



Manual del usuario de PreEmptive Dotfuscator
and Analytics
Community Edition

Versión 5.5

PreEmptive Solutions

© 2002-2012 by PreEmptive Solutions, LLC

Reservados todos los derechos.

Versión del manual 5.5-092911

www.preemptive.com

MARCAS REGISTRADAS

Dotfuscator, Overload-Induction, el logotipo de PreEmptive Solutions y el logotipo de Dotfuscator son marcas registradas de PreEmptive Solutions, LLC

.NET™, MSIL™ y Visual Studio™ son marcas comerciales de Microsoft, Inc.

Todas las otras marcas registradas son propiedad de sus respectivos propietarios.

ESTA PUBLICACIÓN SE PROPORCIONA “TAL CUAL” SIN GARANTÍA DE NINGÚN TIPO, YA SEA EXPRESA NI IMPLÍCITA, INCLUIDAS, AUNQUE SIN LIMITACIÓN, LAS GARANTÍAS DE COMERCIALIZACIÓN, IDONEIDAD PARA UN FIN DETERMINADO O AUSENCIA DE INFRACCIÓN. ESTA PUBLICACIÓN PODRÍA CONTENER ERRORES TIPOGRÁFICOS O IMPRECISIONES TÉCNICAS. ESTE DOCUMENTO Y/O EL SOFTWARE COMPLEMENTARIO PUEDEN SUFRIR ACTUALIZACIONES Y MODIFICACIONES EN CUALQUIER MOMENTO.

PreEmptive Solutions, LLC tiene derechos de propiedad intelectual referentes a la tecnología utilizada en este producto. En particular, y sin limitación, estos derechos de propiedad intelectual pueden incluir una o varias patentes en EE.UU. o solicitudes de patentes pendientes en EE.UU. y/o en otros países.

Este producto se distribuye bajo licencias que restringen su uso, copia, distribución y decompilación. Ningún componente de este producto puede reproducirse de ninguna forma ni por ningún medio sin autorización previa por escrito de PreEmptive Solutions, LLC.

Tabla de Contenidos

. Tabla de Contenidos	0
. PreEmptive Dotfuscator and Analytics CE	1
. Introducción	2
. Funciones	3
. Actualizaciones	4
. Primeros pasos	5-8
. Entradas	9-11
. Propiedades	12-15
. Opciones de configuración	16
. Cambiar el nombre	17-20
. Analytics	21-27
. Resultados	28
. Más allá de la ofuscación	29
. Análisis de las aplicaciones	30
. Tipos de aplicaciones .NET admitidas	31
. Seguimiento de excepciones	32
. Informe de excepciones y el atributo ExceptionTrack	33
. Recopilar la información del informe de excepciones especificada por el usuario	34-35
. Introducción a la defensa activa contra la alteración	36
. Simular la alteración	37
. Tipos de aplicaciones .NET admitidas	38
. Expiración de las aplicaciones	39
. Introducción a las claves de activación de vida útil	40
. Portal gratuito de Runtime Intelligence Services	41
. Índice	42-47

PreEmptive Dotfuscator and Analytics CE

Dotfuscator es el principal compactador y motor de ofuscación .NET que ayuda a proteger los programas contra la ingeniería inversa al tiempo que consigue reducir su tamaño y aumentar su eficacia. Además, Dotfuscator and Analytics proporciona también un modo de inyectar funcionalidad adicional previamente compilada que ofrece seguimiento del uso, detección de alteraciones y expiración en las aplicaciones .NET.

En la documentación de PreEmptive Dotfuscator and Analytics CE

[Introducción](#)

Se explican las ventajas del uso de PreEmptive Dotfuscator and Analytics Community Edition 5.5.

[Primeros pasos](#)

Se explica cómo iniciar PreEmptive Dotfuscator and Analytics Community Edition 5.5 y registrar el producto.

[Opciones de configuración](#)

Se explica cómo ofuscar la aplicación mediante el cambio de nombre. Después se explica cómo usar las capacidades de inyección de código de Dotfuscator para realizar el seguimiento del uso de las funciones, detectar las alteraciones y controlar la expiración en las aplicaciones.

[Resultados](#)

Muestra una representación visual de la aplicación una vez que Dotfuscator and Analytics la recompila.

[Más allá de la ofuscación](#)

Explica la nueva categoría de control de detección de Dotfuscator and Analytics CE, que permite supervisar, administrar y proteger mejor las aplicaciones

[Portal gratuito de Runtime Intelligence Services](#)

Describe los informes y paneles disponibles de forma gratuita que permiten obtener una perspectiva de cómo se usan las aplicaciones.

Introducción

Su copia de Microsoft Visual Studio 11 incluye una licencia gratuita de *PreEmptive Dotfuscator and Analytics Community Edition 5.5* (Dotfuscator and Analytics CE 5.5). Al igual que las versiones anteriores de Dotfuscator incluidas en Visual Studio 2010, 2008, 2005 y 2003, Dotfuscator and Analytics CE proporciona herramientas para proteger y reforzar las aplicaciones .NET. Dotfuscator and Analytics CE 5.5 funciona con ensamblados compilados sin necesidad de realizar ninguna programación adicional ni tener acceso al código fuente.

Además, Dotfuscator and Analytics CE 5.5 ofrece diversos servicios para desarrolladores, arquitectos y evaluadores. Algunos ejemplos de las nuevas capacidades de protección, supervisión y administración incluidas en Dotfuscator and Analytics CE 5.5 son:

- *Seguimiento de excepciones* para supervisar las excepciones no controladas, controladas o iniciadas dentro de la aplicación.
- *Defensa ante la alteración* para detectar la ejecución de las aplicaciones alteradas, transmitir alertas de incidentes y terminar las sesiones alteradas.
- Comportamientos ante la *expiración de las aplicaciones* que codifican la fecha de “fin del período de vigencia”, transmiten alertas cuando las aplicaciones se ejecutan después de la fecha de expiración y/o terminan las sesiones de las aplicaciones que han expirado.
- *Seguimiento de las sesiones* para determinar qué aplicaciones se han ejecutado, las versiones de dichas aplicaciones y durante cuánto tiempo.
- *Seguimiento del uso de las sesiones* para determinar qué funciones se están usando, en qué secuencia y durante cuánto tiempo.

En esta sección

[Funciones](#)

[Actualizaciones](#)

Funciones

Esta sección se centra en las capacidades de Dotfuscator and Analytics CE 5.5 con algunas referencias a las opciones avanzadas disponibles en las actualizaciones.

Dotfuscator and Analytics CE 5.5 es un sistema *posterior a la compilación* destinado a las aplicaciones .NET. Con Dotfuscator and Analytics CE 5.5, los usuarios de Visual Studio puede ofuscar los ensamblados e inyectar funciones para la detección de alteraciones, la expiración de las aplicaciones, la supervisión de las sesiones y el seguimiento de características, todo ello sin necesidad de programar ni de tener acceso al código fuente original. Esta información puede dirigirse a uno o varios extremos (o nubes) para realizar informes y análisis.

Protección de las aplicaciones

El robo de la propiedad intelectual, la piratería y la alteración por sí solas incrementan materialmente el riesgo financiero, operativo y para la reputación tanto de los proveedores como de los consumidores. Dotfuscator and Analytics CE 5.5 incluye una ofuscación de línea base que persigue reducir el riesgo de que se produzcan accesos no autorizados al código fuente a través de técnicas de ingeniería inversa.

Defensa de las aplicaciones

Los avances en las arquitecturas de las aplicaciones y la informática distribuida han hecho que sus componentes sean cada vez más vulnerables a los ataques y que resulte más difícil establecer técnicas de defensa caso por caso y sesión por sesión. Dotfuscator and Analytics CE 5.5 incluye la capacidad de inyectar lógica que detecta la ejecución de código alterado y, si se desea, transmitir una alerta de incidente a un extremo especificado por el desarrollador y/o terminar la sesión alterada, y todo ello en tiempo real y a través de distintos dispositivos, plataformas y modelos de distribución.

Medida de las aplicaciones

El acceso a información detallada relativa al comportamiento de las aplicaciones .NET una vez que han sido implementadas para los usuarios, ya sea en su versión beta, en la versión inicial o cuando están disponibles de forma generalizada, siempre ha sido un objetivo de los desarrolladores de aplicaciones. Dotfuscator and Analytics CE 5.5 puede ayudar a resolver cuestiones como:

- Con qué frecuencia se bloquea una aplicación y con qué frecuencia se encuentran excepciones controladas
- Qué aplicaciones ejecutan los usuarios y qué versiones.
- Qué funciones se ejecutan y en qué orden.
- Cuanto tardan en ejecutarse dichas funciones y cuál es la tendencia a lo largo del tiempo.
- Qué grado de estabilidad experimentan los usuarios con las aplicaciones en su uso práctico.

Tradicionalmente, lo que supone recopilar esta información del rendimiento del usuario final es que algunos equipos de desarrollo deban seguir los pasos para tener acceso a la misma. Y cuando lo hacen, la complejidad derivada de tener que mantener los análisis hace que el proceso sea frágil y resulte difícil adaptarlo a los cambios.

Dotfuscator and Analytics CE 5.5 facilita a los desarrolladores, evaluadores y arquitectos la obtención de esta información con apenas ningún efecto en el rendimiento del usuario final. Dotfuscator and Analytics CE 5.5 puede inyectar la supervisión de sesiones, características y excepciones hasta el nivel de método, y transmitir datos en tiempo de ejecución a uno o varios extremos. Tanto si se encuentra administrando el ciclo de una versión beta, en una cartera de servicios Web o dentro del ciclo de vida de la evaluación de un producto, Dotfuscator and Analytics CE 5.5 puede proporcionar una perspectiva eficaz para la adopción, el comportamiento del usuario, la estabilidad y, a la larga, las consecuencias del software en el negocio.

Administración de aplicaciones

La administración de la cartera de aplicaciones se está convirtiendo en una cuestión esencial para los ISV y empresas similares, lo que significa que está llegando a ser un problema para los desarrolladores. A medida que el software se vuelve más fácil de generar y más difícil de seguir, las organizaciones buscan formas más seguras y confiables de codificar las reglas de negocios y las directivas de TI directamente en él.

El desafío ha pasado a ser cómo satisfacer estos requisitos sin necesidad de efectuar cambios significativos en el software. Para abordar este reto, Dotfuscator and Analytics CE 5.5 puede aplicar fechas tope y transmitir secuencias de audio para realizar el seguimiento del uso pretendido o tras la fecha de expiración de una aplicación.

Actualizaciones

Es conveniente que los desarrolladores profesionales que requieren soporte técnico, mantenimiento y actualizaciones de productos de forma continuada actualicen su copia de Dotfuscator and Analytics CE 5.5 con una edición comercial de Dotfuscator que satisfaga las necesidades únicas de su organización de desarrollo de aplicaciones. Para obtener más información sobre las capacidades avanzadas de seguridad de las aplicaciones, supervisión, medición y administración del producto completo de Dotfuscator, incluida una comparación detallada de todas las ediciones de Dotfuscator, consulte [Información general sobre la familia Dotfuscator](#). Si desea una versión de evaluación de Dotfuscator con soporte técnico completo, solicítela.

