

IV

Mi primer script

Si hay algo importante en el rom hacking son los scripts. Estos nos permiten ir construyendo la historia de nuestro juego y dependiendo de los conocimientos de scripting que tengamos podremos hacer una historia mejor o peor. Por eso, es importante que pongamos empeño en el aprendizaje de este campo del rom hacking. Así que vamos a empezar por aprender cómo hacer nuestro primer script. La herramienta que utilizaremos será **XSE actualizado**. Es muy, muy importante que esté actualizado.

•Construyendo el script:

Bien, en primer lugar vamos a aclarar una cosa importante. Los scripts están compuestos por comandos. Cada comando tiene una función específica, es decir, cada uno tiene un efecto diferente en el juego. Por otra parte, el script es completamente lineal, se lee desde arriba hacia abajo, por lo que nuestros comandos siempre deberán estar bien ordenados para que tengan el efecto deseado sobre el juego.

A parte de esto, tenemos que introducir un nuevo concepto: Los pointers. Estos son bien sencillos, lo que hacen es llevarnos de una parte del script a otra cuando lo deseemos. Por ejemplo, si queremos hacer que algo sólo pase una vez, utilizaremos los pointers para que a la segunda vez, este pointer nos lleve a otro lugar. No obstante, eso es algo que veremos más adelante, ahora vamos a centrarnos en hacer nuestro primer script.

Los pointer son palabras precedidas por una arroba (@) para diferenciarse de los demás. Cada pointer distinto debe tener un nombre distinto dentro del script para diferenciarse. Una vez que se compila el script los pointer desaparecen transformándose en offsets, pero eso se verá también más adelante. Ahora sólo tenéis que saber que los pointers en el mismo scripts siempre tienen que ser diferentes.

Lo intentaré exponer gráficamente.

@hola  #org @hola

@adios  #org @adios

“@hola” llevará a “#org @hola” y “@adios” a “#org @adios”. Pero en el caso de que repitiéramos pointers tendríamos un problema:

@hola  #org @hola
 #org @hola

¿A cuál de los dos apunta el script? Ahí nos daría error. Cuando avancéis en el tutorial volved aquí y revisad esto, veréis como entonces lo comprendéis mejor.

Bien, vamos a empezar a escribir nuestro script:

```
#dynamic 0x(offset)
```

Esto especifica a partir de qué dirección empezará XSE a buscar un espacio vacío donde compilar (escribir) el script. Yo por ejemplo pondré la dirección *0x800000*. A partir de ahí en las bases de Rojo fuego/Verde hoja y Rubí/Zafiro hay mucho espacio libre.

Ahora vamos a poner el pointer inicial. Esto siempre se debe poner al inicio del script, ya que al compilar se convertirá en un offset llamado *offset núcleo* que será el que le asignemos a los minisprites del juego o a los scripts para que nuestro script se ejecute, se pone así:

```
#org @(pointer)
```

Yo lo cambiaré por la palabra “comienzo”, por ejemplo. Así que ahora mismo llevaríamos esto:

```
#dynamic 0x800000
```

```
#org @comienzo
```

A partir de ahí comenzará nuestra cadena de comandos. Lo que vamos a hacer será que un minisprite (persona) nos hable cuando pulsemos “A”. Vamos, como hacen en el juego. Vamos a empezar con los dos comandos más básicos:

- Lock: bloquea el movimiento. Así podremos hablar sin que la persona con la que hablamos se vaya.
- Faceplayer: Hace que la persona con la que hablamos nos mire.

Así que los voy a introducir en el script:

```
#dynamic 0x800000
```

```
#org @comienzo
```

```
Lock
```

```
faceplayer
```

Ahora tenemos que hacer que nos diga algo. Eso lo haremos mediante el comando *msgbox*. Se utiliza de esta forma:

```
Msgbox @(pointer del texto) 0x(tipo)
```

El pointer del texto es, lógicamente, el pointer en el que escribiremos nuestro texto. Yo le pondré, por ejemplo, “mensaje1”. Los pointers no pueden llevar espacios, deben ser de una sola palabra, es decir, “mensaje1” sería un pointer correcto, “mensaje 1” ya no lo sería. Respecto al tipo, dependiendo de si es un cartel, una persona, etc, tiene un tipo. El tipo de *msgbox* nos dice que caja de texto aparecerá en el juego. No obstante eso no es muy importante, por lo que nosotros utilizaremos el universal: 0x6. Bien, pues nuestro script quedaría así:

```
#dynamic 0x800000
```

```
#org @comienzo
```

```
Lock
```

```
faceplayer
```

```
msgbox @mensaje1 0x6
```

