3.4.-- Crear una base de datos IDA

Una vez has elegido un archivo a analizar y especificado tus opciones, IDA inicia la creación de una base de datos. En este proceso, IDA toma el control del módulo del cargador seleccionado, cuyo trabajo es cargar el archivo del disco, analizar cualquier información de encabezados del archivo y reconocerlos, crear varias secciones del programa las cuales contendrán código o datos especificados en los encabezados y finalmente identificar específicos puntos de entrada en el código y después retornar el control a IDA. En este sentido, los módulos de carga se comportan como cargadores de un sistema operativo. El cargador determinará un esquema de memoria virtual basado en la información contenida en los encabezados del programa y configurará la base de datos de acuerdo a ella.

Una vez haya finalizado el cargador, la máquina de desensamblado de IDA se inicia y empieza a pasar una dirección cada vez al módulo de procesador escogido. El trabajo del módulo de procesador es determinar el tipo de instrucción localizada en la dirección, la longitud de la instrucción en dicha dirección y la ubicación o ubicaciones en las cuales continua la ejecución del programa desde dicha dirección, ejemplo: puede ser una instrucción secuencial o de desviación. Cuando IDA ha encontrado todas las instrucciones del ejecutable sin ningún problema, realiza una segunda pasada a través de la lista de direcciones de instrucciones y le pide al módulo de procesador que genere la versión de cada instrucción en lenguaje ensamblador para mostrarla.

Seguidamente a este desensamblado, IDA automáticamente realiza un análisis adicional al archivo binario para extraer más información que será utilizada por el analizador. Los usuarios pueden hallar toda la información siguiente, incorporada en la base de datos, una vez que IDA haya completado su análisis inicial:

3.4.1.--Identificación del compilador

Es utilizado para conocer qué compilador se ha utilizado para construir una parte del software. La identificación del compilador se puede utilizar para ayudarnos a comprender las convenciones utilizadas de las llamadas a funciones utilizadas en el binario para poder determinar qué librerías debe enlazar el binario. Cuando un archivo se carga, IDA intenta identificar el compilador que se ha utilizado para crear el archivo de entrada. Si se puede identificar el compilador, el archivo de entrada es escaneado por secuencias de código conocidas que utiliza dicho compilador. Dichas funciones son codificadas por colores en un esfuerzo de reducir la cantidad de código que necesita ser analizado.

3.4.2.--Identificación de argumentos de función y variables locales

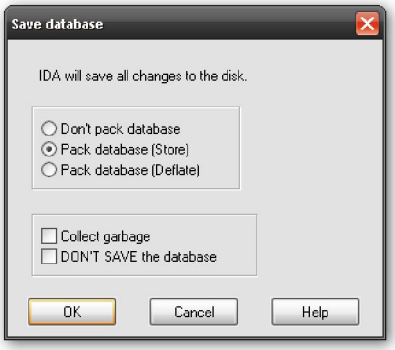
Dentro de cada función identificada, direcciones que son el objetivo de instrucciones **call**, IDA ejecuta un análisis detallado del comportamiento del registro **stack pointer** a fin de reconocer los accesos a las variables localizadas en el **stack** y comprender el esquema del **stack frame**, lo estudiaremos también, de la función. Los **Names** son generados automáticamente para cada variable, basado en su utilización, así como las variables locales dentro de la función o como argumentos pasados a la función como parte del proceso de llamada a dicha función.

3.4.3.--Información del tipo de datos

Utilizando el conocimiento de las funciones de las librerías comunes y sus parámetros requeridos, IDA añade comentarios en la base de datos indicando las ubicaciones de los parámetros que serán pasados a las funciones. Estos comentarios ahorran al analizador una cantidad tremenda de tiempo utilizado proporcionando información, que de otra forma se necesitaría guardar, de referencias de distintas APÏ.

3.5.--Cerrar las bases de datos de IDA

Cada vez que se cierre una base de datos, si cierras la aplicación IDA o simplemente cambias a una base de datos distinta, se te mostrará el siguiente diálogo **Save database**.



Si es el primer guardado de la base de datos creada, el nombre de la base de datos es derivado del nombre del archivo entrado reemplazando la extensión de entrada por la extensión **.idb**, por ejemplo **CRACKME.EXE** producirá una base de datos **CRACKME.IDB**. Cuando el archivo entrado no tiene ninguna extensión, en su base de datos se le añade también **.idb** para formar el nombre de dicha base, por ejemplo para **httpd** será **httpd.idb**. Las opciones disponibles de guardado y sus implicaciones asociadas se resumen en la siguiente lista:

Don’t pack database
Esta opción simplemente copia los cambios a los archivos de las cuatro bases de datos y cierra el área de trabajo sin crear un archivo **IDB**. Esta opción no es recomendada para cerrar tus bases de datos.