Las versiones comerciales de Dotfuscator incluyen una seguridad mejorada, una integración más estrecha con Visual Studio y otras extensiones importantes de las funciones de administración y supervisión de Dotfuscator and Analytics CE 5.5. Por ejemplo:

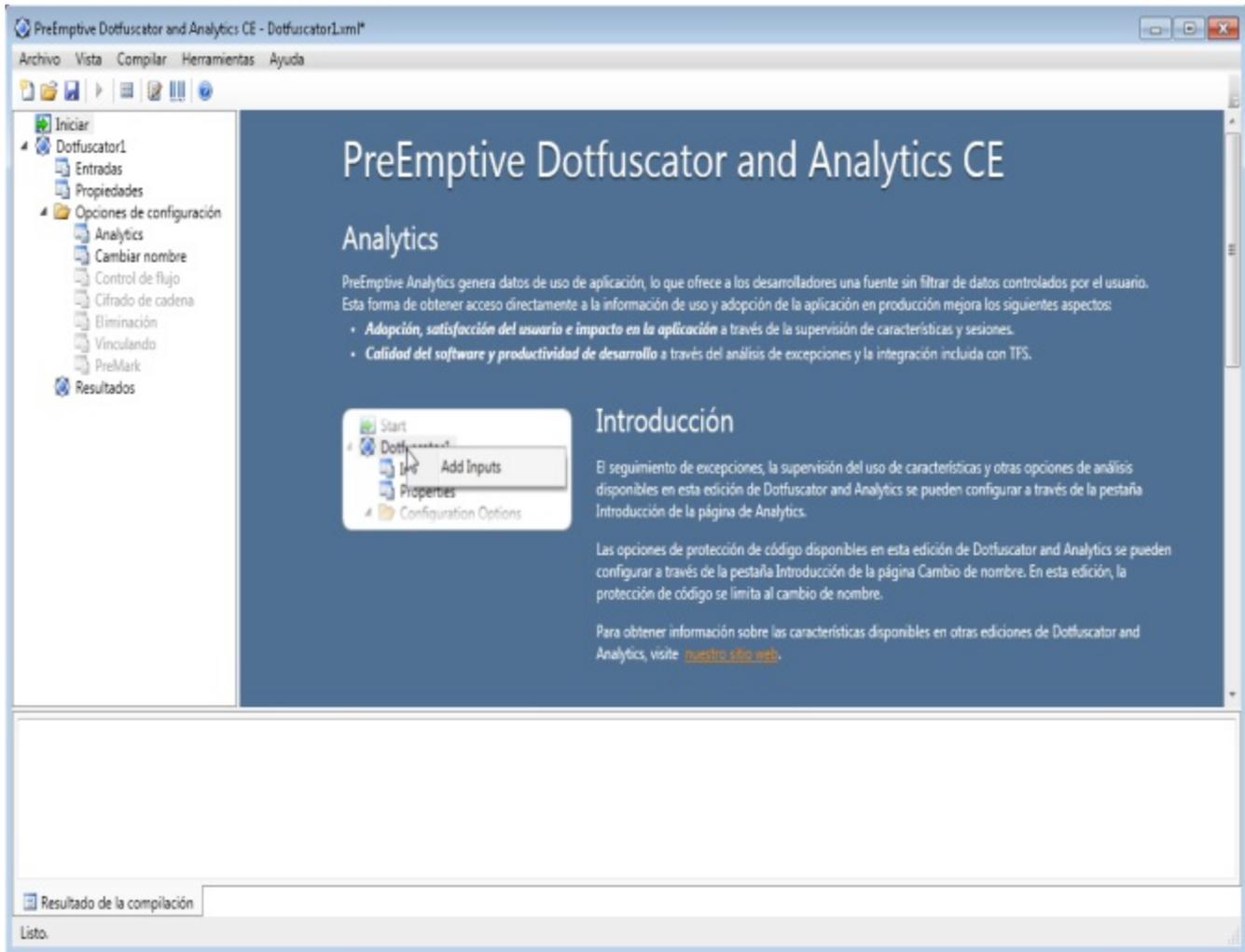
- *Protección de las aplicaciones*: acceso a toda la gama de transformaciones de ofuscación expandidas, así como la capacidad de integrar marcas de agua y de vincular o eliminar ensamblados.
- *Defensa de las aplicaciones*: capacidad de inyectar comportamientos de defensa de las aplicaciones y activar servicios de informe de incidentes de alteración.
- *Medida de las aplicaciones*: capacidad de crear "señales" de datos extensibles, crear transmisiones cifradas, realizar el seguimiento de un número ilimitado de características y métodos, tener acceso a un variado conjunto de servicios analíticos de aplicaciones y la posibilidad de generar informes sobre excepciones detectadas o iniciadas (además de las no controladas).
- *Administración de las aplicaciones*: capacidad de inyectar comportamientos personalizados, agregar una fecha de advertencia, obtener analíticas de informes y cifrar las transmisiones.

Primeros pasos

Para empezar a usar Dotfuscator and Analytics CE, realice estos pasos:

- Inicie Visual Studio.
- En la barra de menús de Visual Studio, haga clic en **Herramientas > PreEmptive Dotfuscator and Analytics**.

Cuando se abra PreEmptive Dotfuscator and Analytics CE, se mostrará la pantalla de inicio en su escritorio:



El programa consta de tres paneles: el árbol de navegación, el área de trabajo y los resultados de la compilación.

En la pantalla Inicio, puede realizar varias acciones. Si se habilita el [Contenido dinámico](#), puede ver las notas de la [versión más reciente](#) de Dotfuscator Pro y las últimas noticias de PreEmptive Solutions. Si no se habilita el [Contenido dinámico](#), la opción para configurar las preferencias de usuario aparecerá en la pantalla de inicio. La opción para [registrar Dotfuscator and Analytics](#) está visible en todo momento. El contenido dinámico se puede habilitar o deshabilitar en [Preferencias de usuario](#).

Configurar las preferencias de usuario

En la pantalla Inicio, si el contenido dinámico está deshabilitado, hay un vínculo en el que puede hacer clic para establecer las preferencias de usuario o puede hacer clic en **Herramientas > Preferencias de usuario**. Independientemente de la ruta de acceso que elija para llegar a Preferencias de usuario, aparece este cuadro de diálogo:

The screenshot shows a dialog box titled "Preferencias de usuario" with the following sections and controls:

- Configuración de red:**
 - Dirección de proxy: [Text input field]
 - Puerto de proxy: [Text input field]
 - Nombre de usuario de proxy: [Text input field]
 - Contraseña de proxy: [Text input field]
- Aplicación de visor XML:**
 - Ruta de acceso de aplicación: [Text input field] [Examinar.]
- Noticias y actualizaciones:**
 - Permitir contenido dinámico en el panel de inicio
 - Permitir que Dotfuscator and Analytics busque actualizaciones periódicamente

At the bottom of the dialog, there is a link: [Lea nuestra directiva de privacidad en línea.](#) and two buttons: "Aceptar" and "Cancelar".

Si es necesario, escriba los detalles de configuración de los requisitos del servidor proxy en la sección *Configuración de red* del cuadro de diálogo Preferencias de usuario. La información de proxy no se necesita si no tiene un servidor proxy o si su configuración se controla a través de Internet Explorer.

Si desea usar un visor XML para ver los archivos XML de Dotfuscator XML que *no* sea su visor predeterminado del sistema, puede hacer clic en **Examinar...** en la sección *Aplicación de visor XML* y buscar la aplicación que desee usar.

En la sección *Noticias y actualizaciones*, puede optar por permitir que Dotfuscator busque periódicamente si hay actualizaciones. También es en esta sección donde puede habilitar el Contenido dinámico para que se muestre en la pantalla Inicio.

Última versión de Dotfuscator and Analytics

El área de trabajo contiene un vínculo a las notas de la versión más reciente de Dotfuscator and Analytics. Si el Contenido dinámico está deshabilitado, este vínculo le llevará a la página del historial de versiones de Dotfuscator and Analytics en www.preemptive.com.

Estado del registro

Puede registrar Dotfuscator and Analytics CE a través del vínculo **Registrarse ahora**. Al hacer clic en este vínculo, se muestra el cuadro de diálogo de registro.

Registro de Dotfuscator and Analytics Community Edition

Registre su copia de Dotfuscator and Analytics para obtener acceso a actualizaciones y soporte técnico en línea.

PreEmptive Solutions le enviará por correo electrónico noticias, productos actualizados y servicios (puede darse de baja si no desea recibirlos).

Importante: se requiere una dirección de correo electrónico válida y se utilizará para enviarle el código de confirmación.

Nombre: *

Apellidos: *

Dirección de correo electrónico: *

Número de teléfono: *

Compañía/asociación: *

Título: *

Programador

Programador senior

Analista de sistemas

Dirección 1:

Dirección 2:

Ciudad:

Provincia o estado:

Código postal:

País o región:

¿Dónde conoció Dotfuscator?

Registrarse por correo electrónico

Registrarse por Web

Introduzca el nombre en el campo de nombre.

[Lea nuestra directiva de privacidad en línea.](#)

Siguiente > Cancelar

Registre su producto si desea recibir actualizaciones y mejoras para Dotfuscator and Analytics CE. Escriba la información que se solicita en cada uno de los campos. Para automatizar el envío, seleccione **Registrarse por web**.

Al hacer clic en **Siguiente**, se le dirige a la segunda página del cuadro de diálogo Registro. Si es necesario, especifique los detalles de configuración del servidor proxy para la red. Haga clic en **Enviar** cuando haya terminado.

Una vez que la solicitud de registro se reciba y procese, recibirá un mensaje de correo electrónico de confirmación de PreEmptive Solutions que contendrá el código de comprobación del registro y las credenciales para el área de soporte técnico registrada de www.preemptive.com, donde podrá tener acceso a las versiones actualizadas de Dotfuscator and Analytics CE.

Ahora está preparado para comenzar a ofuscar e instrumentar su aplicación con Dotfuscator and Analytics CE.

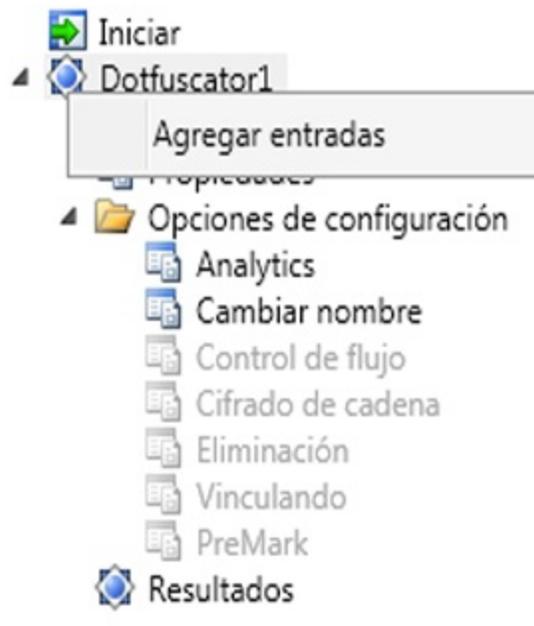
En esta sección

[Entradas](#)

[Propiedades](#)

Entradas

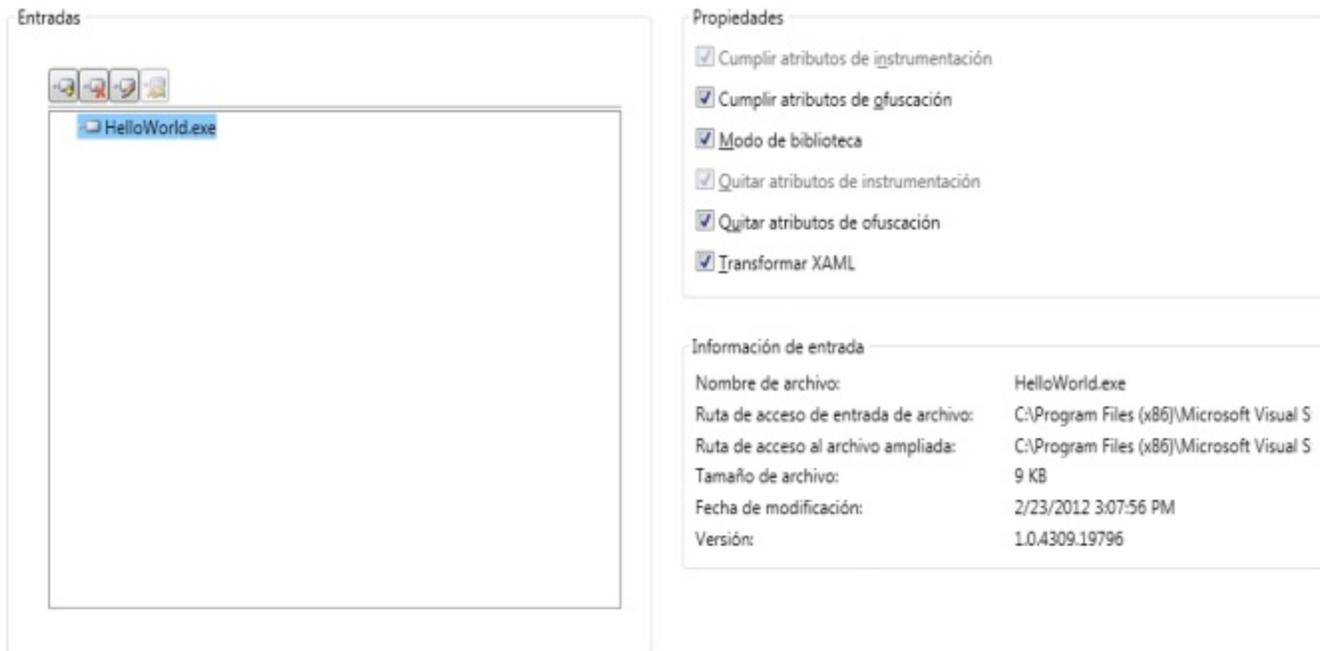
Para comenzar a usar Dotfuscator and Analytics CE, agregue uno o varios paquetes o ensamblados de entrada al proyecto. Para agregar entradas al proyecto, debe tener abierta la pantalla Entradas en el área de trabajo. Para activar esta pantalla, haga clic con el botón secundario en el nodo del proyecto en el árbol de navegación y seleccione **Agregar entradas**, o seleccione la opción Entradas en el árbol de navegación.



