9.0.-- Las caras de IDA

9.1.—IDA en modo consola

El corazón de la consola de IDA es una **librería I/O** basada en **Borland** llamada **TVision**, esta es "portable" en distintas plataformas, Windows, Linux y Mac OS X, además de otras. El código fuente de **TVision** lo podemos descargar, previo pago, de la página de descarga de IDA.

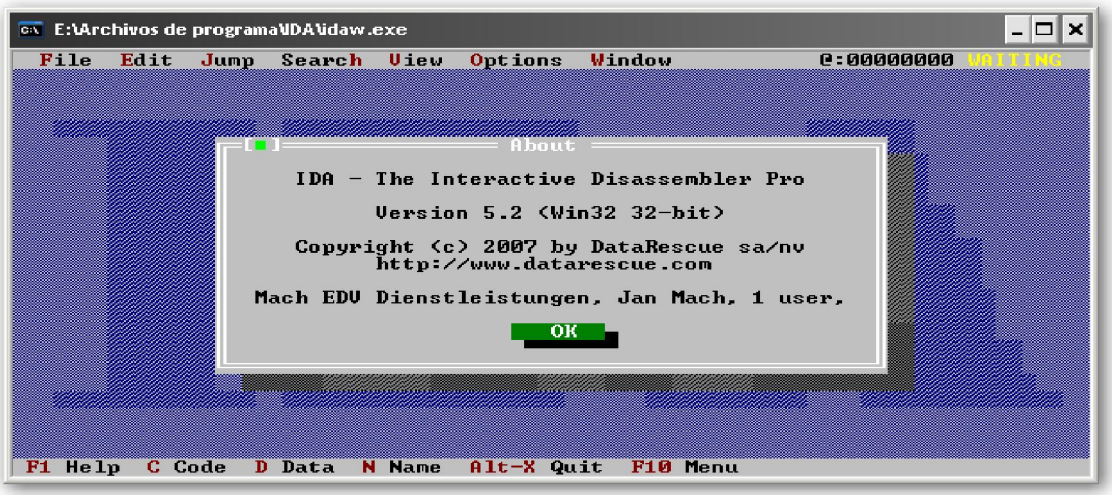
La utilización de una librería común en todas las plataformas nos proporciona una interfaz de usuario consistente. No obstante existen varios inconvenientes al cambiar de plataformas, como pueden ser; funcionamiento del ratón, redimensionados y teclas de atajo de la aplicación. Algunos de estos problemas los trataremos a medida que vayamos avanzando.

9.1.1.— Características comunes en modo consola

El término **modo consola** implica a, todas las versiones IDA basadas en texto que se ejecutan en una **terminal o shell** de algún tipo. Como hemos dicho estas consolas pueden tener problemas con el ratón, redimensionado, etc dándo lugar a limitaciones con las cuales tendremos que aprender a trabajar. Los tipos de dichas limitaciones dependerán de la plataforma y terminal que elijamos para trabajar.

La pantalla de consola de usuario, consiste en una barra de menú a lo largo de la línea superior de la pantalla mostrando un menú de opciones y estado; y otra barra de operaciones a lo largo de la línea inferior de la pantalla similar a una barra de texto. Las operaciones permitidas se activan utilizando teclas atajo o, cuando lo soporta, utilizando el ratón. Virtualmente, cada orden de la versión GUI es soportada de alguna forma en la versión consola, la mayoría de atajos asociados de la GUI funcionan correctamente.

El espacio que existe entre las dos barras es la ventana de visión. Sin embargo, ésta está limitada, utilices la terminal que sea, a una pequeña ventana de 80 x 25 caracteres y sin gráfica. Por lo tanto en las versiones IDA de consola, sólo se abren por defecto la ventana de desensamblado y la ventana de mensajes. Para acceder a las otras vistas de la versión GUI, IDA utiliza las librerías **TVision** solapando dichas ventanas y pudiéndolas abrir utilizando **F6**, en lugar de las pestañas. Cada ventana es numerada secuencialmente y el **ID** de la ventana se muestra en la esquina superior izquierda.