Pack database (Store)
Seleccionarla da como resultado un solo archivo **IDB** en el cual se encuentran archivadas las cuatro bases de datos.

Pack database (Deflate)
Esta opción es igual a la anterior, con la excepción de que los componentes de las bases de datos son comprimidos en el archivo **IDB**.

Collect garbage
Recolectar basura, produce en IDA eliminar de la base de datos cualquier página de memoria no utilizada antes de cerrarla. Seleccionar esta opción conjuntamente con **Deflate** da como resultado la creación del archivo **IDB** más pequeño posible. Esta opción no se utiliza a no ser que el espacio del disco sea muy pequeño.

DON'T SAVE the database
Puedes preguntarte porque escogerías no guardar tu trabajo realizado. Esta opción es la forma para desechar los cambios realizados en una base de datos desde la última vez que se guardó. Cuando se elige esta opción, IDA simplemente borra los cuatro archivos de base de datos y deja intactos los que están en el archivo IDB. Esta opción es la más parecida a la acción **Undo**.

3.6.--Reabrir una base de datos

En circunstancias normales, volver a trabajar con una base de datos existente es tan simple como seleccionar la base de datos utilizando uno de los métodos para abrir un archivo en IDA. Dichos archivos se abren rápidamente ya que no tienen que realizar ningún análisis para ejecutarse. Como característica añadida, IDA restaura el área de trabajo en el mismo estado en que fue cerrado.

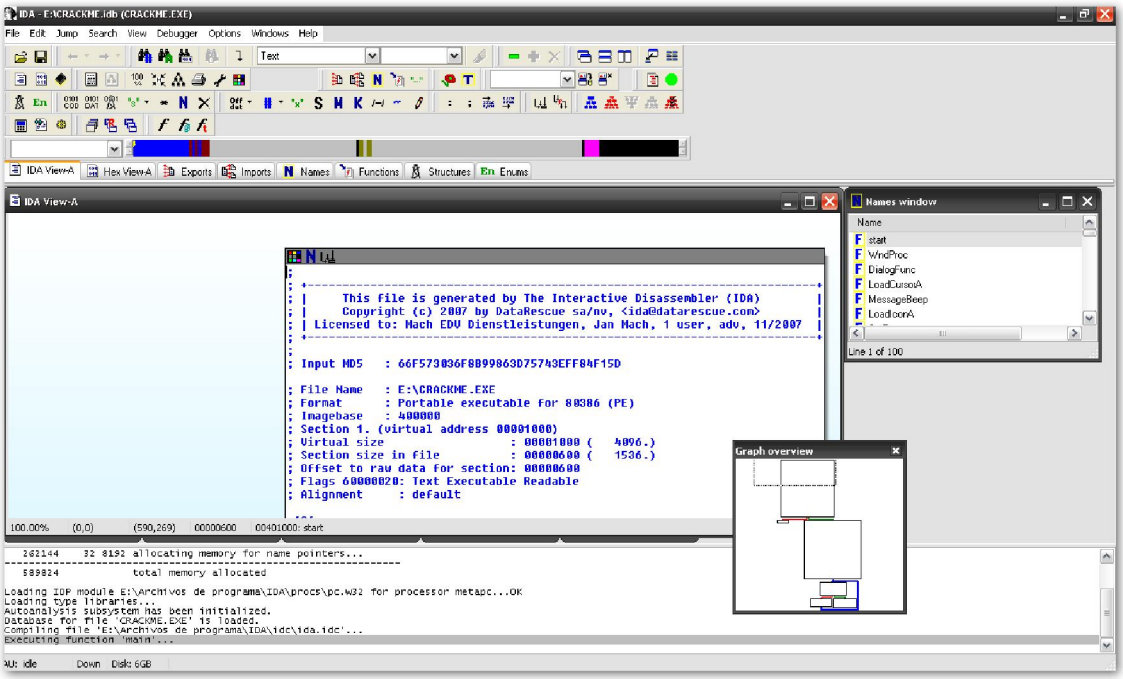
Ahora un punto negativo. Lo creas o no IDA falla de vez en cuando. La causa puede ser un error en el mismo IDA o en algún plugin instalado, estos fallos dejan las bases de datos abiertas potencialmente corrompidas. Cuando reiniciamos, IDA intentará reabrir la base de datos afectada, probablemente mostrará uno de los diálogos siguientes.

