Ingeniería Inversa del Software

- Parte III -

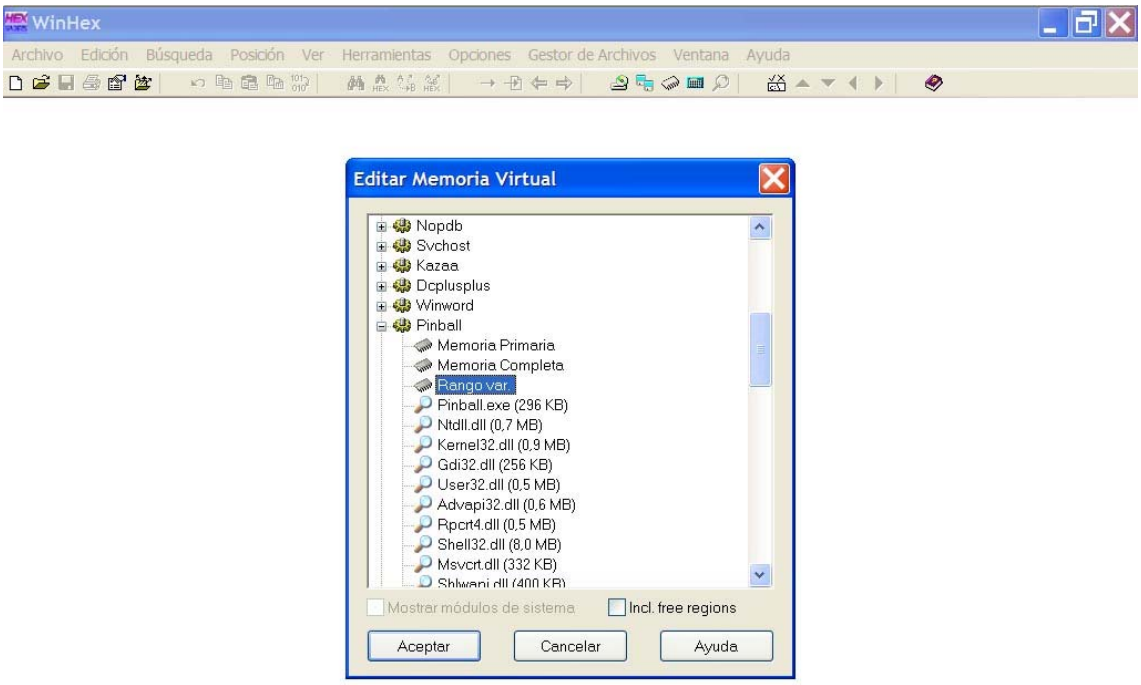
El editor de memoria

Como ya se comentó en el primer escrito de esta colección sobre la IIS, el editor de memoria es una utilidad que permite localizar y modificar el valor de variables que se encuentran en la memoria RAM del sistema. En estas variables se almacenan todos los datos que maneja un programa, incluso la información sobre si está registrado o cuantos días de prueba quedan. De la misma manera, este tipo de editor también se puede emplear para hacer trampas en los juegos, lo cual no deja de ser un ataque al software.

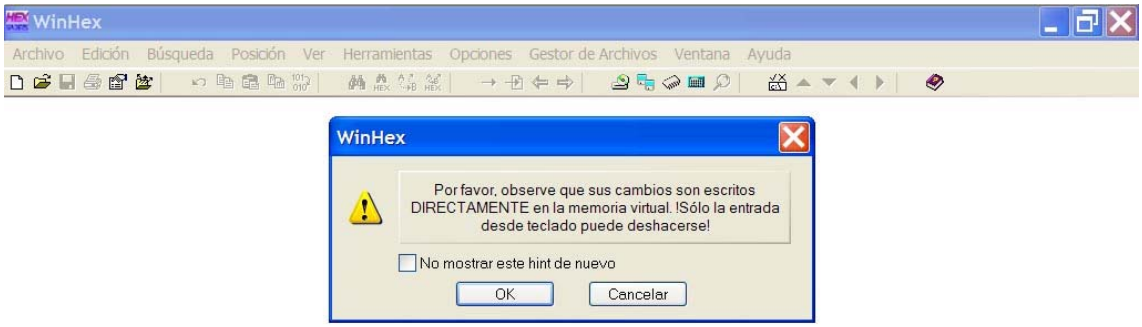
Cuando se trabaja directamente con la memoria del sistema no se dispone de la lista de nombres de las variables ni nada por el estilo, solo se tienen códigos hexadecimales representando sus valores. Por lo tanto, la parte más compleja del proceso que se llevará a cabo para modificar el valor de una variable de un programa consiste en conocer la posición donde está almacenada en memoria esta variable.