Para agregar los ensamblados o paquetes de entrada a la lista de entradas, puede arrastrar y colocar los directorios o archivos .exe o .dll en la sección *Entradas* de la página.

También puede hacer clic en el botón "Agregar entrada" o hacer clic con el botón secundario del mouse en el área *Entradas* y elegir Agregar entrada. Cuando agregue entradas de esta manera, puede examinar el lugar donde se encuentra su entrada y seleccionarla o escribir la ruta de acceso a la entrada manualmente.

Propiedades de entrada



En la sección *Propiedades* del panel del área de trabajo, seleccione o anule la selección de:

Cumplir atributos de instrumentación: al seleccionar *Cumplir atributos de instrumentación*, se indica que Dotfuscator procese estos atributos y realice las transformaciones de instrumentación indicadas en el ensamblado de destino. Al anular la selección de esta opción, se indica a Dotfuscator que debe omitir los atributos de instrumentación.

 [Los atributos de instrumentación](#) son atributos personalizados que se pueden aplicar en el código de origen para realizar el seguimiento de la estabilidad, funciones y uso de las aplicaciones, además de agregar la funcionalidad de vida útil.

Cumplir atributos de ofuscación: al seleccionar *Cumplir atributos de ofuscación*, se indica que Dotfuscator procese estos atributos y realice las transformaciones de ofuscación indicadas en el ensamblado de destino. Al anular la selección de esta opción, se indica a Dotfuscator que debe omitir los atributos de ofuscación.

 Los atributos de ofuscación son atributos personalizados que se pueden aplicar en el código fuente para declarar de forma explícita la inclusión o exclusión de tipos, métodos, enumeraciones, interfaces o miembros de varios tipos de ofuscación. El atributo que se usaría para incluir o excluir tipos, métodos, enumeraciones, interfaces y miembros de la ofuscación es **System.Reflection.ObfuscationAttribute**. Si deseara denotar que un ensamblado específico tendrá sus elementos incluidos o excluidos de la ofuscación, debería usar **System.Reflection.ObfuscateAssemblyAttribute**. Dotfuscator and Analytics CE solo admite el cambio de nombre. Hay más funciones de ofuscación adicionales disponibles en la versión comercial de Dotfuscator.

Modo de biblioteca: este valor le indica a Dotfuscator que el ensamblado de entrada seleccionado constituye una biblioteca. Para la ofuscación, una biblioteca se define como un ensamblado al que se hace referencia desde otros componentes no especificados como una de las entradas de esta ejecución. Cuando un ensamblado se ofusca en el modo de biblioteca, los elementos visibles públicamente no se cambian de nombre, con lo que se hace que su API pública siga estando accesible para los que la llaman de forma externa.

Quitar atributos de ofuscación: Dotfuscator puede quitar todos los atributos de ofuscación cuando se complete el procesamiento, de modo que los ensamblados no contengan pistas sobre cómo se ofuscaron. Al seleccionar esta opción, se indica que Dotfuscator quite estos atributos del ensamblado de salida de destino. Al desactivar esta opción se indica que Dotfuscator deje los atributos en el ensamblado de salida a menos que los atributos individuales designen que deberían quitarse a través de la propiedad StripAfterObfuscation.

Transformar XAML: Dotfuscator puede cambiar el nombre de los elementos de los recursos XAML que se usan en las aplicaciones Silverlight, así como los recursos XAML compilados (BAML) de las aplicaciones de Windows Presentation Foundation. El valor predeterminado es Verdadero, que indica a Dotfuscator que intente cambiar el nombre de los elementos de los recursos de marcado y que el nuevo nombre coincida con las referencias a los elementos en el código subyacente. Si se deja esta opción habilitada, se refuerza significativamente la ofuscación de las aplicaciones de Windows Presentation Foundation y Silverlight, y también se reduce el número de elementos que deben excluirse manualmente de la operación de cambio de nombre.

Información de entrada

Esta sección del área de trabajo Entradas proporciona datos generales acerca del ensamblado seleccionado actualmente, incluidos el nombre de archivo, la ruta de acceso al archivo, la ruta de acceso al archivo ampliada (útil si se usa `${configdir}` u otras propiedades), el tamaño de archivo, la fecha de la última modificación y la versión.

Propiedades

El elemento **Propiedades** del árbol de navegación lleva al área de trabajo de configuración de propiedades. Contiene seis pestañas, tres de las cuales están disponibles para los usuarios de Dotfuscator and Analytics CE.

Propiedades del proyecto

Se puede considerar *Propiedades del proyecto* como simples macros de sustitución de cadenas que pueden usarse siempre que se requiera un nombre de archivo o una ruta de acceso. En la pantalla **Propiedades del proyecto** se pueden ver, eliminar y agregar manualmente pares de nombre y valor definidos por el usuario, como propiedades del proyecto; también se pueden ver las propiedades externas. Se consideran propiedades externas las que define Dotfuscator (como `configdir` o `appdatadir`) o las que se pasan, a través del modificador `/p`, en la línea de comandos. Las propiedades del proyecto son especialmente útiles al crear plantillas de compilación de ofuscación para poder usar varios proyectos y configuraciones. Para hacer referencia a las propiedades, se usa la sintaxis siguiente:

Sintaxis de referencia de las propiedades

```
${nombre_propiedad}
```

Las referencias a las propiedades distinguen mayúsculas y minúsculas, por lo tanto `${MiDirDeProy}` hace referencia a una propiedad diferente de `${midirdeproy}`. Las referencias a las propiedades se interpretan literalmente y pueden no estar anidadas. En este momento, las referencias a propiedades solo se pueden usar como valores en los atributos `dir` or `name` del elemento `<file>`.

Dotfuscator usa el algoritmo siguiente para encontrar un valor asociado a la propiedad:

- Se busca un valor en la lista de propiedades externas.
- Si no se encuentra, se busca una variable de entorno con el mismo nombre que la propiedad.
- Si no se encuentra, se busca una propiedad del proyecto.
- Si sigue sin encontrarse, se usa una cadena vacía como valor.

Propiedades del proyecto	
Rutas de acceso de carga de ensamblados	
Cadenas de asignaciones de función	
Configuración de la compilación	
Generar eventos	
Firma	

Propiedades externas:	
Propiedad	Valor
configdir	C:\Users\Test\Documents
appdatadir	C:\ProgramData\PreEmptive Solutions\Common
applicationdir	C:\Program Files (x86)\Microsoft Visual Studio 11.0\PreEmptive Solutions\Dotfuscator and Analytics Community Editic

Propiedades del proyecto:	
Propiedad	Valor
projectdir	myproject

Propiedades externas

La columna *Property* contiene el nombre de la propiedad y la columna *Value* contiene su valor. Dotfuscator and Analytics CE define las tres propiedades y valores siguientes:

- `applicationdir` refleja el directorio de instalación de Dotfuscator and Analytics, donde se encuentran los archivos de aplicación de Dotfuscator.
- `appdatadir` refleja el directorio de datos locales de Dotfuscator and Analytics.
- `configdir` refleja el directorio en el que reside el archivo de configuración del proyecto actual.

Propiedades del proyecto

En esta sección, se tiene en cuenta la definición y asignación de propiedades adicionales arbitrarias que se van a usar durante el proceso de compilación. Estas propiedades se evaluarán del mismo modo que las propiedades externas. En el ejemplo anterior, `projectdir` es la propiedad *Property* y `myproject` es su parámetro *Value*.

Cadenas de asignaciones de función

La sección Cadenas de asignaciones de característica se usa para la ofuscación declarativa, que se implementa a través de la decoración de atributos dentro del código fuente mientras se escribe. Los atributos que se usan para controlar la ofuscación declarativa son `System.Reflection.ObfuscateAssemblyAttribute` y `System.Reflection.ObufscationAttribute`. `System.Reflection.ObfuscateAssemblyAttribute` controla la ofuscación del ensamblado como un todo. `System.Reflection.ObufscationAttribute` controla la ofuscación de tipos individuales y de sus miembros. Cadenas de asignaciones de función permite declarar, dentro del código fuente, lo que se debe y no se debe ofuscar mediante atributos.

Propiedades del proyecto	Cadenas de asignaciones de función:	
Rutas de acceso de carga de ensamblados	Cadena de asignación	Funciones
Cadenas de asignaciones de función	testmode	renaming
Configuración de la compilación		
Generar eventos		
Firma		

En esta sección, puede agregar, modificar y quitar cadenas de asignaciones de función. Para agregar una cadena de asignación, haga clic en el campo *Cadena de asignaciones* y escriba el **nombre** de la cadena. Cuando lo haga, una fila de casillas se muestra debajo de la fila actual que contiene la función que seleccione. Por ejemplo,

- `testmode` es el **nombre** de la *Cadena de asignaciones* que se escribió
- `renaming` es la **Función** de la cadena de asignaciones que se seleccionó en la fila de abajo.

En Dotfuscator and Analytics CE, la única característica disponible es el cambio de nombre (renaming). Las versiones comerciales de Dotfuscator son compatibles con las funciones adicionales siguientes:

Cadena de función	Acción
*controlflow	atributo que configura la ofuscación del control de flujo.
*stringencryption	atributo que configura el cifrado de cadenas
*trigger	atributo que configura la eliminación marcando el elemento anotado como punto de entrada
*conditionalinclude	atributo que configura la eliminación incluyendo condicionalmente el elemento anotado

Al decorar los elementos con un atributo de ofuscación y asegurarse de que se establecen sus ensamblados de entrada para cumplir los atributos de ofuscación, puede especificar a qué elementos de la aplicación no se les debe cambiar el nombre. En el ejemplo de código siguiente se invoca su método a través de la reflexión y se debe excluir del cambio de nombre.

Invocado a través de reflexión, excluido del cambio de nombre:

```
[System.Reflection.Obfuscation(Exclude=true, Feature="renaming")]
public void CalledFromReflection(int someValue) {
```

Si tiene que excluir un método de la ofuscación en una configuración concreta de la compilación, cree una declaración de Asignación de función y establezca el valor de la función para el atributo de ofuscación de modo que coincida. En la captura de pantalla anterior, hemos establecido una cadena de asignación de función "testmode" que se usará para excluir elementos del cambio de nombre. En el ejemplo de código siguiente, hemos decorado nuestro método de modo que Dotfuscator cambie su nombre a menos que se defina la cadena de función "testmode".

Declaración de la asignación de función testmode:

```
[System.Reflection.Obfuscation(Exclude=true, Feature="testmode")]
public void OnlyCalledFromReflectionInTestMode(int someValue) {
```

Configuración de compilación

Aquí es donde se define el directorio de destino, la configuración de la compilación y de la ofuscación global.

Directorios

El *Directorio de destino*: se requiere ya que es donde residirá el resultado de la compilación. Por ejemplo, escribiría `${projectdir}\output` en este campo o haría clic en **Examinar** para determinar su ubicación.

Configuración de la compilación

Al seleccionar *Investigar únicamente* se muestra lo que ocurrirá como resultado del procesamiento de los ensamblados de entrada, sin escribirlos realmente. Esto resulta útil para crear archivos de asignaciones e informes.

Detalle del resultado de la compilación: permite determinar la cantidad

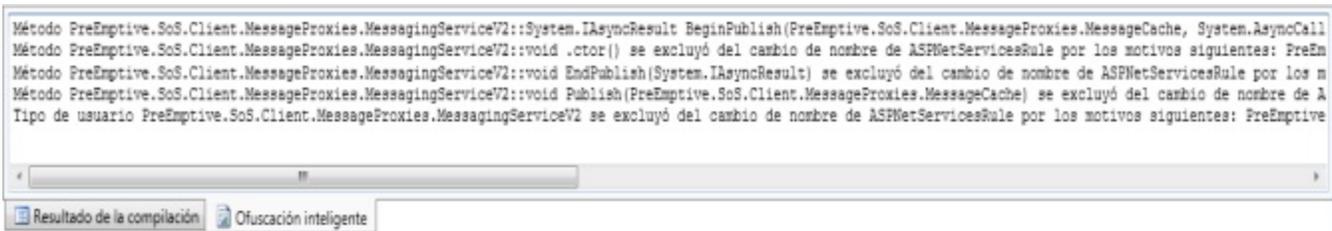
El *Directorio temporal*: es opcional y se usa para almacenar los archivos temporales durante el procesamiento. De forma predeterminada, Dotfuscator and Analytics CE usa el directorio temporal de Windows. Si desea especificar este directorio, escriba el **nombre** del directorio en este campo. Por ejemplo, podría escribir `${projectdir}\temp` en este campo o hacer clic en **Examinar** para determinar su ubicación.

de información que se presenta en el panel [Resultado de la compilación](#). Al seleccionar **No interactivo**, se consigue que la información resultante sea la mínima posible. Al seleccionar **Detallado**, se ofrece información detallada acerca de todas las acciones que ocurren durante el proceso de compilación. La opción Detallada es especialmente útil si se intenta optimizar los resultados.