Cuando IDA falla, no tiene la oportunidad de cerrar la base de datos activa, y los archivos intermedios de la base de datos no pueden ser borrados. Si no es la primera vez que trabajas con la base de datos, se te presentarán dos archivos uno potencialmente corrupto. El archivo **IDB** representa el último guardado correcto de la base de datos, mientras que los archivos intermedios contienen los cambios realizados en la última actuación. En este caso se te proporciona la opción de revertir a la versión guardada o resumir la potencialmente corrupta. Elegir **Continue with unpacked Base** no garantiza que puedes recuperar el trabajo. Dicha elección proporcionará un diálogo mostrado a continuación. En este caso el mismo IDA recomienda restaurar la base de datos empaquetada.

La segunda situación puede ocurrir cuando no se ha guardado nunca la base de datos activa, esa es la única forma en que se presentan los archivos intermedios en el momento del fallo. En este caso, IDA ofrece la opción de reparación en el momento que trates de abrir de nuevo el archivo original.

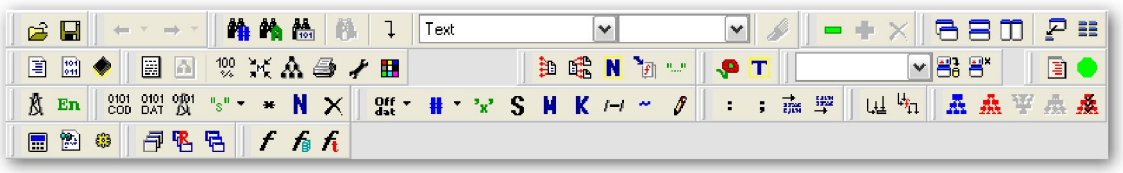
3.7.--Introducción al escritorio de IDA

Dada la cantidad de tiempo que utilizarás mirando fijamente a esta área de trabajo, querrás utilizar algún tiempo en familiarizarte con sus distintos componentes. Veamos una visión general del escritorio por defecto. El comportamiento de dicha área de trabajo durante el análisis la comentaremos después.



3.7.1.--Área de herramientas

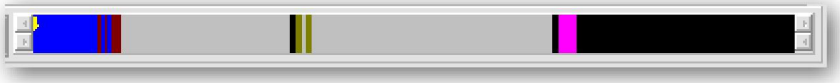
El área de barras de herramientas, contiene las herramientas correspondientes a las operaciones comúnmente utilizadas por IDA. Las barras de herramientas son añadidas o quitadas del área de trabajo utilizando la acción **View > Toolbars**. Utilizando atrapar y arrastrar puedes reposicionar cada una de las barras de herramientas en el lugar que necesites.



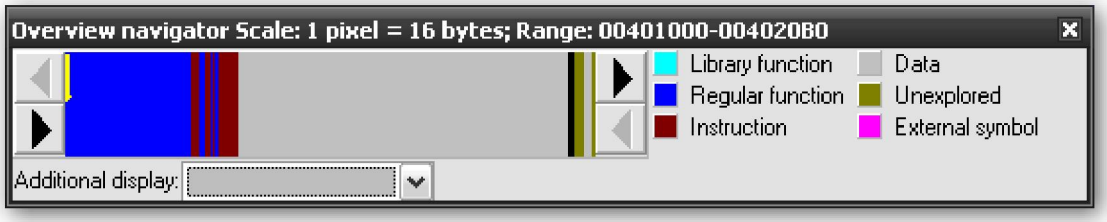
3.7.2.--Banda de navegación

La banda de colores horizontal es el navegador de visión general o banda de navegación. Dicha banda de navegación muestra una vista lineal del espacio de direcciones cargadas del archivo. Por defecto se representa todo el rango de direcciones del binario. Puedes ampliar o reducir (**zoom in, zoom out**) el rango de direcciones

haciendo click derecho en cualquier parte de la banda de navegación y también seleccionar cualquiera de las opciones disponibles de zoom. Los distintos colores



representan distintos tipos de contenido del archivo, como datos o código. Un pequeño indicador de posición, amarillo, por defecto, señala en la banda de navegación la dirección que corresponde al rango actual de direcciones mostradas en la ventana de desensamblado. Colocando el cursor del ratón en distintas partes de la banda de navegación aparece un rectángulo indicándote la ubicación de este punto en el binario. “Clickando” (participio del verbo “click”, jeje), en la banda de navegación aparecerá en la vista de desensamblado la ubicación del binario seleccionada. Los colores utilizados en la banda de navegación pueden modificarse utilizando la acción **Options > Colors**. Arrastrar la banda de navegación fuera de la zona de barras de herramientas nos proporciona esta vista general del navegador.