En este reportaje se pretendía mostrar como se puede modificar una variable (la puntuación) de un programa (el Pinball de Windows) con un editor de memoria (WinHex), pero debido a la incompatibilidad con Windows XP de los editores de memoria que conozco y manejo (WinHex y GameWizard32) no me es posible mostrar el desarrollo completo, por lo que se intentará clarificar al máximo las partes que no se han podido realizar. Los pasos a seguir en este proceso son las siguientes:

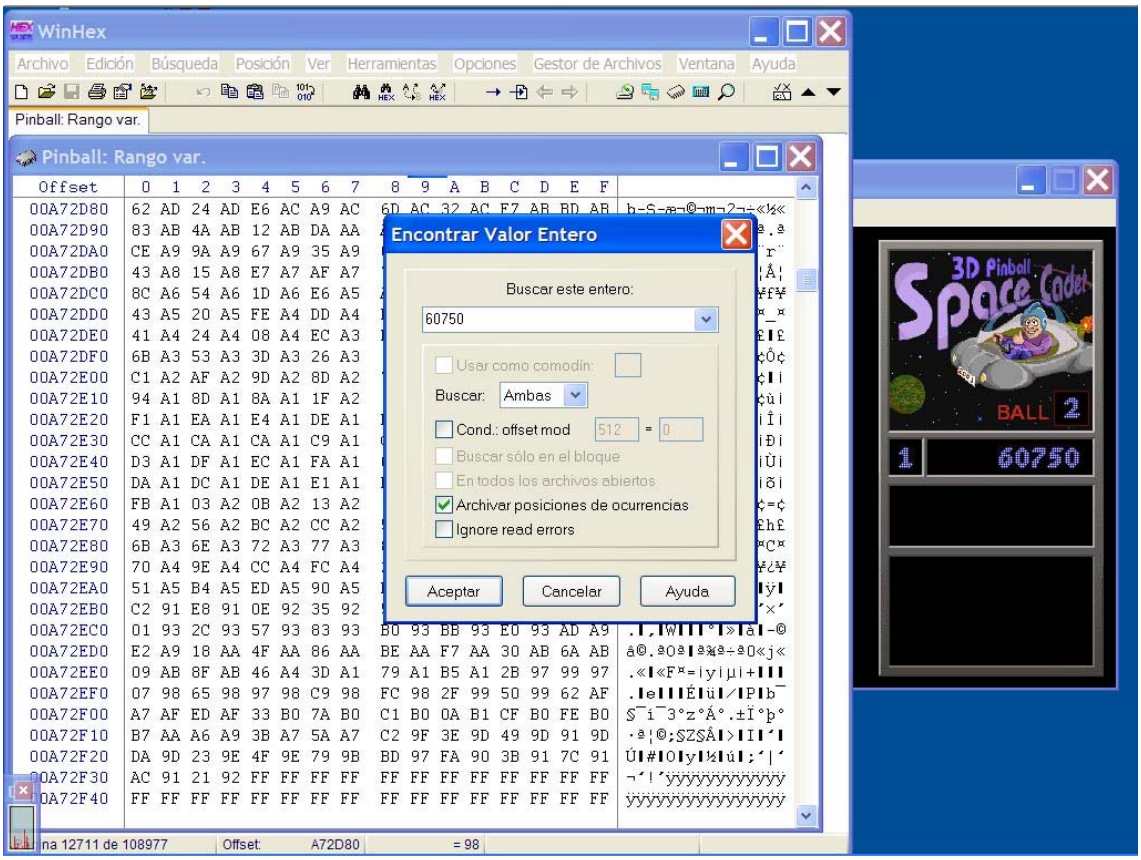
1. Primero hay que seleccionar sobre qué proceso en ejecución se trabajará.