Configuración de ofuscación global

En esta sección, se puede seleccionar o anular la selección de las opciones *Heredar atributos de ofuscación* y *Ofuscación inteligente*. *Heredar atributos de ofuscación* especifica si un atributo de ofuscación que se aplica a un tipo se debe aplicar también a los tipos derivados. *Ofuscación inteligente* usa el análisis estático para determinar qué elementos deben excluirse del cambio de nombre. A veces, una regla puede reconocer que es necesario realizar una acción, pero no puede determinar qué acción específica llevar a cabo porque el análisis estático no genera suficiente información. Cuando esto sucede, la regla emite una advertencia.

También puede determinar si desea recibir notificaciones de ofuscación inteligente si selecciona **Todos** para ver todos los mensajes, **Ninguno** para no ver ningún mensaje o **Solo advertencias**. Cuando se habilita la ofuscación inteligente, al compilar el proyecto, se genera un informe de ofuscación inteligente en su propia pestaña, junto a la pestaña Resultado de la compilación. Si se generan advertencias, se presentarán en la pestaña Advertencias de ofuscación inteligente.



Rutas de acceso de carga de ensamblado, eventos de compilación y firma

Estas funciones son exclusivas de Dotfuscator Professional Edition. Para solicitar información y una evaluación gratuita de 14 días (donde corresponda), haga clic en el botón [Probar ahora](#) en la barra de herramientas. Para saber cómo comprar Dotfuscator Professional Edition, haga clic en el botón [Comprar ahora](#) en la barra de herramientas. Otra alternativa es seleccionar estas opciones en el menú Ayuda.

Opciones de configuración

En la sección Opciones de configuración se pueden establecer las opciones y exclusiones para cambiar los nombres, elegir las reglas de cambio de nombre integradas y determinar la configuración de Analytics.

Opciones como Ofuscación del flujo de control, Cifrado de cadenas, Eliminación, Vinculación y Marcado previo son exclusivas de Dotfuscator Professional Edition. Para solicitar información y una evaluación gratuita de 14 días (donde corresponda), haga clic en el botón [Intentar ahora](#) de la barra de herramientas. Para obtener información sobre cómo comprar Dotfuscator Professional Edition, haga clic en el botón [Comprar ahora](#) de la barra de herramientas. Otra alternativa es seleccionar estas opciones en el menú Ayuda.

En esta sección

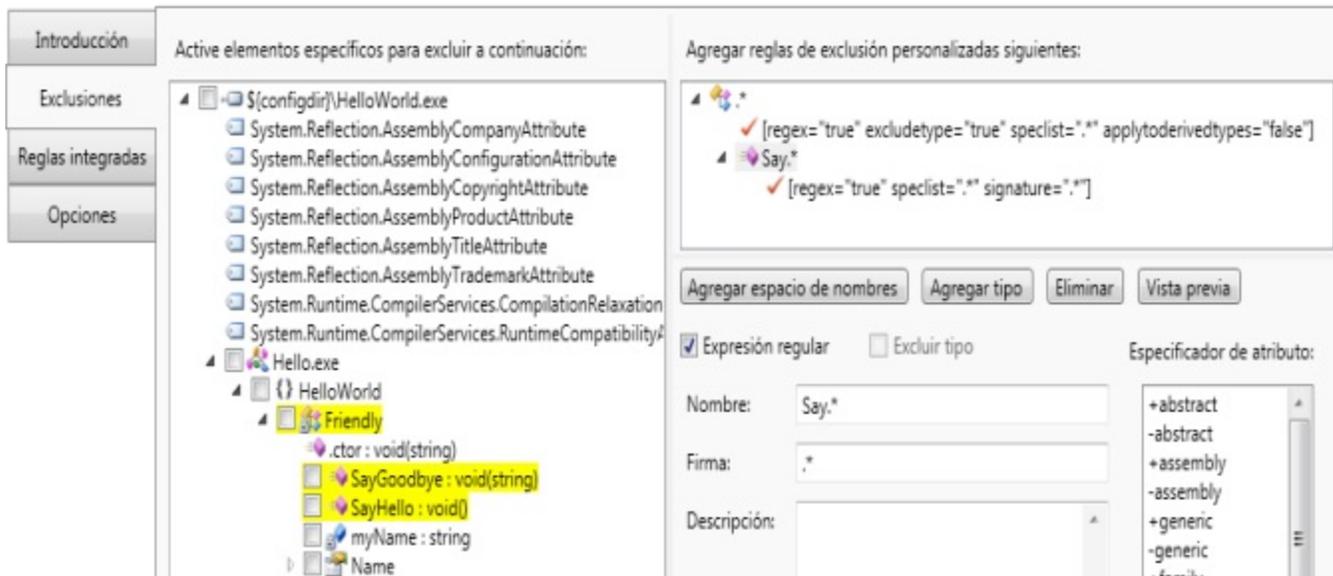
[Cambiar el nombre](#)

[Analytics](#)

Cambiar el nombre

El editor de cambio de nombre muestra cuatro pestañas de configuración: Introducción, que describe varias características para la protección de las aplicaciones y explica cómo configurar las opciones de cambio de nombre y las exclusiones; [Exclusiones](#) es una pestaña que sirve para definir gráficamente reglas de exclusión personalizadas; la pestaña [Opciones](#) se usa para configurar otras opciones relacionadas con el cambio de nombre; y [Reglas integradas](#) es una pestaña que muestra reglas de exclusión de cambio de nombre preconfiguradas que se aplican a tipos de aplicaciones o tecnologías concretas.

Exclusiones de cambio de nombre



La ficha **Exclusiones de cambio de nombre** le permite determinar qué partes de un programa se van a excluir del proceso de cambio de nombre. Puede excluir elementos específicos del cambio de nombre examinando el árbol de navegación y activando esos elementos.

La ficha *Exclusiones* de cambio de nombre le ofrece un completo control detallado sobre todas las partes de un programa que puede desear excluir del proceso de cambio de nombre.

Puede excluir elementos específicos del cambio de nombre examinando la vista de árbol de la aplicación y activando los elementos que desee excluir. Además, puede crear visualmente sus propias reglas personalizadas para seleccionar varios elementos para la exclusión.

Se pueden definir reglas personalizadas para excluir mayores selecciones de elementos sin necesidad de seleccionarlas individualmente para la exclusión en la vista de árbol. Se pueden definir reglas para excluir espacios de nombres, clases, métodos, propiedades, eventos y capos basándose en su nombre. Además, se puede usar una expresión regular para proporcionar una manera de especificar un modelo con cualesquiera elementos que coincidan con ese modelo, que se excluirán de la ofuscación.

Para ayudar a optimizar las reglas de exclusión, puede obtener una vista previa de los efectos acumulados en cualquier momento haciendo clic en el botón **Vista previa**. La vista del árbol de aplicación atenúa todos los elementos seleccionados para la exclusión.

Para obtener una vista previa de una regla específica, haga clic con el botón secundario en la regla cuyos efectos desee ver, seleccione **Vista previa** y Dotfuscator resaltará los elementos que se vayan a excluir debido a esa regla.

La exclusión de un espacio de nombres excluirá todos los tipos y sus miembros en el espacio de nombres especificado. Para excluir un espacio de nombres, seleccione el botón **Agregar espacio de nombres** y escriba el nombre del espacio de nombres que desee excluir en el campo *Nombre*. Si desea usar una expresión regular que coincida con varios espacios de nombres, asegúrese de que la casilla *Expresión regular* esté activada y de que el campo *Nombre* contiene la expresión regular con la que desee comparar.

Los tipos se pueden excluir mediante solo coincidencias de sus nombres o si se incluye opcionalmente un conjunto de especificadores de atributos que restringirán aún más los tipos excluidos del cambio de nombre. Las clases anidadas pueden excluirse mediante una barra inversa “\” como delimitador entre la clase externa y la interna. Al activar la casilla *Expresión regular*, el valor del campo *Nombre* se comparará como expresión regular, en lugar de usar una cadena literal, cuando se determine qué elementos se excluyen de la ofuscación. La exclusión de tipos puede seguir las reglas de herencia. Si se selecciona la opción *Aplicar a tipos derivados*, la regla de exclusión se aplicará además a los tipos coincidentes y a los tipos que hereden de ellos.

Hay disponibles otras opciones de exclusión, que se definen como reglas secundarias para una regla de tipo contenedor. Para crear una regla secundaria, haga clic con el botón secundario en el tipo de regla que desee mejorar y seleccione la implementación de regla secundaria concreta que desee agregar.

Las reglas para excluir métodos, campos, propiedades y eventos se establecen al crear un tipo de regla para el tipo o los tipos que contienen esos elementos y al agregar una regla secundaria para el elemento adecuado. Al igual que ocurre con los espacios de nombres y los tipos, la exclusión de otros elementos puede basarse en su nombre literal o en una expresión regular.

Los tipos, métodos, campos y propiedades se pueden seleccionar para la exclusión si se anotan con atributos personalizados y, a continuación, se crea una regla de exclusión de tipos que contenga una regla secundaria **CustomAttribute**. La coincidencia con el nombre del CustomAttribute se usa para determinar si se debe excluir un elemento y la lógica de coincidencia con la expresión regular se implementará si está seleccionada la opción *Expresión regular*. La exclusión por atributo personalizado también puede propagarse por la jerarquía de herencia mediante la opción *Permitir herencia*, que excluye adicionalmente cualquier subtipo o reemplazo de métodos y propiedades con el atributo especificado.

Los supertipos también se pueden excluir de la ofuscación si se agrega una regla secundaria de supertipo a una definición de regla de tipo. Un elemento se excluye de la ofuscación si se hereda de uno o más supertipos que coinciden con la condición de exclusión del supertipo.

Para especificar que una regla de exclusión se aplica solo a un método, campo, propiedad, evento, supertipo o atributo personalizado y para incluir el tipo contenedor en la ofuscación, seleccione la opción *Excluir tipo* al definir la regla de tipo. Esta opción deja el tipo incluido para los fines de cambio de nombre y solo evalúa las reglas secundarias.

Reglas integradas de cambio de nombre

Introducción

Exclusiones

Reglas integradas

Opciones

Reglas disponibles:

- Campos con AccessedThroughPropertyAttribute
- Campos de UserControl de Silverlight y WPF
- Tipos con atributos System.Web.Services
- Configuración de aplicación autogenerada
- Tipos y campos marcados como serializables
- Actividades de flujo de trabajo de Windows
- ResourceManager autogenerado
- Métodos con atributos System.Web.Services
- Objetos CrystalReport ReportClass
- Agentes de tareas programadas de Windows Phone

Campos con AccessedThroughPropertyAttribute

Excluye del espacio de nombres System.Runtime.CompilerServices los campos decorados con el atributo AccessedThroughPropertyAttribute. En Visual Basic, cualquier propiedad WithEvents accesible contiene un campo privado con este atributo. El atributo no se quita durante la compilación de modo que otros ensamblados podrían examinarlo mediante reflexión, por lo que esta regla solo se puede usar en modo Biblioteca.

La pestaña **Reglas integradas de cambio de nombre** muestra las reglas de exclusión de cambio de nombre preconfiguradas que se aplican a tipos de aplicaciones o tecnologías específicas. Cada regla tiene una descripción que muestra en la pantalla cuándo se selecciona. Para aplicar una regla integrada a su proyecto, basta con activarla.

Opciones de cambio de nombre

La pestaña **Opciones de cambio de nombre** se usa para establecer las opciones de cambio de nombre e identificar las opciones de salida del archivo de asignaciones. La sección *Opciones de cambio de nombre* contiene las opciones de configuración que controlan cómo se realiza el cambio de nombres.