También nos muestra la posición actual del indicador (flecha amarilla) y las claves de color para identificar en grupos funcionales el contenido del archivo.

3.7.3.--Solapas

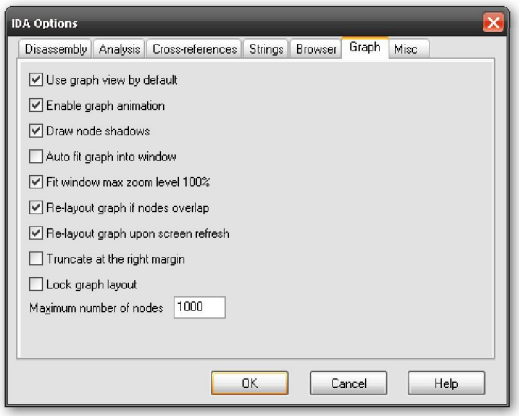
Cada una de ellas nos proporciona la vista de los datos actuales abiertos. Las vistas de datos contienen la información extraída del binario y representan distintas vistas de la base de datos.

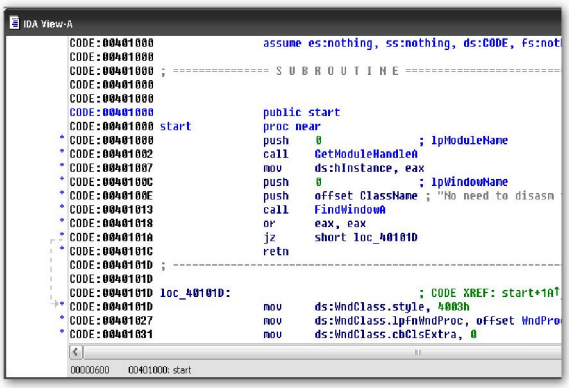
La gran parte del trabajo de análisis se realizará, probablemente, con la interacción de los datos disponibles en estas vistas. En la figura anterior vemos algunas de las vistas disponibles. **IDA View-A, Names, Exports, …**Si quieres tener más vistas de datos puedes añadirlas realizando la siguiente acción **View > Open Subviews**, y también puedes utilizar este menú para restaurar cualquier vista que hubieras cerrado inadvertidamente.

3.7.4.--Vista de desensamblado (disassembly view)

La vista de desensamblado es la primera vista de datos del binario. Existen dos estilos de vista de desensamblado distintos: la vista gráfica (**graph view**), por defecto, y la vista del listado (**listing view**).

En la vista gráfica, figura arriba, IDA nos muestra un ordinograma de una función en un momento dado. Cuando combinamos esta vista con la vista gráfica general (**graph overview**), ventanita pequeña de la figura, podemos obtener una comprensión del flujo de ejecución de la función utilizando el detalle visual de la estructura de dicha función. Cuando la ventana **IDA-View,** figura general, está activada, puedes pasar de la vista gráfica a la vista listado, figura abajo, simplemente pulsando la barra de espaciado. Si quieres tener una vista por defecto tienes que deseleccionar **Use Graph View by Default** en la solapa **Graph** realizando la acción **Options > General**, te mostrará el siguiente diálogo.





3.7.5.--Vista gráfica general (graph overview)