Cuando podamos utilizar el ratón en la consola, podremos redimensionar la ventana haciendo click y arrastrando la esquina superior izquierda al tamaño deseado. Para reposicionar la ventana haremos click en el borde superior de la ventana. Suponiendo que el ratón no es soportado, puedes mover y redimensionar la ventana con la acción **Window > Resize/Move o CTRL-F5** y utilizar las flechas del teclado para redimensionar la ventana activa. Si puedes redimensionar la terminal del programa con el ratón, IDA reconocerá el tamaño de la terminal nueva y la expandirá o encogerá la medida apropiada que hayas elegido.

Sin capacidad gráfica, el modo gráfico del desensamblado no funciona, y las flechas de control de flujo no se muestran en la parte izquierda del listado de desensamblado. Sin embargo todas las sub vistas de la GUI se soportan en consola. Al igual que en la GUI la mayoría de las sub vistas se muestran realizando la acción **View > Open Subviews**. La principal diferencia es que la ventana del volcado hexadecimal sólo la podemos ver en una única ventana. En vez de eso, puedes ir cambiando del desensamblado al volcado hexa realizando la acción **Options > Dump/Normal View o CTRL-F4**. Para poder tener ambas vistas abiertas simultáneamente, deberemos abrir una segunda ventana de desensamblado con **View > Open Subviews > Disassembly** y cambiar la nueva vista al volcado hexa. Por desgracia, no hay forma de sincronizar la vista hexa con la vista de desensamblado.

Si el ratón funciona, navegar a través del desensamblado es casi igual que en la versión GUI, si hacemos doble click en cualquier nombre nos lleva a su correspondiente dirección. De forma alternativa, si posicionamos el cursor en el nombre y pulsamos **ENTER** la vista salta a la correspondiente ubicación del nombre. Si presionamos **ENTER** mientras el cursor está posicionado en el nombre de una variable de pila nos abrirá la vista detallada del **stack frame** asociado a la función. Sin soporte de ratón, los menús funcionan de manera similar a otras aplicaciones de consola, empleando el método **ALT-x**, donde **x** es el carácter subrayado de la ventana.

9.1.2.— IDA en la consola Windows

La terminal Windows **cmd.exe** es terriblemente inflexible, pero soporta bien la versión de consola IDA. Para Windows la versión consola de IDA tiene el nombre de **idaw.exe**, mientras que la versión GUI se llama **idag.exe**. Lo mismo para la versión 64-bit, **idaw64.exe** y **idag64.exe**, respectivamente.

Para que el ratón funcione en IDA, debemos asegurarnos que tenemos deshabilitado el modo **QuickEdit** en la terminal en la cual ejecutamos IDA. Para configurar dicho modo en las propiedades de una terminal, hacemos click derecho en la barra de título de la terminal y seleccionamos **Properties**; entonces deseleccionamos **QuickEdit mode** en la pestaña **Options**. Esto lo deberemos de hacer antes de ejecutar IDA, sino el cambio no será reconocido por IDA.

A diferencia de las terminales Linux, las de Windows no se pueden expandir utilizando el ratón. En Windows, la versión consola de IDA sólo nos proporciona realizando la siguiente acción, **Window > Set Video Mode**, un menú de seis opciones para redimensionar a cmd.exe en tamaños fijos y como máximo de 255 por 100.

Aunque ningún modo gráfico es soportado en la ventana de desensamblado, los modos de graficado **legacy** de IDA se pueden utilizar mientras se está corriendo el programa en

una terminal Windows. Para seleccionarlo utilizaremos la siguiente acción **View > Graph**, esto ejecutará el programa **wingraph32** y nos mostrará el resultado del graficado. En las versiones de IDA para Windows, es posible abrir varios graficados a la vez y continuar utilizando IDA mientras estos están abiertos.

9.2.—Utilizar IDA en modo Batch

Todas las versiones de IDA pueden ejecutarse en modo **batch** para facilitar el proceso automático de sus tareas. El principal propósito para utilizar el modo **batch** es ejecutar IDA a través de un **script IDC** específico y cerrarlo una vez el script se ha completado. Disponemos de varias opciones de órdenes, para controlar el proceso que se ejecutará utilizando la ejecución en modo **batch**.