The screenshot shows the 'Opciones de cambio de nombre' configuration window. On the left, a sidebar contains 'Introducción', 'Exclusiones', 'Reglas integradas', and 'Opciones' (selected). The main area is titled 'Opciones de cambio de nombre' and contains the following settings:

- Habilitar cambio de nombre
- Usar Overload Induction mejorada
- Usar Overload Induction mejorada en tipos serializables
- Asegurar la compatibilidad con el serializador XML
- Introducir invalidaciones de métodos explícitos al cambiar de nombre
- Cambiar nombre de prefijo: [Empty text box]
- Control de espacio de nombres: **Acoplar y cambiar el nombre** (dropdown menu)
- Esquema de cambio de nombre: **Alfabeto en minúsculas** (dropdown menu)

Below this, the 'Ofuscación en aumento' section includes:

- Archivo de asignaciones de entrada: [Empty text box]
- Usar el archivo de asignaciones para cambiar el nombre de las referencias que no se encuentran en los ensamblados de entrada

The 'Mapa de salida' section includes:

- Mostrar archivo de asignaciones: `$(configdir)\Dotfuscated\Map.xml`
- Sobrescribir sin hacer copia de seguridad de la copia existente
- Guardar como HTML
- Usar hoja de estilo personalizada: [Empty text box]

- **Habilitar cambio de nombre** : active esta opción para habilitar el cambio de nombre. También puede hacer clic con el botón secundario en **Cambiar nombre** en el árbol de navegación y activar **Habilitar** para habilitar el cambio de nombre. Igualmente, si el cambio de nombre está habilitado, puede deshabilitarlo haciendo clic con el botón secundario en **Cambiar nombre** en el árbol de navegación y desactivando **Habilitar**. Si esta opción se desactiva, Dotfuscator no realizará el cambio de nombre cuando se compile el proyecto.
- **Asegurar la compatibilidad con el serializador XML**: seleccione esta opción para configurar el cambio de nombre de modo que se permita la serialización XML.
- **Introducir invalidaciones de métodos explícitos al cambiar de nombre** : actívela para permitir invalidar métodos para tener nombres diferentes de aquellos a los que invalidan.

También puede seleccionar las opciones apropiadas de *Control de espacio de nombres*:

- **Acoplar y cambiar el nombre** quita los espacios de nombres.
- **Solo cambiar nombre** cambia el nombre de los espacios de nombres, pero mantiene la jerarquía.
- **Preservar** mantiene los espacios de nombres exactamente como están en el ensamblado de entrada.

La sección *Asignación de resultados* es donde se establece el destino del archivo de asignaciones de salida. También tiene la opción de sobrescribir el archivo de salida cada vez que compile la aplicación sin generar una copia de seguridad de los archivos de asignaciones existentes.

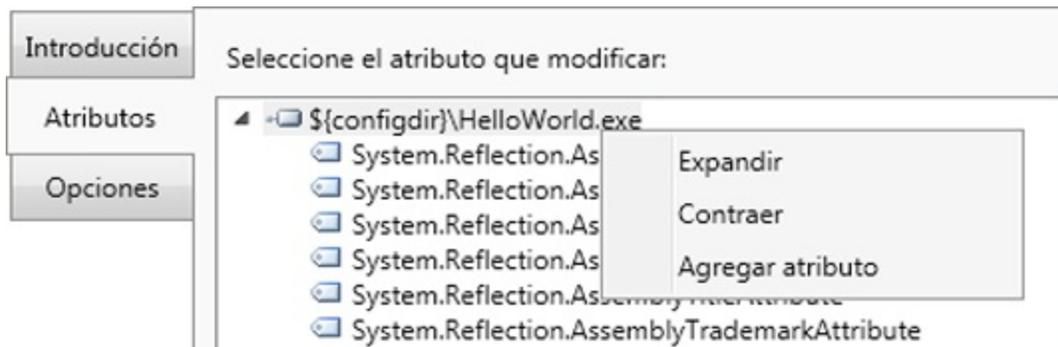
Analytics

Analytics agrega a las aplicaciones funcionalidad previamente compilada para el seguimiento de su uso, notificación de excepciones, detección de la alteración binaria y expiración de las aplicaciones, sin necesidad de código adicional. Dotfuscator and Analytics inyecta el código necesario en su lugar. Puede definir puntos de inyección a través de atributos personalizados en el código fuente o de atributos extendidos especificados dentro de la interfaz de usuario de Dotfuscator and Analytics.

El editor de Analytics permite agregar, modificar y revisar los atributos personalizados y extendidos. En Dotfuscator and Analytics CE, el editor de Analytics muestra tres pestañas de configuración: Introducción, [Atributos](#) y [Opciones](#). En la pestaña Introducción se describen las distintas opciones de Analytics disponibles, así como el modo de configurarlas. La pestaña Atributos es donde se agregan a los métodos los nuevos atributos extendidos o se modifican los atributos personalizados ya existentes. La pestaña [Opciones](#) es donde puede seleccionar qué tipos de mensaje (análisis de aplicación, notificación de vida útil y alerta de alteración) se deben enviar al servicio Runtime Intelligence.

Atributos de ensamblado

Los atributos de ensamblado son aquellos que solo pueden aplicarse en el nivel de ensamblado. Se usan en el proceso de inyección de código para agregar datos de identificación únicos a un ensamblado, como la información de la compañía, o para realizar el seguimiento de las excepciones en todo el ensamblado.



Atributos de ensamblado obligatorios

Para instrumentar los Runtime Intelligence Services, hay dos atributos de ensamblado obligatorios. Estos atributos se agregan a la aplicación haciendo clic con el botón secundario en un nodo de ensamblado en el árbol *Seleccione el atributo que modificar:*.

BusinessAttribute

BusinessAttribute es un atributo necesario que contiene los valores que identifican la compañía propietaria de la aplicación que es objeto de instrumentación. Este atributo está formado por una clave *CompanyKey*, que se debe establecer en "7d2b02e0-064d-49a0-bc1b-4be4381c62d3" para su uso con el portal gratuito de Runtime Intelligence Services. Por comodidad, *CompanyKey* se establece en este GUID de manera predeterminada. Si desea usar como destino un extremo diferente, use el valor de *CompanyKey* que le haya proporcionado PreEmptive Solutions o genere un identificador único nuevo con el botón "..." asociado al área de entrada de *CompanyKey*. *CompanyName* se puede dejar vacío, pero se recomienda especificar el nombre de la compañía para la personalización del portal de Runtime Intelligence Services.

ApplicationAttribute

El atributo ApplicationAttribute es obligatorio y contiene los valores que se usan para identificar la aplicación que se instrumenta:

- **ApplicationType.** Identifica el tipo de aplicación que se instrumenta como una aplicación Windows.
- **GUID.** Es un identificador único de la aplicación que se genera haciendo clic en "..." en este campo.
- **Nombre** Nombre de la aplicación. Si lo deja en blanco, el nombre se rellenará mediante reflexión (si se permite en tiempo de ejecución). Rellénelo solo si desea que el nombre en el portal sea diferente del ensamblado o si no se puede usar la reflexión en la aplicación en tiempo de ejecución (por ejemplo, si se trata de una

aplicación Silverlight).

- **Version.** Es la versión de la aplicación. Si lo deja en blanco, el nombre se rellenará mediante reflexión (si se permite en tiempo de ejecución). Rellénelo solo si desea que la versión en el portal sea diferente de la versión en el ensamblado o si no se puede usar la reflexión en la aplicación en tiempo de ejecución (por ejemplo, si se trata de una aplicación Silverlight).

BinaryAttribute

BinaryAttribute contiene un valor que se usa para especificar los ensamblados que componen la aplicación que se instrumenta. El *GUID* es el identificador único del ensamblado y se usa para especificar qué ensamblado se alteró cuando se detectó la alteración. Aunque este atributo no es obligatorio, si no se usa la detección de alteración, debe establecer un atributo binario en cada ensamblado para que las notificaciones puedan incluir la información de ensamblado.

ExceptionTrackAttribute (nivel de ensamblado)

ExceptionTrackAttribute se usa para realizar el seguimiento de las excepciones no controladas en el nivel de ensamblado y de método. En el nivel de ensamblado, ExceptionTrackAttribute se puede usar para detectar las excepciones que se producen en cualquier parte de un ensamblado. Cuando Dotfuscator and Analytics encuentra ExceptionTrackAttribute, agrega código que detecta las excepciones no controladas mediante el registro de un controlador de evento UnhandledException en el AppDomain actual (para las aplicaciones .NET Framework) o la aplicación actual (para las aplicaciones Silverlight). Se puede establecer una opción de participación explícita en el informe de excepciones, a través de un cuadro de diálogo de excepción integrado o personalizado, que también puede presentar un URI con la directiva de privacidad y recopilar la información de contacto del usuario final o información acerca del contexto en el que se produjo la excepción, tal y como lo ha indicado el usuario final.

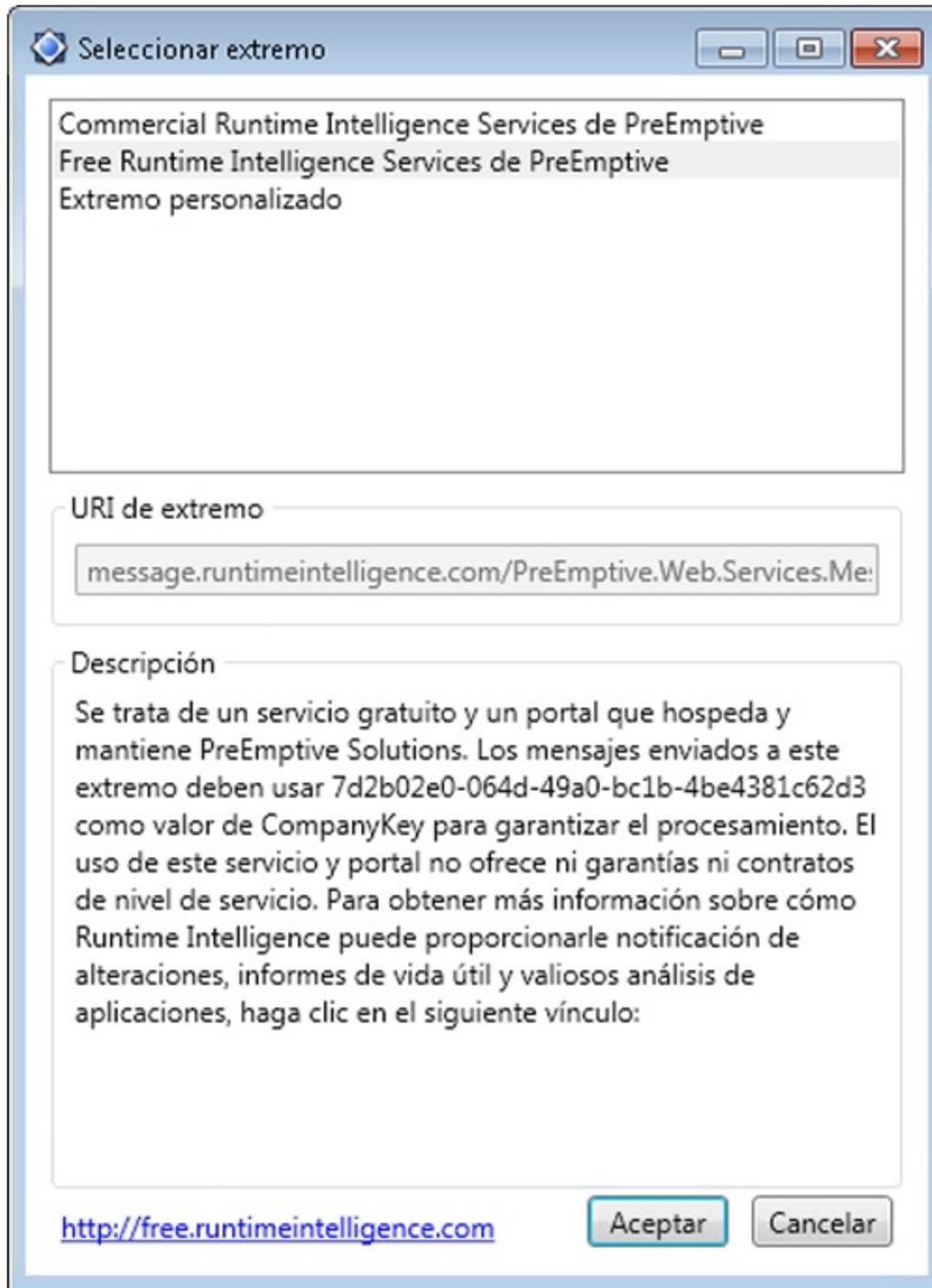
Atributos funcionales

Los atributos funcionales se pueden usar para realizar el seguimiento de la estabilidad de las aplicaciones, el uso de funciones, los intentos de alteración y la frecuencia de uso de una aplicación. También se pueden usar para hacer que una aplicación deje de funcionar después de una fecha especificada.

SetupAttribute

El atributo SetupAttribute se usa para realizar el seguimiento de cuándo se inicia una aplicación. Dotfuscator and Analytics inyecta código de inicio para el envío de mensajes y envía el mensaje de inicio. Debe haber uno o varios métodos con este atributo en un ensamblado o en la aplicación que use Runtime Intelligence. Este atributo se debe colocar lo más cerca posible de la lógica de inicio normal de la aplicación. En tiempo de ejecución, la funcionalidad que genera este atributo comenzará a enviar los datos de uso al portal de Runtime Intelligence Services. Opcionalmente, puede especificar la ubicación en el código de la funcionalidad que controla la transmisión de mensajes (el comportamiento de alta o baja). Dotfuscator and Analytics usará esta información al generar el código de inicialización.