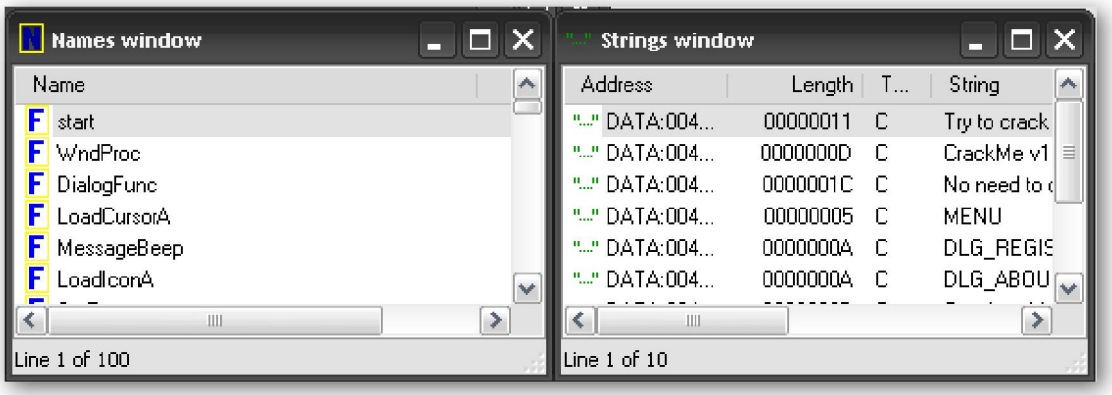
En el área de la vista gráfica, es muy raro poder ver la gráfica entera de una función de una vez. La gráfica de visión general, se muestra solamente cuando la vista gráfica está activada, proporcionando una instantánea reducida de la estructura gráfica básica. Un rectángulo de puntos en la vista general gráfica, indica la vista gráfica actual de desensamblado. “Clickeando” en esta, reposiciona la vista gráfica de acuerdo a la posición del click realizado.

3.7.6.--Ventana de mensajes (message windows)

La ventana de mensajes es donde puedes hallar cualquier información de salida generada por IDA. Aquí hallarás mensajes de status concernientes al progreso de la fase de análisis del archivo, conjuntamente con cualquier mensaje de error que se derive de las operaciones pedidas por el usuario. Esta ventana es similar a una ventana de consola.

3.7.7,-- Ventanas

Las dos vistas de datos que se muestran por defecto son las ventanas **Names y Strings**, las cuales también estudiaremos más adelante.



3.8.--Comportamiento del escritorio durante el análisis inicial

Una tremenda actividad se realiza en el área de trabajo durante el autoanálisis inicial de un archivo abierto recientemente. Puedes ir comprendiendo dicho análisis observando las distintas vistas del escritorio durante el proceso de análisis. Las actividades que podrás observar son:

** Seguimiento de los mensajes imprimados en la ventana de mensajes.

** Salida de ubicación y desensamblado inicial generada en la ventana de desensamblado.

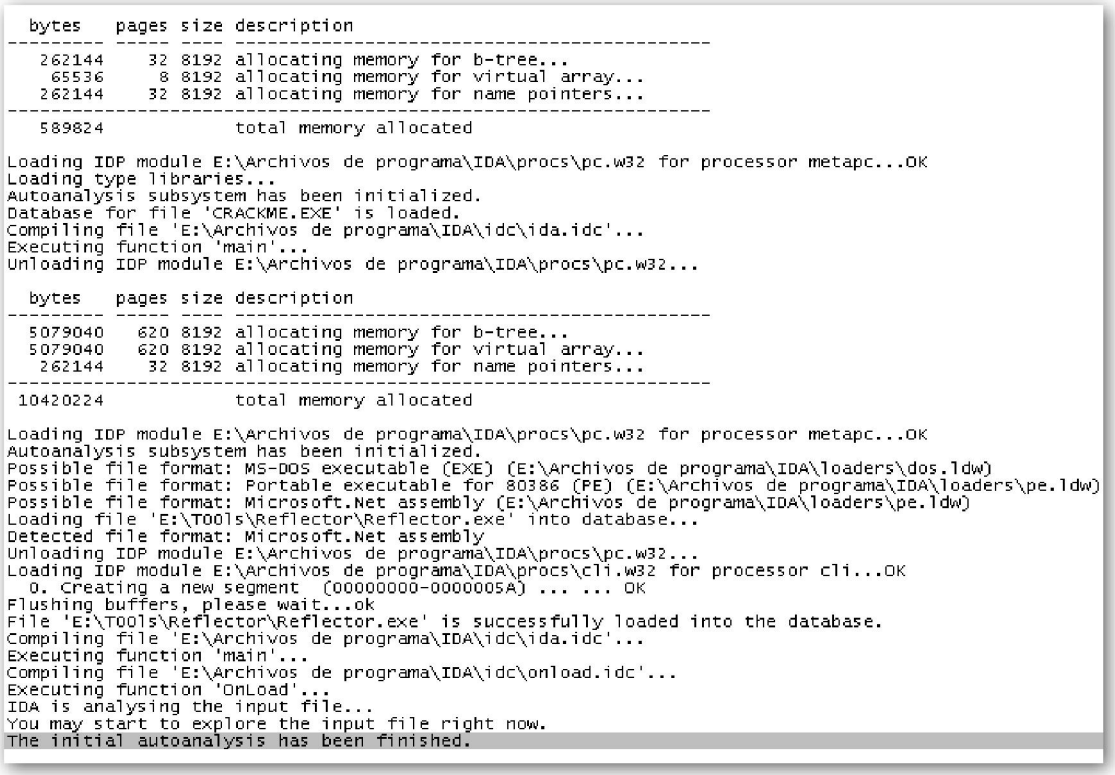
** Llenado inicial de la ventana **Strings**, seguido de un escaneado final de cadenas de caracteres (strings) al final de la fase de análisis.