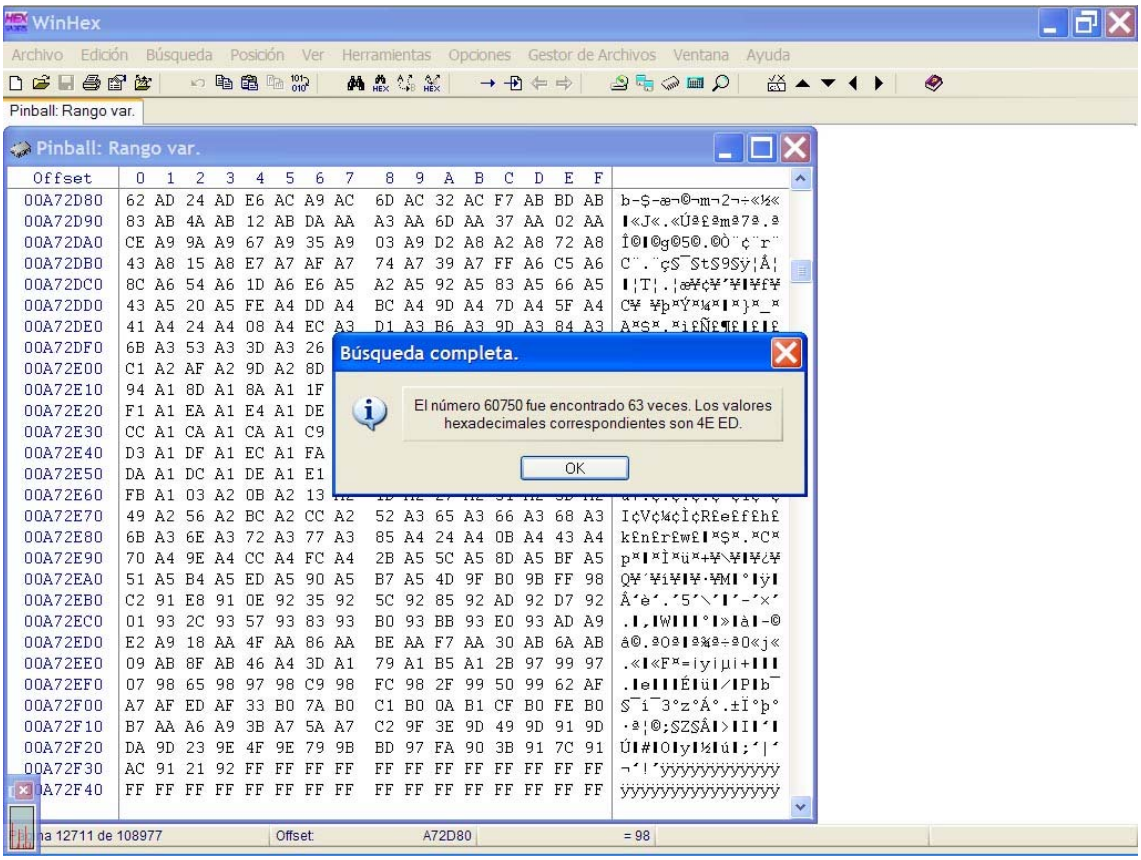
Nota: el propio editor avisa del riesgo que implica su uso.



2. Se pide al editor que monitorice todas las posiciones de la memoria donde se encuentre el valor que tiene actualmente la variable. Si se trata de un número, hay que indicar en que formato se quiere buscar (byte, short, integer, float, double...) ya que según el formato empleado se almacenará con un código hexadecimal u otro. Éste es el único detalle que puede complicar un poco este acto vandálico, aunque no mucho porque la mayoría de los programadores suelen usar o bien el menor tamaño de dato que les valga, o bien el tipo integer siempre que no se necesite otro mayor.



Lo normal es que aparezcan varias ocurrencias de la búsqueda, ya que puede repetirse por casualidad o por copia de valores en otras variables del programa, como por ejemplo, en este caso, en rutinas que manejen la puntuación.



Si el WinHex funcionara bien, aparte de decir el número de ocurrencias debería crear un documento de texto indicando ordenadamente cada una de las direcciones donde se ha encontrado. Esto no ocurre así, por lo que a partir de aquí no podré mostrar más capturas.

3. Al haber localizado más de una aparición del dato buscado, se hace necesario cambiar su valor. En este caso, por ejemplo, se aumenta la puntuación jugando un poco más.

4. Seguidamente, si se usa el WinHex, hay que buscar el nuevo valor de la variable que se quiere controlar y comparar a ojo los dos documentos generados con las listas de direcciones donde ha aparecido.

   Si se usara el GameWizard32, los puntos anteriores habrían sido idénticos, pero en este solo se le tendría que preguntar en que posiciones de las encontradas anteriormente se ha producido un cambio al nuevo valor. Es por tanto un proceso más rápido y sencillo que si se usa el WinHex.

Así habría que repetir estos dos últimos pasos hasta que solo quedara una única posición posible. Si no quedara ninguna posición, está claro que hay que cambiar el tipo de dato y probar con otro.

5. Finalmente, una vez localizada la posición, se va a ella y se modifica manualmente. También es posible crear un script en cualquiera de estos programas para hacer que cuando la variable monitorizada cambie a un determinado valor, se modifique tomando otro valor distinto. Esto es útil, por ejemplo, para conseguir vidas infinitas en un juego.

Jöx€@n Nieto