Al agregar el atributo SetupAttribute, puede establecer el destino de **Extremo personalizado** de los mensajes. Entre las opciones de extremo se incluyen el [Portal gratuito de Runtime Intelligence Services](#) de forma predeterminada, el [Portal de Runtime Intelligence Services](#) disponible comercialmente o un extremo personalizado que prefiera.



El atributo SetupAttribute también proporciona la capacidad de implementar la funcionalidad de alta o baja dentro de la aplicación. Esta funcionalidad permite a los usuarios elegir si proporcionan sus datos de uso o no. Para definir el valor que determina si se van a recopilar los datos de uso y ejecución, y se van a enviar al [Portal de Runtime Intelligence Services](#), puede especificar un método, argumento de método, campo o propiedad que contenga o devuelva un valor booleano. Si el valor es **true**, la aplicación transmitirá los datos de uso según se haya configurado. Si el valor es **false**, la aplicación no transmitirá ningún dato de uso.

The screenshot shows the configuration interface for the `PreEmptive.Attributes.SetupAttribute` attribute. The left pane shows a tree view of the application structure, with the `OptIn` property selected under the `HelloWorld.Hello` method. The right pane shows the configuration details for the attribute, including the `OptInSourceElement` property set to `Method`, `OptInSourceName` set to `OptIn`, and `OptInSourceOwner` set to `HelloWorld.Hello`.

Propiedades de atributo:	
CustomEndpoint	message.runtimeintelligence.com/PreEmptive.Wf
ExtendedKeyMethodArguments	
ExtendedKeySourceElement	None
ExtendedKeySourceName	
ExtendedKeySourceOwner	
InstanceIdSourceElement	None
InstanceIdSourceName	
InstanceIdSourceOwner	
OfflineStateSourceElement	None
OfflineStateSourceName	
OfflineStateSourceOwner	
OfflineStorageResultSinkElement	None
OfflineStorageResultSinkName	
OfflineStorageResultSinkOwner	
OptInSourceElement	Method
OptInSourceName	OptIn
OptInSourceOwner	HelloWorld.Hello
UseSSL	False

OptInSourceOwner
Si `OptInSourceElement` es un campo o propiedad, `OptInSourceOwner` indica el nombre del tipo que define el campo o propiedad. Si no se establece, el elemento de origen c...

Para configurar el comportamiento de alta o baja, debe configurar la propiedad **OptInSourceElement** con el tipo de elemento de código apropiado que vaya a contener el valor booleano. La propiedad **OptInSourceName** debe ser el nombre del elemento que contenga el valor booleano que se vaya a usar para determinar si los datos de Runtime Intelligence se van a recopilar y transmitir. En tiempo de ejecución, el valor de origen se debe establecer en el código de la aplicación antes de llamar al método de configuración. La propiedad **OptInSourceOwner** debe dejarse sin valor si el elemento se define en la misma clase que el método con el atributo SetupAttribute. Si la propiedad **OptInSourceElement** es el argumento de un método, debe corresponderse a un parámetro del método que se anote con el atributo SetupAttribute.

TeardownAttribute

El atributo TeardownAttribute se usa para realizar el seguimiento del momento en que una aplicación se detuvo correctamente. Dotfuscator and Analytics inyecta código de desensamblado para el envío de mensajes y envía el mensaje de apagado. Debe haber uno o varios métodos con este atributo en un ensamblado o aplicación que use la analítica de Runtime Intelligence. El método Teardown no tiene que ser necesariamente el último al que se llame, pero sí debería tener la propiedad que se ejecute exactamente una vez cuando la aplicación se cierre, lo más cerca posible del momento en que se salga de ella.

FeatureAttribute

El atributo FeatureAttribute sirve para marcar las características de las que se desea realizar un seguimiento con el servicio de Runtime Intelligence. Este atributo se puede incluir en cualquier método o propiedad. Runtime Intelligence realiza el seguimiento de las funciones con el nombre que se proporciona en esta propiedad. Se puede dar el mismo nombre a varios métodos, con lo que se actualizará el mismo indicador de uso en el portal de Runtime Intelligence Services. Una función puede ser que pueda medirse la duración de la ejecución o un evento atómico. Para realizar simplemente el seguimiento de si una función se ha ejecutado, use el valor 'Tick' en FeatureEventType. Para medir la duración de la ejecución de la función, use dos atributos de función con el

mismo nombre, uno con el valor 'Start' para FeatureEventType y el segundo con el valor 'Stop'. En el informe Feature Scorecard del portal, verá las duraciones mínima, máxima y promedio de la ejecución de cada característica.

InsertShelfLifeAttribute

Agregue un atributo `InsertShelfLifeAttribute` a cualquier propiedad o método en el que le gustaría que se produjera la comprobación de la expiración de una aplicación. En tiempo de ejecución, la aplicación puede enviar un mensaje de notificación de expiración al portal de Runtime Intelligence Services si la aplicación se ejecuta después de la fecha de expiración. Tras la expiración, el comportamiento predeterminado es enviar el mensaje de notificación de expiración; sin embargo, puede establecer en `DefaultAction` la propiedad `ExpirationNotificationSinkElement`, con lo que, si se desea, se enviará un mensaje de expiración al portal de Runtime Intelligence Services y esto hará que la aplicación se cierre inmediatamente. Para que se envíen mensajes de notificación de expiración, la aplicación debe contener métodos marcados con un atributo `SetupAttribute` y `TeardownAttribute`.

Para instrumentar una aplicación con esta característica, se requiere una clave de activación de vida útil (SLAK). Para obtener una SLAK, vaya al sitio de soporte técnico: <http://www.preemptive.com/support/index.html>.

InsertSignofLifeAttribute

El atributo `InsertSignofLifeAttribute` envía un mensaje cada vez que se llama al método etiquetado con este atributo. En tiempo de ejecución, la señal del código de vida envía un mensaje que indica que la aplicación se ha ejecutado. Para usar este atributo, la aplicación debe contener métodos marcados con un atributo `SetupAttribute` y `TeardownAttribute`.

Para instrumentar una aplicación con esta característica, se requiere una clave de activación de vida útil (SLAK). Para obtener una SLAK, vaya al sitio de soporte técnico: <http://www.preemptive.com/support/index.html>.

InsertTamperCheckAttribute

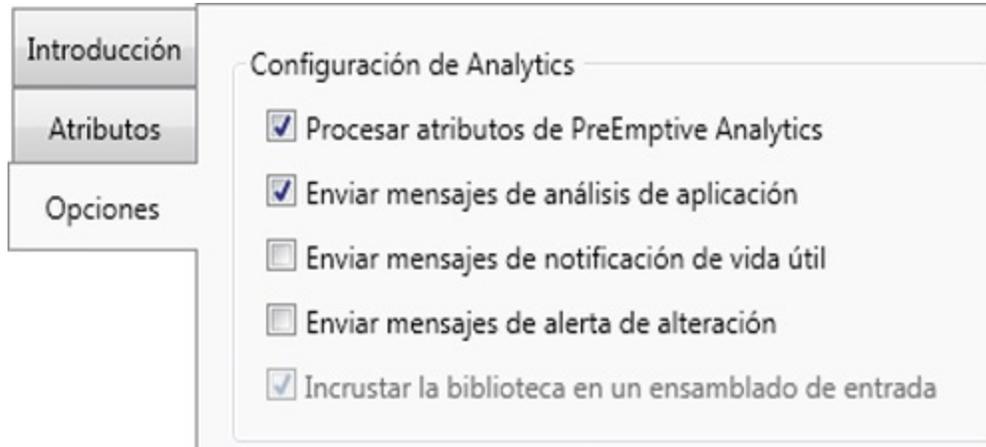
El atributo `InsertTamperCheckAttribute` inyecta código que detecta si un usuario ha modificado los ensamblados. Si detecta que los ensamblados han sido alterados, puede enviar un mensaje de notificación de alteración al portal de Runtime Intelligence Services. Tras la detección, el comportamiento predeterminado es enviar el mensaje de notificación de alteración y cerrar la aplicación; sin embargo, puede establecer en `DefaultAction` la propiedad `ApplicationNotificationSinkElement`, lo que hará que la aplicación se cierre inmediatamente. Si desea enviar mensajes de alteración de Runtime Intelligence, no ponga este atributo en el mismo método que contenga el atributo `SetupAttribute`. Además, asegúrese de que cualquier método marcado con el atributo para comprobar la alteración se ejecute después del método que contenga el atributo `SetupAttribute`.

ExceptionTrackAttribute (nivel de método)

En el nivel de método se puede usar `ExceptionTrackAttribute` para detectar las excepciones no controladas localmente que pasan a través de un método concreto. Cuando Dotfuscator and Analytics encuentra `ExceptionTrackAttribute` en el nivel de método, agrega código que incluye el método en un bloque `try/catch` y vuelve a iniciar la excepción. Los atributos `ExceptionTrackAttribute` en el nivel de método pueden recopilar la misma información adicional y de participación que en el nivel de ensamblado.

Opciones de Analytics

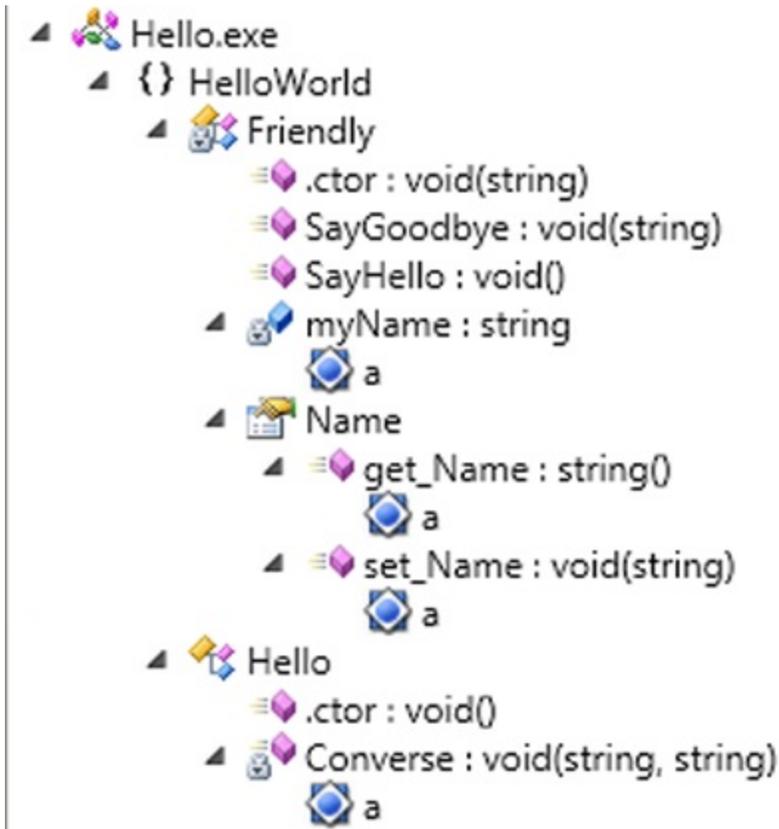
La pestaña **Opciones de Analytics** es el lugar donde se puede configurar el comportamiento de las aplicaciones instrumentadas. La instrumentación (inyección de código) de las aplicaciones puede habilitarse o deshabilitarse para el proyecto actual desactivando la casilla *Procesar atributos de PreEmptive Analytics* o haciendo clic con el botón secundario en Analytics en el árbol de navegación y desactivando la opción **Habilitar**. Si se desactiva esta opción, Dotfuscator no llevará a cabo la inyección de código cuando se compile el proyecto.



La sección *Configuración de Runtime Intelligence* de esta pestaña es donde puede seleccionar qué tipos de mensaje se enviarán al [portal de Runtime Intelligence Services](#). La opción *Enviar mensajes de análisis de aplicación* habilita el envío de mensajes relacionados con el uso de las características. La opción *Enviar mensajes de notificación de vida útil* habilita el envío de mensajes de expiración de vida útil y de señal de vida. La opción *Enviar mensajes de alerta de alteración* habilita el envío de mensajes de notificación de alteración. Desactivar las opciones para enviar mensajes de vida útil o alteración solo afectará a la transmisión de dichos mensajes; si ha seleccionado que la acción predeterminada tenga lugar, la aplicación seguirá cerrándose como estaba previsto.

Resultados

Una vez compilado el proyecto, puede inspeccionar los resultados en la pestaña *Resultados*.



Aquí puede examinar la vista de árbol y ver cómo Dotfuscator and Analytics cambia el nombre de los tipos, métodos y campos. Los nuevos nombres aparecen como nodos secundarios debajo de los nodos originales en el árbol. Esta información también se guarda en el archivo de asignaciones tal como se configura en las [Opciones de cambio de nombre](#).