Como pusimos el comando “lock” para congelar el movimiento del minisprite, ahora tenemos que liberarlo, para ello utilizaremos el comando “release”. Siempre que se utilice “lock” hay que utilizar “release”. Y como ya hemos acabado el script, introduciremos el último comando: *end*. Este, obviamente, hay que ponerlo al final del script. Ya tendríamos nuestro script:

```
#dynamic 0x800000
```

```
#org @comienzo
```

```
Lock
```

```
Faceplayer
```

```
msgbox @mensaje1 0x6
```

```
release
```

```
end
```

Ya está, nuestro primer script es tan simple como eso. Pero pasa algo y es que nos falta el texto del msgbox. Bueno, no pasa nada, creamos un nuevo script donde estará nuestro texto:

```
#org @mensaje1
```

¿Por qué “mensaje1”? Pues porque en el msgbox hemos puesto que se llama al pointer “mensaje1” que es donde estará el texto. Ahora debemos poner el texto, que se escribe precedido de un igual (=) de la siguiente forma:

```
#org @mensaje1
```

```
= (texto)
```

Ahora escribimos un texto y el script quedaría así:

```
#dynamic 0x800000
```

```
#org @comienzo
```

```
Lock
```

```
Faceplayer
```

```
msgbox @mensaje1 0x6
```

```
release
```

```
end
```

```
#org @mensaje1
```

```
= Hola ¿Qué tal? El profesor Oak dice que vayas al laboratorio.
```

Este sería el script, bastante sencillo. Lo que pasa es que el texto se saldría de la caja en el juego ¿Por qué? Porque antes de escribirlo tenemos que aprender a usar los códigos en ellos. Así que ese será el siguiente apartado que veamos.

•Escribiendo textos:

Debéis de saber que XSE trae una herramienta para crear los textos automáticamente. Se accede a ella en “Herramientas>Ajustes de línea” o pulsando “ctrl+t”. Podéis hacer unas pruebas para ver cómo funciona y así ayudaros a entender el funcionamiento. Pero lo primordial es que aprendáis a hacer las líneas vosotros mismos y es lo que explicaré a continuación.

Estos son los comandos en las líneas:

`\p` = Crea un nuevo párrafo. Es decir, el textbox se limpia y escribe una nueva línea.

`\n` = Escribe una segunda línea en el msgbox de manera automática.

`\l` = Escribe una segunda línea en el msgbox al pulsar “A”. Es igual que la anterior pero al pulsar.

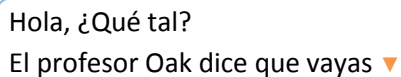
No obstante, para ver cómo funciona esto hay que hacer pruebas y eso es lo que os recomiendo hacer. Ahora vamos a tomar nuestra frase para insertarle los comandos.

= Hola ¿Qué tal? El profesor Oak dice que vayas al laboratorio.

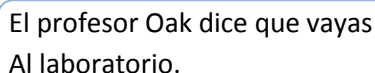
Y ahora vamos a añadirle los comandos:

= Hola ¿Qué tal?`\n`El profesor Oak dice que vayas`\l`al laboratorio.

Y eso es todo. Los comandos se insertan en la línea sin espacios ni nada. Eso es importante. Ahora el mensaje debería verse así en el emulador:



(y al pulsar “A” bajaría una línea quedando así:)



(Por último, al pulsar “A” de nuevo acabaría el msgbox)

Una vez que hagáis vuestro primer script, lo recomendable es que hagáis más haciendo pruebas con el texto y ayudándoos del editor de XSE para comprenderlo bien.

•Compilando y asignando el script:

Ahora que ya tenemos nuestro script completamente preparado, lo insertaremos. Si recordamos, el fruto de nuestro trabajo fue el siguiente:

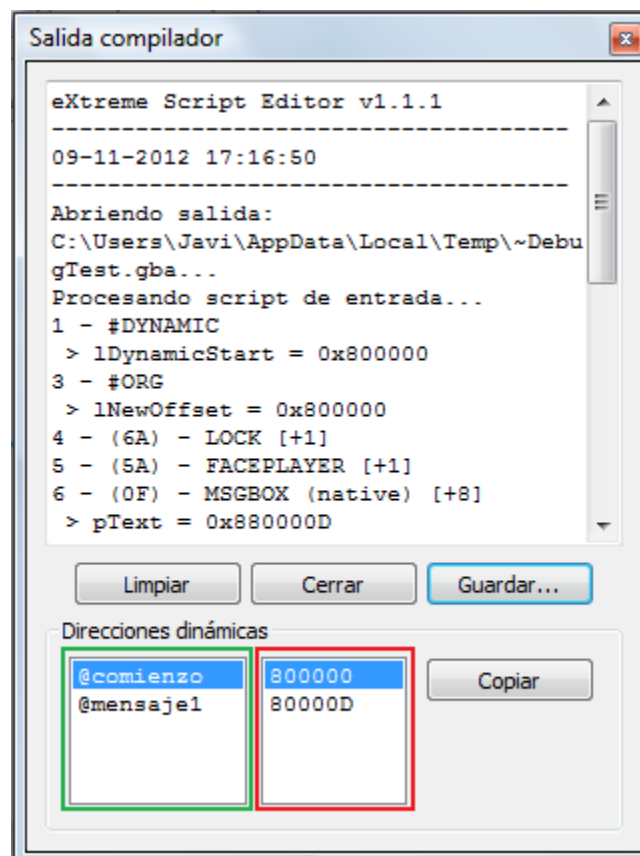
```
#dynamic 0x800000
```

```
#org @comienzo
Lock
Faceplayer
msgbox @mensaje1 0x6
release
end
```

```
#org @mensaje1
= Hola ¿Qué tal?\nEl profesor Oak dice que vayas\lal
laboratorio.
```

Deberíais haber ido construyendo el script en XSE tú mismo a medida que leías el manual, pero como es muy probable que no lo hayas hecho, copia y pega el script en XSE. Pero antes deberás hacer una cosa más importante: Cargar el rom.

Vamos a “Archivo>Abrir” y seleccionamos nuestro juego. Una vez hecho esto, ya está todo preparado para compilar. Así que ahora pulsaremos en el icono de los engranajes y la herramienta compilará nuestro script de forma automática. Ahora deberemos ver una ventana como esta:



He resaltado dos partes de la ventana con colores y es lo único que realmente nos importa de esta ventana. Como podéis ver, en la caja coloreada en verde están los pointers que habéis utilizado en el script. En la caja coloreada en rojo, por otra parte, se muestra el offset el que se ha compilado esa parte de nuestro script. Los pointers realmente no se compilan, por eso al decompilar el script no volverán a aparecer, los pointers son utilizados como títulos para el script de manera que podamos llamar a ese script desde otro. Simplemente es una manera de

ordenar el script, por así decirlo. Realmente se llama al offset donde está compilado, pero de esta forma es mucho más sencillo, claro.

¿Recordáis lo del *offset núcleo*? Era el offset principal de nuestro script, la parte más importante, de donde parte nuestro script. Su corazón. En nuestro caso a ese *offset núcleo* le hemos llamado “@comienzo”, así que lo seleccionamos y le damos a “Copiar”. Automáticamente se copiará el offset del script, en nuestro caso y como podemos ver, el “800000”.

Ahora ya solo queda asignarle ese offset a un mini del juego para que al interaccionar con él se ejecute el script y se comporte de la manera que hemos especificado en el mismo, es decir, nos mirará y nos dirá el texto que hemos puesto. Así que abrimos Advance Map y seleccionamos el mini que va a llevar el script. Una cosa importante que debéis saber es que un mini solo puede llevar un único script, por eso son tan necesarias las flags y las variables que veremos más adelante.

Una vez elegido el mini vamos a la parte derecha y buscamos donde dice “Script offset”. Deberemos ver algo así “\$80BBBC” si el mini ya tenía un script asignado o así “\$000000” si no tenía ninguno o es un nuevo mini que hemos añadido nosotros mismos. Ahí deberemos poner nuestro offset sin borrar el “\$”, es decir, en nuestro caso quedaría así: “\$800000”. Ya hemos asignado el offset al mini, ahora ya debería ejecutarse al interactuar con él, así que abrimos el emulador, nos dirigimos al mini y comprobamos que, en efecto, se ejecuta nuestro script.

Lo más importante en este momento es hacer vuestras propias pruebas. Haced el mismo script con otro diálogo, probad con la opción de ajuste de líneas de XSE, etc. Es imprescindible que comprendáis esta parte del manual para poder comprender la otra.

“Mi primer script”

Manual redactado por **Javi4315**.

Queda estrictamente prohibida su distribución.