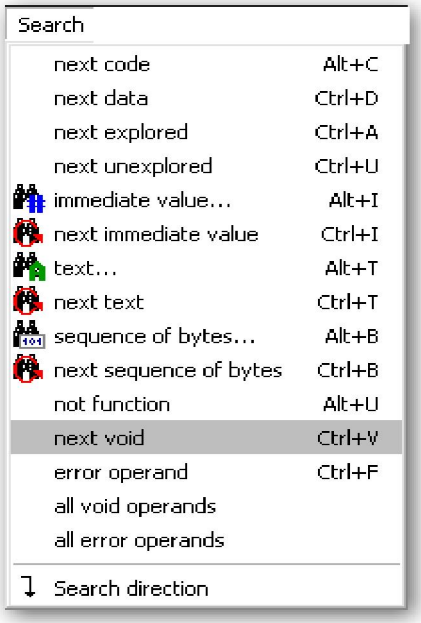
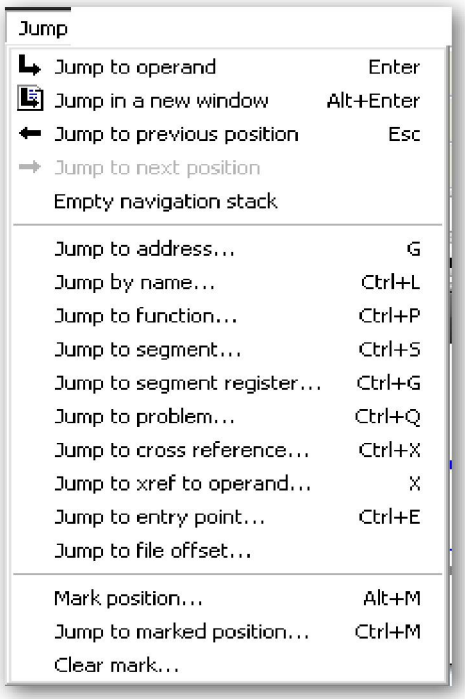
5.3.—Buscar en la base de datos

Como ya hemos visto es fácil desplazarse por el desensamblado de IDA utilizando toda la información que nos proporciona, asimismo igual de fácil es encontrar la información que nos pueda interesar dentro de las diferentes vistas de datos que nos ofrece, como puede ser (names, strings, imports y otras). Bien, pues con esta premisa vamos a investigar qué características de ayuda nos proporciona IDA para realizar la búsqueda de información en la base de datos que crea del archivo. Si abrimos el menú **Search**, hallaremos una larga lista de opciones, la mayoría relacionada con el siguiente (**next**) elemento de distintas categorías.



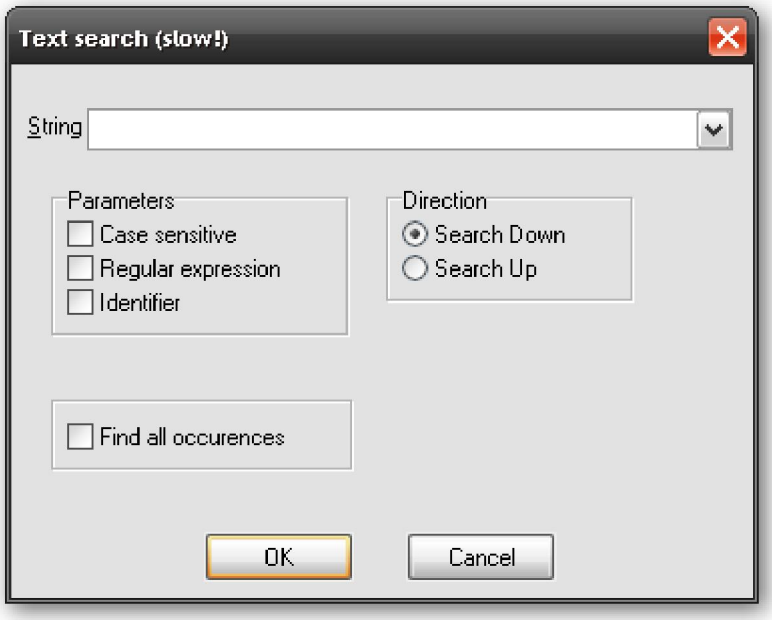
Por ejemplo, si optamos por la acción **Search > Next Code** moveremos el cursor en el desensamblado hasta la siguiente ubicación que contenga una instrucción. Otra opción de búsqueda la encontramos en la opción de menú **Jump** con las cuales también nos tendremos que familiarizar.