Más allá de la ofuscación

Dotfuscator and Analytics CE combina su tecnología de ofuscación de confianza con la tecnología de inyección para proporcionar vistas casi en tiempo real de la implementación, estabilidad y uso de las aplicaciones. Además de la ofuscación, Dotfuscator and Analytics CE ofrece una nueva categoría de control de detección para supervisar, administrar y proteger mejor las aplicaciones. Dotfuscator and Analytics CE puede:

- Proporcionar una vista casi en tiempo real de la integridad y la actividad de las aplicaciones en tiempo real.
- Instrumentar las aplicaciones para detectar si han sido alteradas y, si fuera así, enviar un mensaje, si se desea, al servicio Runtime Intelligence.
- Instrumentar las aplicaciones con lógica de expiración, desactivación y notificación que reaccione ante la expiración de las mismas saliendo de ellas y/o enviando un mensaje de Runtime Intelligence Services.

En esta sección

[Introducción a la defensa activa contra la alteración](#)

[Expiración de las aplicaciones](#)

[Análisis de las aplicaciones](#)

[Seguimiento de excepciones](#)

Análisis de las aplicaciones

Runtime Intelligence es un servicio y tecnología que aporta a los creadores y usuarios de aplicaciones una perspectiva sobre cómo se usan estas. Dotfuscator and Analytics se puede emplear con las aplicaciones y componentes .NET habilitados para Runtime Intelligence.

Dotfuscator and Analytics se usa para instrumentar una aplicación de forma que se envíe un mensaje cuando esta se inicie y se detenga, y cuando se usen las características designadas. El servicio Runtime Intelligence agrega estos datos del ciclo de vida de la aplicación y los expone a través del portal de Runtime Intelligence.

En esta sección

[Tipos de aplicaciones .NET admitidas](#)

Tipos de aplicaciones .NET admitidas

Dotfuscator CE puede realizar el procesamiento de Runtime Intelligence de todos los ensamblados .NET excepto los siguientes:

- Ensamblados de entrada de C++ administrado que contiene código nativo y administrado.
- Ensamblados de entrada de varios módulos.
- Ensamblados de entrada de .NET 1.0

Seguimiento de excepciones

El *seguimiento de excepciones* es un método que detecta las excepciones en la aplicación de destino y reacciona automáticamente, a medida que se producen. Dotfuscator and Analytics inyecta código que encuentra las excepciones detectadas, iniciadas o no controladas. A continuación, el código de seguimiento de excepciones puede recopilar detalles del usuario y notificar la excepción detectada a un extremo de Runtime Intelligence. Un usuario puede permitir que se envíe un informe de excepciones de forma explícita, aunque haya rechazado previamente el envío de mensajes de Runtime Intelligence, y puede proporcionar comentarios e información de contacto para que se envíen con el informe. Además, el desarrollador puede especificar la acción personalizada que se debe llevar a cabo cuando se detecte la excepción.

Para facilitar el caso habitual de la notificación de excepciones no controladas, Dotfuscator and Analytics puede inyectar un cuadro de diálogo de informe de excepciones, creado previamente, que proporcione al usuario una experiencia uniforme en los informes de excepciones. Se requiere una configuración mínima para indicar a Dotfuscator and Analytics que realice el seguimiento de las excepciones no controladas, muestre el cuadro de diálogo de informe de excepciones que obtendrá el consentimiento explícito del usuario, recopile comentarios y datos de contacto opcionales del usuario y envíe el informe al extremo de Runtime Intelligence configurado.

En esta sección

[Informe de excepciones y el atributo ExceptionTrack](#)

[Recopilar la información del informe de excepciones especificada por el usuario](#)

Informe de excepciones y el atributo ExceptionTrack

Informe de excepciones

Una vez detectada una excepción, se puede notificar al extremo de Runtime Intelligence configurado si se establece la propiedad **SendReport** del atributo ExceptionTrack en true (el valor predeterminado). Para el envío de informes de excepciones, se respetará la configuración de participación del usuario, si **OptInSource** se ha configurado. Dotfuscator and Analytics se puede configurar para obtener el consentimiento explícito del usuario para enviar el mensaje de informe de excepciones. En este caso, el consentimiento explícito del usuario invalidará la configuración de participación de Runtime Intelligence, si existe. Para obtener el consentimiento explícito del usuario para enviar el mensaje de informe de excepciones, especifique **ReportInfoSource**.

Dotfuscator and Analytics también se puede configurar para obtener información del usuario tal como una descripción de las acciones que condujeron a la excepción y una dirección de contacto que el desarrollador puede usar para solicitar más información o para notificar algún problema que se ha corregido. Esta información se adjuntará al mensaje de informe de excepciones. Para obtener este tipo de información proporcionada por el usuario, especifique **ReportInfoSource**.

La configuración del origen de los datos del informe es opcional. Si se omiten, no se recopilará la información proporcionada por el usuario y el envío de los mensajes de informe de excepciones se controlará mediante la configuración de participación de Runtime Intelligence.

Recopilar la información del informe de excepciones especificada por el usuario

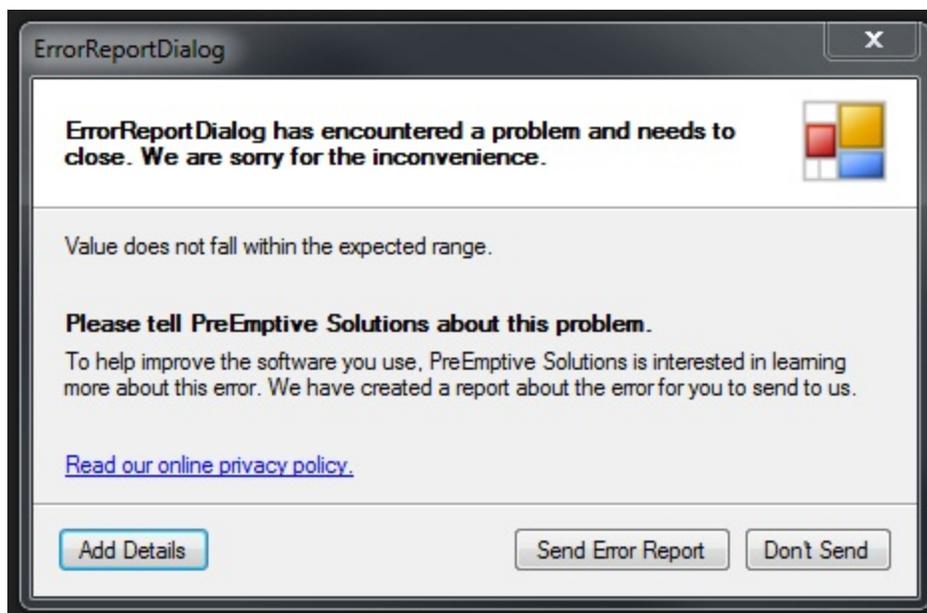
Cuando se usa el seguimiento de excepciones, Dotfuscator and Analytics se puede configurar para obtener el consentimiento explícito del usuario para recopilar sus comentarios y datos de contacto. Se proporcionan como pares clave-valor que se leen en tiempo de ejecución durante la creación de un mensaje de informe de excepciones.

Para proporcionar esta información del informe especificada por el usuario, especifique **ReportInfoSource** en el atributo que corresponde al mensaje que desea enviar. Dotfuscator and Analytics usa **ReportInfoSource** para generar el código que recopila los pares clave-valor en tiempo de ejecución. **ReportInfoSource** es una propiedad, método o campo con valor de **IDictionary** o **IDictionary<string, string>** o, cuando se usa el seguimiento de excepciones de nivel de método, un argumento de método; es responsabilidad del desarrollador asegurarse de que los valores correctos estén disponibles en **ReportInfoSource** en el momento en que se detecta una excepción.

Usar el cuadro de diálogo de informe de excepciones integrado como ReportInfoSource

Dotfuscator and Analytics puede inyectar un cuadro de diálogo de informe de excepciones, creado previamente, para facilitar la configuración en la mayoría de los escenarios y proporcionar al usuario una experiencia coherente para la notificación de excepciones. Para usar el cuadro de diálogo integrado, el ensamblado debe tener como destino la versión 1.1 (o posterior) de .NET Framework o Silverlight versión 2 (o posterior). Para indicar a Dotfuscator and Analytics que use el cuadro de diálogo integrado cuando **ReportInfoSource** establezca el valor de **ReportInfoSourceElement** en "DefaultAction".

Cuando el cuadro de diálogo integrado se usa en .NET Framework, se crea y muestra mediante la API de Windows Forms. Esto puede tener consecuencias imprevistas para las aplicaciones de consola o servicios, por lo que sería preferible usar un **ReportInfoSource** en estos casos. Si su ensamblado todavía no hace referencia a las bibliotecas correctas de Windows Forms, se agregarán referencias.



Usar un ReportInfoSource personalizado

El **ExceptionTrackAttribute** define tres propiedades para especificar un **ReportInfoSource**:

- **ReportInfoSourceElement**. El **ReportInfoSourceElement** puede ser cualquiera de los valores definidos en la enumeración **SourceElements**: un campo, propiedad o método, o cuando se usa el seguimiento de excepciones de nivel de método, un argumento de método. Si **ReportInfoSourceElement** es un argumento de método, debe corresponderse con un parámetro del método al que se adjunta el atributo.

- **ReportInfoSourceOwner.** Si **ReportInfoSourceElement** es un campo, método o propiedad, o clase **ReportInfoSourceOwner** debe indicar la clase que define el campo, método o propiedad.
- **ReportInfoSourceName.** El **ReportInfoSourceName** se debe establecer en el nombre del campo, método, propiedad o argumento de método, de tipo **IDictionary** o **IDictionary<string, string>** que contiene la información del informe especificada por el usuario en tiempo de ejecución.

Hay tres pares clave-valor que se pueden incluir en el diccionario proporcionado por **ReportInfoSource**:

- **consent.** Es una representación de cadena de un valor booleano que indica si el usuario ha aceptado o rechazado explícitamente el envío del mensaje de informe de excepciones actual. Este consentimiento es independiente de la configuración de participación global de Runtime Intelligence, y la invalida.
- **comment.** Es un comentario personalizado que proporciona el usuario opcionalmente. Se puede usar para solicitar información del usuario, como qué hacía cuando se produjo la excepción.
- **contact.** Es un punto de contacto que proporciona el usuario opcionalmente. Su contenido carece de estructura y puede contener, por ejemplo, una dirección de correo electrónico, un número de teléfono o el nombre de usuario de un sitio web de redes sociales. El cuadro de diálogo integrado solicita al usuario que lo proporcione como una dirección de correo electrónico.

Los pares clave-valor que sean distintos de los descritos anteriormente no se tendrán en cuenta.

La recopilación de datos del informe especificados por el usuario es opcional. Si el diccionario recuperado es null, no contiene una clave **consent** o el valor de la clave **consent** es null o no se analiza como booleano, se respeta la configuración de participación global de Runtime Intelligence. Si se omite la clave **comment** o **contact**, el mensaje resultante de Runtime Intelligence no incluye esta información.

Ejemplo **ExceptionTrack** uso de atributo con **ReportInfoSource** definido como un método denominado **"GetDictionary"**:

Uso de atributos de seguimiento de excepciones con ReportInfoSource

```
[ExceptionTrack(
    ReportInfoSourceElement = SourceElements.Method,
    ReportInfoSourceName = "GetDictionary"
)]
private void Foo() {
    ...
}

// Crea y rellena un diccionario con los datos del informe especificados por el
// usuario
public IDictionary<string, string> GetDictionary() {
    Dictionary<string, string> dict = new Dictionary<string, string>();
    dict.Add("consentimiento", "true");
    dict.Add("comentario", "El método Foo() inició una excepción.");
    dict.Add("contacto", "foo@bar.com");
    return dict;
}
```

Introducción a la defensa activa contra la alteración

Dotfuscator and Analytics CE proporciona un medio para que las aplicaciones detecten y, si lo desea, le notifiquen si han sido alteradas desde que se instrumentaron con la notificación de alteración.

Para detectar alteraciones, coloque atributos **InsertTamperCheck** en uno o varios métodos de la aplicación que siempre se ejecuten. Cuando Dotfuscator and Analytics encuentra un atributo **InsertTamperCheck** durante su procesamiento, agrega código que realiza una comprobación de integridad de nivel de ensamblado en tiempo de ejecución. Si la comprobación de integridad no tiene éxito, envía un mensaje de alteración detectada al servicio de Runtime Intelligence. También puede llamar a código que cierre la aplicación. **InsertTamperCheck** no es necesario en tiempo de ejecución; por lo tanto Dotfuscator and Analytics los quita de la aplicación de salida.

