar de permisos de ejecución al código que compone a las aplicaciones de software?
10. ¿La alta gerencia de la compañía está consciente y participa en la estrategia de seguridad informática?

Anexo 2

Ejemplo

El ejemplo descrito en el anexo 2 tiene por propósito ejemplificar la forma en que la seguridad de acceso al código puede proteger los accesos de lectura con respecto a los archivos almacenados en un disco duro en específico. Se plantea el escenario según el cual, un componente desarrollado en el lenguaje C# que escribe sus resultados en la línea de comandos, podría ser utilizado por un programa de código malicioso o un usuario no autorizado para abrir y desplegar en la línea de comandos, el contenido de un archivo .txt . Se parte de la base de que el usuario que intenta utilizar el componente ha logrado burlar la seguridad de acceso al usuario, basada en el sistema operativo o que simplemente dicho componente se ejecuta en una computadora de red de área local en donde no es necesario la autenticación de los usuarios para ejecutar las aplicaciones.

En este código se describe la clase que imprime el contenido del archivo Readme.txt:

```
public class myClass2
{
        public static void myOpenFile()
        {
                try
                {       StreamReader din = File.OpenText(@"C:\ReadMe.txt");
                        Console.WriteLine("File opened: " + din.ReadLine());
                }
                catch
                {
                        Console.WriteLine("Error al abrir archivo");
                }
        }

}
```

En este código se describe la restricción del uso del método "myOpenFile" mediante la denegación del uso del permiso "FileIOPermission", lo que ejemplifica una técnica CAS.

```
using System;
using System.IO;
using System.Security;
using System.Security.Permissions;

namespace WindowsApplication3
{
        public class myClass1
        {
                static void Main(string[] args)
                CodeAcessPermission myPermission = new
                        FileIOPermission(FileIOPermissionAccess.AllAccess,@"C:\");
                                //Creación del permiso "myPermission" que controla el
                                //acceso de lectura a los archivos del directorio c:
                myPermission.Deny();
                                //Establecimiento del modo de  lectura de archivos en
                                //denegado.
                MyClass2.MyOpenFile;
        }
}
```

Al ejecutar esta aplicación, el resultado será el siguiente mensaje de texto: "Error al abrir archivo". Cabe señalar que en el código anterior el archivo Readme.txt no se puede abrir en función de la sentencia "myPermission.Deny". Dicho permiso se aplica independientemente del usuario que ejecute la clase, es decir, la seguridad de usuario basada en roles o en sistema operativo no sobrescribe a la seguridad CAS. El programador del componente que utilice a la clase myClass2 y su método "myOpenFile" podrá dar permiso de lectura al archivo "ReadMe.txt" mediante la sentencia "CodeAccessPermission.RevertDeny()". Esta última sentencia la incluirá el programador en el código de su aplicación a partir de realizar una evaluación sobre el origen del usuario o software que requiere ejecutar al método "myOpenFile".