La mayoría de opciones te ofrecen distintas posibilidades de elección de saltos. Por ejemplo si realizamos la acción **Jump > Jump to function**, se nos abrirá una ventana de todas las funciones del archivo, permitiéndonos elegir una y desplazarnos a ella rápidamente. Además de estas dos opciones de búsqueda a través de menú, existen dos formas más de búsqueda general las cuales merecen explicarlas detalladamente; una es la búsqueda de texto y la otra búsqueda binaria.

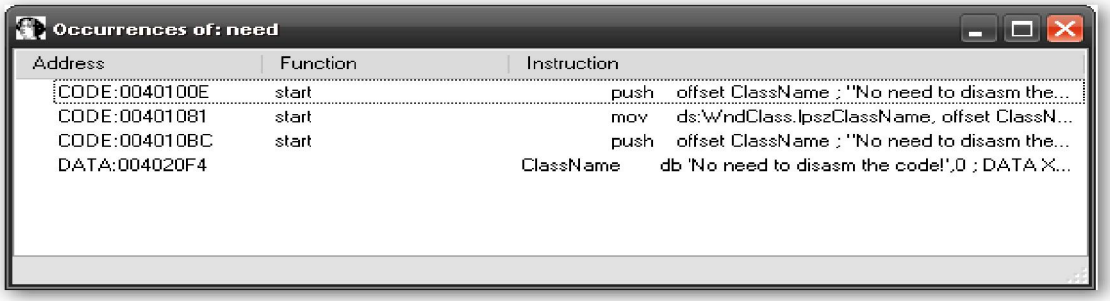
5.3.1.— Búsqueda de texto

La búsqueda de texto en IDA, realiza la búsqueda de subcadenas de texto a lo largo de toda la vista de listado de desensamblado. Dicha búsqueda se inicia con la acción **Search > Text**, o el atajo **ALT-T**, con lo cual se nos muestra el diálogo, figura abajo.



Vemos que existen varias opciones para especificar detalles concernientes a la forma de ejecutar la búsqueda. La opción **Case sensitive** es evidente, **Regular expression** nos permite encontrar cadenas con estilo **POSIX** y la opción **Identifier** es para hallar algo renombrado. En realidad restringen la búsqueda para hallar las palabras, que coincidan con las opciones, en cualquier línea de desensamblado, incluidos **opcodes mnemonicos** o **valores constantes**. Una búsqueda como **Identifier** de **00401253** nos puede hallar un símbolo llamado **sub_401116**.

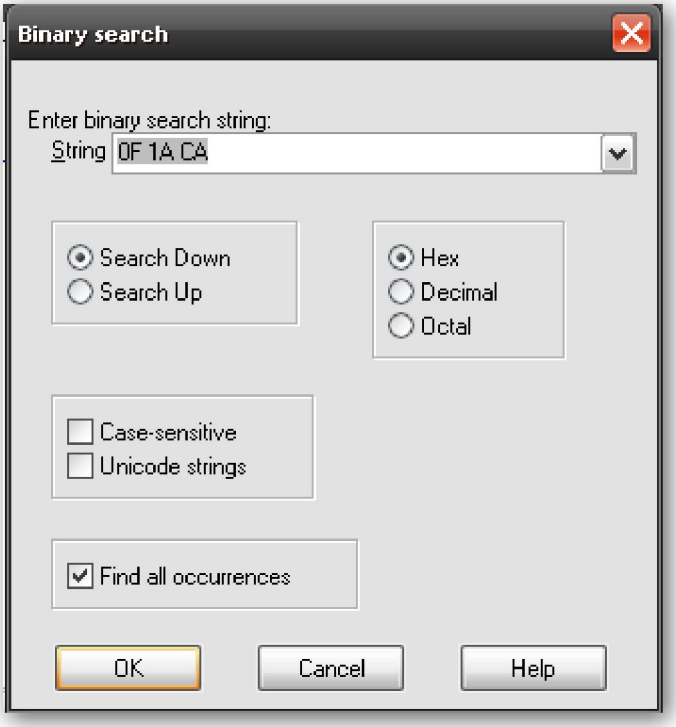
Si seleccionamos la opción **Find all occurences** los resultados de la búsqueda nos abrirá una nueva ventana, permitiéndonos el desplazamiento a cualquier elemento hallado según el criterio de búsqueda.