Una aplicación puede contener cualquier cantidad de atributos **InsertTamperCheck**. En el caso de que una aplicación haya sido alterada, se enviarán varios mensajes de alteración detectada de la misma sesión de aplicación con el mismo identificador de grupo.

No coloque este atributo en el mismo método que contiene el atributo SetupAttribute. Los métodos con este atributo deben ejecutarse después del método que contiene el atributo SetupAttribute.

En Dotfuscator and Analytics CE, los atributos **InsertTamperCheck** solo se pueden colocar en ensamblados que sean archivos EXE. Dotfuscator and Analytics CE no admite la detección de alteraciones en ensamblados de biblioteca (DLL).

En esta sección

[Simular la alteración](#)

[Tipos de aplicaciones .NET admitidas](#)

Simular la alteración

Dotfuscator incluye una utilidad de línea de comandos sencilla que 'altera' un ensamblado. Se denomina **TamperTester.exe** y se instala en la misma carpeta que Dotfuscator.

Uso

```
tampertester <file_name> [destination folder]
```

Al ejecutar los ensamblados de Dotfuscator and Analytics con esta utilidad, puede probar que los mensajes de notificación de alteración están generándose y enviándose según lo previsto. También puede probar el código de aplicación que ha escrito para ejecutarse en respuesta a la detección de una alteración.

Tipos de aplicaciones .NET admitidas

Dotfuscator and Analytics CE puede realizar el procesamiento de las notificaciones de alteración de todos los ensamblados .NET excepto los siguientes:

- Ensamblados de entrada de C++ administrado que contiene código nativo y administrado.
- Ensamblados de entrada de varios módulos.
- Ensamblados de entrada de .NET 1.0
- Ensamblados de .NET Compact Framework.
- Ensamblados de Silverlight.
- Ensamblados de biblioteca.

Expiración de las aplicaciones

Vida útil es una función de administración del inventario de las aplicaciones que permite incrustar lógica de expiración, desactivación y notificación en una aplicación a través de la inyección de código. Dotfuscator and Analytics inyecta código que reacciona ante la expiración de las aplicaciones saliendo de ellas y/o enviando un mensaje de Runtime Intelligence. Esta funcionalidad resulta útil en el caso de las aplicaciones beta, ya que les permite autorregularse y aplica directivas de expiración y caducidad. Puede programar la expiración o desactivación de una aplicación para una fecha concreta o un número de días después de la instrumentación de la misma.

En esta sección

[Introducción a las claves de activación de vida útil](#)

Introducción a las claves de activación de vida útil

Una clave de activación de vida útil (**SLAK, Shelf Life Activation Key**) es un archivo de datos necesario para inyectar la funcionalidad de Shelf Life en las ubicaciones apropiadas dentro de una aplicación.

PreEmptive emite una clave de activación de vida útil que el usuario proporciona a Dotfuscator and Analytics durante la configuración de la vida útil. Para obtener una clave de activación de vida útil, póngase en contacto con PreEmptive Solutions. PreEmptive le emitirá un archivo de datos que contenga la clave que se va a almacenar en el equipo de compilación.

Una vez obtenida la clave, puede agregar el atributo Shelf Life a un método o grupo de métodos. En la pestaña *Análisis* de la sección *Editor de atributos*, en el campo *ActivationKeyFile*, debe seleccionar la ruta de acceso al archivo de claves de activación de vida útil, con lo que se activa la vida útil en su aplicación.

Portal gratuito de Runtime Intelligence Services

Una vez que una aplicación se ha instrumentado, comenzará a enviar datos de uso al [portal gratuito de Runtime Intelligence Services](#), donde puede ver diversos informes generados con los datos acumulados. El portal gratuito de Runtime Intelligence Services permite el almacenamiento limitado de los datos de uso y no tiene un contrato de nivel de servicio mínimo ni requiere la autenticación del usuario; por lo tanto, sus datos se consideran públicos. Para que el almacenamiento de los datos, vistas de detalle y resumen adicionales de los datos sean permanentes y los SLA estén garantizados, póngase en contacto con nosotros con el fin de obtener una [evaluación](#) de nuestra versión disponible comercialmente.

El portal gratuito de Runtime Intelligence Services proporciona los informes siguientes:

- **Application Overview:** proporciona una representación visual de alto nivel de los sistemas operativos, entornos, marcos de trabajo y estabilidad de las aplicaciones.
- **Application Scorecard:** proporciona una representación completa en forma de texto del uso de las aplicaciones estructurada según la información de la versión incrustada en las aplicaciones.
- **Feature Overview:** proporciona una representación visual de los datos de uso de las funciones recopilados para las aplicaciones.
- **Feature Scorecard:** proporciona una representación detallada en forma de texto de los datos de uso de las funciones recopilados para las aplicaciones.

Para empezar a ver el modo en que los usuarios usan sus aplicaciones, vaya a <http://free.runtimeintelligence.com> y escriba el valor de [CompanyKey](#) que usó al instrumentar la aplicación como nombre de usuario.

Se dispone de información más detallada acerca de los informes en los temas de Ayuda que se encuentran en el portal de Runtime Intelligence Services.

Indice

.exe, 9-11

.NET, 38 , 31

compact framework, 38

.xaml, 9-11

acción de función, 12-15

Actualizaciones, 4

alteración, 21-27 , 29 , 36 , 37

defensa, 2 , 3 , 36

detección, 36

notificación, 36

simulación, 37

Análisis de las aplicaciones, 30

analytics, 16 , 21-27 , 40 , 30

configuración, 16

opciones, 21-27

editor, 21-27

pestaña, 40

aplicación

seguridad, 2

expiración, 2 , 3 , 29 , 39

excepciones, 2 , 3 , 4 , 21-27 , 32 , 33 , 34-35

protección, 3 , 4

defensa, 3 , 4

medida, 3 , 4

administración, 3 , 4

atributo, 21-27

GUID, 21-27

nombre, 21-27

tipo, 21-27

versión, 21-27

análisis, 29 , 30

desactivación, 39

informe de introducción, 41

informe scorecard, 41

application, 21-27 , 29 , 30

archivo

nombre, 9-11

ruta de acceso, 9-11

tamaño, 9-11

asignación de resultados, 17-20

assembly, 9-11

atributo, 12-15 , 21-27

ofuscación, 12-15

personalizado, 21-27

- extendido , 21-27
- ensamblado, 21-27
- obligatorio, 21-27
- empresa, 21-27
- binario, 21-27
- funcional, 21-27
- setup, 21-27
- teardown, 21-27
- feature, 21-27
- insertar vida útil, 21-27
- insertar señal de vida, 21-27
- insertar comprobación de alteración, 21-27
- atributo de ofuscación, 12-15**
- atributo insert tamper check, 36**
- atributo personalizado, 17-20**
- baml, 9-11**
- código administrado, 38 , 31**
- código nativo, 38 , 31**
- C++, 38 , 31**
- cadena de función, 12-15**
- cadena de asignaciones de función, 12-15**
- cambiar el nombre, 17-20**
- cambio de nombre**
 - exclusiones, 17-20
 - opciones, 17-20
 - reglas integradas, 17-20
 - introducir invalidaciones de métodos explícitos, 17-20
- característica**
 - informe overview, 41
- cifrado de cadenas, 16**
- CompanyKey, 41**
- compilación, 12-15**
 - configuración, 12-15
 - detalle de salida, 12-15
 - eventos, 12-15
- configuración, 1 , 16**
 - opciones, 16
- configuración de ofuscación global, 12-15**
- contenido dinámico, 5-8**
- cuadro de diálogo de informe de excepciones, 34-35**
- cumplir atributos de instrumentación, 9-11**
- cumplir atributos de ofuscación, 9-11**
- defensa activa contra la alteración, 29**
- desactivación, 29**
- directorío, 12-15**
 - destino, 12-15

- temporal, 12-15
- dll, 9-11**
- Dotfuscator, 1, 5-8**
- eliminación, 16**
- ensamblado**
 - propiedad, 9-11
 - agregar, 9-11
 - información, 9-11
 - entrada, 9-11
- ensamblado de entrada**
 - varios módulos, 38, 31
- entradas, 5-8**
- espacio de nombres, 17-20**
 - opciones de control, 17-20
- exclusiones de cambio de nombre, 16**
- expiration, 29**
- extremo personalizado, 21-27**
- fecha de modificación, 9-11**
- file, 9-11**
- firma, 12-15**
- función, 41**
 - seguimiento del uso, 2, 3
 - informe scorecard, 41
- Funciones, 3**
- Informe de excepciones y el atributo ExceptionTrack, 33**
- Introducción, 2**
- Introducción**
 - Introducción, 2
 - Funciones, 3
 - Actualizaciones, 4
- Introducción a la defensa activa contra la alteración, 36**
- Introducción a las claves de activación de vida útil, 40**
- Más allá de la ofuscación, 29**
- Más allá de la ofuscación**
 - Más allá de la ofuscación, 29
 - Introducción a la defensa activa contra la alteración, 36
 - Simular la alteración, 37
 - Tipos de aplicaciones .NET admitidas, 38, 31
 - Expiración de las aplicaciones, 39
 - Introducción a las claves de activación de vida útil, 40
 - Análisis de las aplicaciones, 30
 - Seguimiento de excepciones, 32
 - Informe de excepciones y el atributo ExceptionTrack, 33
 - Recopilar la información del informe de excepciones especificada por el usuario, 34-35
- marcado previo, 16**
- mensaje, 21-27**

- analítica de aplicación, 21-27
- vida útil, 21-27
- notificación de alteración, 21-27
- señal de vida, 21-27
- modo de biblioteca, 9-11**
- notificación, 29**
 - lógica, 29
- notificación de alteración, 38**
- ofuscación, 9-11 , 12-15**
- ofuscación declarativa, 12-15**
- ofuscación del flujo de control, 16**
- ofuscación inteligente, 12-15**
 - informe, 12-15
- opciones de cambio de nombre, 16**
- Opciones de configuración, 16**
- Opciones de configuración**
 - Opciones de configuración, 16
 - Cambiar el nombre, 17-20
 - Analytics, 21-27
- opciones de exclusión, 17-20**
- package, 9-11**
- paquete**
 - agregar, 9-11
 - información, 9-11
 - entrada, 9-11
 - propiedad, 9-11
- Portada, -2--1**
- Portada**
 - Portada, -2--1
- Portal**
 - Portal gratuito de Runtime Intelligence Services, 41
- Portal gratuito de Runtime Intelligence Services, 41**
- PreEmptive Dotfuscator and Analytics CE, 1**
- PreEmptive Dotfuscator and Analytics CE**
 - PreEmptive Dotfuscator and Analytics CE, 1
- preferencias de usuario, 5-8**
 - configuración, 5-8
- Primeros pasos, 5-8**
- Primeros pasos**
 - Primeros pasos, 5-8
 - Entradas, 9-11
 - Propiedades, 12-15
- propiedades, 5-8 , 12-15**
 - sintaxis de referencia, 12-15
 - valor, 12-15
- propiedades del proyecto, 12-15**

- propiedades externas, 12-15
- quitar atributos de ofuscación, 9-11
- Recopilar la información del informe de excepciones especificada por el usuario, 34-35
- reflexión, 12-15
- registrarse ahora, 5-8
- registro, 5-8
 - estado, 5-8
- reglas integradas, 16
 - cambio de nombre, 16 , 17-20
- reglas personalizadas, 17-20
- resultados, 1 , 28
 - ficha, 28
- Resultados**
 - Resultados, 28
- Runtime Intelligence Services, 1 , 2 , 29 , 30 , 41**
 - Portal, 1 , 2 , 21-27
 - Portal gratuito, 1 , 2 , 21-27 , 41
 - extremo personalizado, 21-27
 - configuración, 21-27
 - informe Application Overview, 41
 - informe Application Scorecard, 41
 - informe Feature Overview, 41
 - informe Feature Scorecard, 41
- ruta de acceso de carga de ensamblado, 12-15
- señal de vida, 21-27
- seguimiento de excepciones, 2 , 3 , 21-27 , 32 , 33 , 34-35
- seguimiento de la sesión, 2 , 3
- serializador XML, 17-20
- Silverlight, 38
- Simular la alteración, 37
- SLAK, 39 , 40
- Software + Servicios, 2
- supertipos, 17-20
- TamperTester.exe, 37
- tipo, 17-20
- tipo de aplicación, 38
 - .NET admitida, 38
- Tipos de aplicaciones .NET admitidas, 38 , 31
- transformación xaml, 9-11
- version, 5-8 , 9-11 , 12-15
- vida útil, 21-27 , 39 , 40
 - clave de activación, 39 , 40
 - configuración, 40
 - atributo, 40
- vinculación, 16
- vista previa, 17-20

Visual Studio, 5-8