** Llenado inicial de la ventana **Names**, Seguido de una actualización constante mientras el análisis se realiza.

** Transformación de la banda de navegación a medida de que son reconocidas nuevas áreas de datos y código del binario, los bloques de código son reconocidos como funciones y finalmente las funciones son reconocidas como librerías de código utilizando las técnicas de verificación de modelos de IDA.

** El indicador de posición de la banda de navegación va moviéndose por esta mostrando las regiones que actualmente se están analizando.

La siguiente salida representa los mensajes generados por IDA durante el análisis inicial de un archivo binario nuevo. Observa que la forma en que se muestran los mensajes es la explicación del proceso de análisis y ofrece una significativa secuencia de las operaciones realizadas por IDA durante el análisis al cargar, como ejemplo, el archivo Reflector.exe.



Hay dos mensajes de ayuda en particular, interesantes que son: **You may start to explore the input file right now** y **The inicial autoanálisis has been finished**. El primer mensaje te informa que IDA ha estado analizando ya el archivo y puede empezar a navegar a través de distintas vistas de datos. Navegar no implica cambiar, sin embargo tendrás que esperar, a realizar los cambios que quieras en la base de datos, hasta que la fase de análisis se haya completado. Si intentas cambiar la base de datos antes de acabar la fase de análisis, el motor de análisis puede cambiar tus cambios, incluso puede impedir que se realice el análisis correctamente. El segundo de los mensajes él mismo indica que no se realizarán más cambios automáticamente. Este es el momento en el que puedes empezar a realizar los cambios que quieras en la base de datos.

3.9.—Algunos recursos y trucos del área de trabajo de IDA

IDA ofrece una cantidad enorme de información, y su escritorio puede estar muy saturado. He aquí algunos trucos para sacarle mayor partido al escritorio:

** Cuando sea posible trabajar con IDA en un monitor grande o dos a la vez.

** No olvides que realizando la acción **View > Open Subview** es el medio para restaurar datos mostrados que hayas cerrado inadvertidamente.

** La acción **Windows > Reset Desktop** te ofrece la forma de restaurar el escritorio al esquema original.

** Utilizando la acción **Windows > Save Desktop** guardas el esquema de tu configuración actual del escritorio. Por otra parte la acción **Windows > Load Desktop** te permite revertir rápidamente el esquema guardado.

** La única ventana en la cual se puede cambiar el tipo de fuente es la ventana de desensamblado (**Disassembly**), tanto la vista gráfica como la de listado. Los distintos tipos de fuente se pueden habilitar realizando la acción **Options > Font.**

Para finalizar este primer tanteo a IDA, podemos decir que familiarizarte con el área de trabajo de IDA te proporcionará mucha experiencia con él. Realizar ingeniería inversa de código binario con IDA es bastante difícil hasta que se saben utilizar bien todas sus herramientas. Las opciones escogidas en el cargado inicial y los autoanálisis subsecuentes realizados por IDA al archivo cargado repercuten en mucho para todo los análisis posteriores. Puede ser que sólo con el análisis inicial de IDA hayas satisfecho tus necesidades de análisis del archivo. Pero si no es así, tendrás que realizar análisis posteriores interactivos con IDA, si es así, ahora estamos preparados para introducirnos en la funcionalidad de algunas vistas de datos de IDA. Por lo tanto seguiremos nuestro camino estudiando cada una de las vistas primarias, las circunstancias en las que se hallarán cada cosa que nos interese y como utilizar estas vistas para mejorar y actualizar sus bases de datos.

Performance Bigundill@