Para finalizar diremos que la búsqueda realizada se puede repetir para localizar la siguiente coincidencia utilizando el atajo **CTRL-L** o realizando la acción **Search > Next Text.**

5.3.2.— Búsqueda binaria

Si lo que necesitas es buscar un contenido binario específico, como una secuencia de bytes conocida, entonces la búsqueda de texto no te sirve. En vez de esa necesitas utilizar la característica que te facilita IDA para búsqueda binaria. Mientras que la búsqueda de texto se realiza en la ventana de desensamblado, la búsqueda binaria se realiza en la ventana **Hex view**. En esta se puede buscar cualquier volcado de datos hexadecimales o de datos ASCII, esto dependerá de cómo se especifique la búsqueda de la cadena. Para realizar una búsqueda binaria utilizaremos la acción **Search > Sequence of Bytes** o el atajo **ALT-B**, la figura abajo, muestra el diálogo **Binary Search**. Para buscar una secuencia de bytes hexadecimales, la cadena de búsqueda deberá especificarse como una lista de valores hexa de dos dígitos separados por un espacio cada pareja, por ejemplo **0F 1A CA**, y aunque tengas la opción **Case-sensitive** seleccionada la búsqueda realizada como **0f 1a ca** dará el mismo resultado.



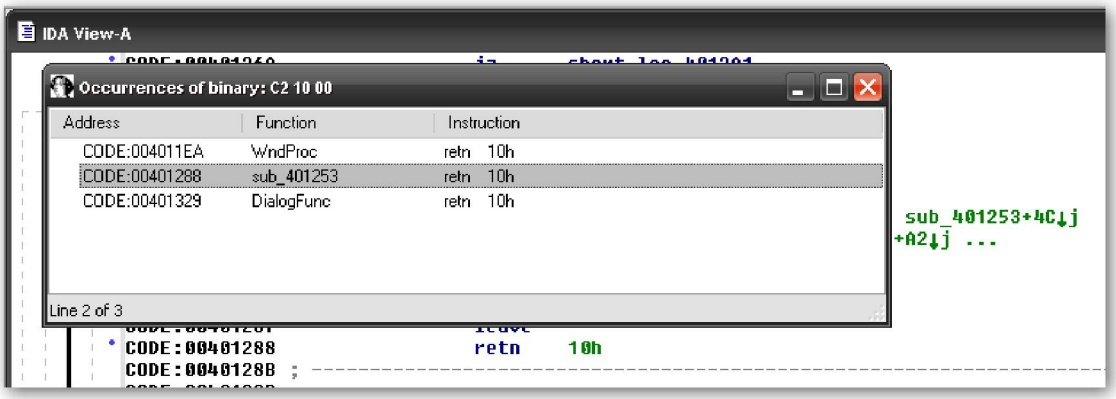
Como búsqueda alternativa de cadenas de datos adjuntadas al código, para realizar una búsqueda efectiva de datos ASCII en el volcado de la ventana **Hex View**, debemos poner **comillas** a las cadenas. Si usamos la opción **Unicode Strings** se buscará la cadena en versión Unicode.

La opción **Case-sensitive** puede ser causa de confusiones. La búsqueda de cadenas es directa; una búsqueda para **CrackSLatinoS** dará el mismo resultado que **crackslatinos** esté o no seleccionado **Case-sensitive**. La cosa cambia un poco cuando se realizan búsquedas hexadecimales sin seleccionar Case-sensitive. Veamos si buscas el valor **E9 41 C3** la búsqueda te puede sorprender mostrándote como coincidente a **E9 61 C3**. Las dos cadenas se consideran coincidentes debido a que **0x41** corresponde al carácter **A**

mientras que **0x61** corresponde al carácter **a**. Esto ocurrirá siempre que realices una búsqueda hexadecimal sin seleccionar **Case-sensitive**.

Observación: Cuando realices búsquedas hexadecimales, asegúrate de especificar Case-sensitive si quieres restringir la coincidencia exacta. Esto es muy importante si lo que quieres buscar secuencias específicas de opcodes.

Para buscar sucesivas coincidencias de datos binarios, se realiza con el atajo **CTRL-B** o la acción **Search > Next Sequence of Bytes**. Para finalizar diremos que la búsqueda binaria de datos no es necesario realizarla en la misma ventana **Hex View**. IDA nos permite especificar un criterio de búsqueda binaria en la vista de desensamblado mientras esté activa, en cuyo caso nos desplazará en la ventana de desensamblado a la primera coincidencia encontrada.



Performance Bigundill@