IDA no requiere de una consola para ejecutarse, haciendo muy fácil el poderlo incorporar virtualmente en cualquier tipo de script o programa adjunto. Cuando ejecutamos en modo batch, las versiones GUI de IDA, **idag.exe y idag64.exe**, no se muestra ningún componente gráfico. Ejecutar las versiones de consola **idaw.exe y idaw64.exe** genera una visión total de la consola la cual es cerrada automáticamente cuando el proceso batch es finalizado. La vista de la consola puede ser suprimida redireccionando su salida a un "null device", NUL para cmd.exe y /dev/null para cygwin, como podemos ver:

Los parámetros de control de IDA en modo batch son:

La opción **–A**, produce que IDA se ejecute de forma autónoma, lo que significa que no se mostrará ningún diálogo de interacción con el usuario.

La opción **–c**, produce que IDA elimine cualquier base de datos asociada con el archivo especificado en la línea de órdenes y genera una nueva base de datos de él.

La opción **–S**, se utiliza para especificar qué script IDC deberá ejecutar IDA cuando se ponga en marcha. Por ejemplo para ejecutar **miscript.idc**, la sintaxis sería la siguiente **–Smiscript.idc**, sin espacio entre la S. IDA busca el script nombrado en el directorio Archivos de programa\IDA\idc.

La opción **–B** invoca al modo batch y es equivalente a suministrar las órdenes **–A –c –Sanalysis.idc** para su ejecución. El script **analysis.idc** se adjunta con IDA y se utiliza par analizar el archivo nombrado en la línea de órdenes antes de realizar el volcado de un listado de desensamblado, archivo **.asm**, del desensamblado y cerrar IDA permitiendo el guardado y cerrado de la nueva base de datos que se ha generado.

En realidad la opción **–S** es la clave del modo batch, ya que solamente finalizará IDA cuando el script ordene que se cierre. Si el script no cierra IDA, utilizaremos la combinación de las opciones necesarias para automatizar el proceso de IDA. La programación de scripts IDC la veremos más adelante.

9.3.—La IDA GUI en plataformas no Windows

Bien si nos encontramos en el dilema de que queremos utilizar la GUI de IDA, pero nos negamos a utilizar el sistema Windows ¿Qué podemos hacer? Esta es la pregunta que se

realiza todo el mundo que quiere utilizar un software específico para Windows sin utilizarlo, pues la solución es la misma que para otro cualquier software.

La primera opción es utilizar un software de virtualización tal como **VMware Workstation** donde ejecutar una copia del sistema Windows y ejecutar IDA dentro de él. Con lo cual estás ejecutando IDA dentro de Windows.

La segunda opción es ejecutar la GUI de IDA utilizando el programa **Wine** el cual ejecuta ejecutables nativos en un sistema no Windows. Wine funciona con cualquier sistema Linux y también funciona bien en OS X.

Una vez tengamos instalado y configurado a Wine, podemos realizar la instalación de Ida para Windows y desde Wine ejecutarlo. IDA funciona aceptablemente bajo Wine y nos proporciona total funcionalidad, incluido el acceso a los modos gráficos como wingraph32. El único problema que podemos tener es que la fuente soportada no se visualice adecuadamente en el listado de desensamblado. Para solucionar este problema podemos instalar una o más fuentes, si instalamos la fuente Courier funciona bien tanto con IDA como con Wine.

Independientemente de la solución elegida, es importante recordar que la capacidad de depuración local de IDA dependerá de la versión que estés ejecutando, no el sistema operativo que utilices. En otras palabras, no puedes depurar localmente binarios Linux utilizando la GUI de IDA y ejecutándola en Wine. La versión Windows GUI de IDA sólo puede ejecutar depuración local de binarios Windows. La depuración remota es otro apartado del cual hablaremos con más detalle más adelante.

Performance Bigundill@