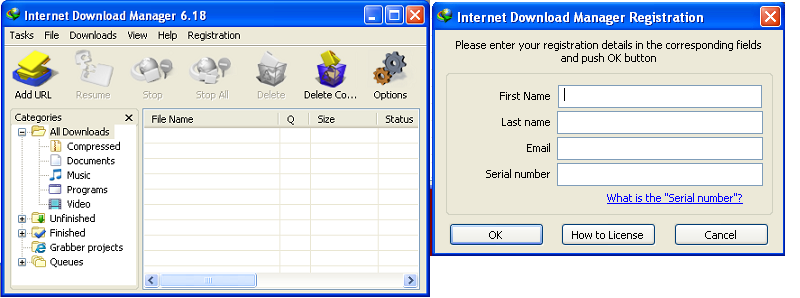
|  |  |  |  |
| --- | --- | --- | --- |
| Programa | Internet Download Manager | | |
| Protección | Serial | | |
| Descripción | KeyGen & Patch | | |
| Dificultad | Fácil | | |
| DownLoad | www.internetdownloadmanager.com | | |
| **Compilador** | VC++ 6 | | |
| **Herramienta** | Olly – Sniffer – UltraEdit | | |
| **Cracker** | TrashAmbishion | Fecha | 23/10/2013 |

**Disclaimer**: El autor de este materia para nada se hace responsable del posible mal uso que se le pueda dar, su único objetivo es didáctico. La razón de porque hago publico este material es porque ya existen KeyGen, Patch y Crack’s realizados a este software.

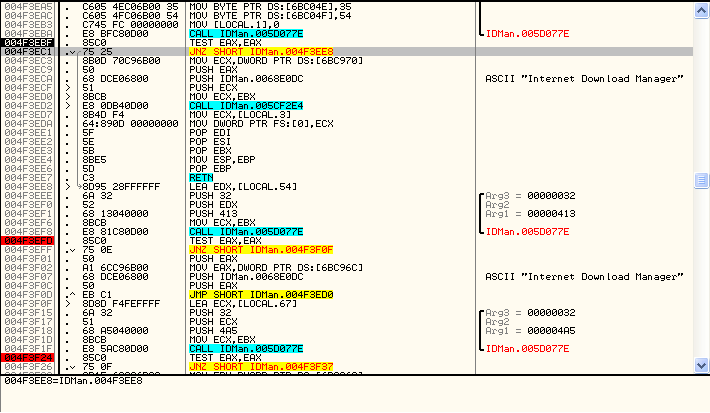
**Prologo:**

Este programa me lo paso un colega con su serial pero en el TXT donde venía los datos ponían un mensaje adjunto que dice que tengo que bloquearle el acceso a [www.internetdownloadmanager.com](http://www.internetdownloadmanager.com) y la razón de ello es que trata de verificar por nuevas actualizaciones y de paso mira si nuestro serial es genuino, entonces me daba **FIACA** el tener que hacer ese bloqueo por lo que me puse a estudiarlo y ya de paso me embulle a realizarle el **KeyGen**. *Esto fue después de que en varias ocasiones me bloqueara el programa porque el serial no lo tenían ellos registrado en sus servidores.* :-) Raro ehhh..

**Manos a la Obra:**



Aquí tenemos al programa y el cartelito para ingresar los datos, lo llenamos y antes de dar Ok, vamos al Olly y ponemos un Bp GetWindowTextA ahora damos en Ok y para en nuestro BP avanzamos con Ctrl + F9 (3 veces) y después con F8, llegamos aquí



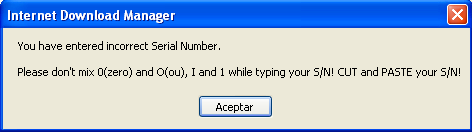
Fíjense que más debajo de la función de la que acabo de salir se repite la llamada en 2 ocasiones más (más arriba esta la otra) recuerden que ingresamos 4 datos aquí entonces está guardando esos datos para hacer algo con ellos, si desean ver donde los guarda el parámetro Arg2 es el indicado.

Traceamos con F8 hasta:

004F403C |. 83F9 17 CMP ECX, 17

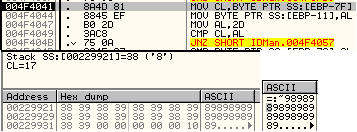
004F403F |.^ 75 98 JNZ SHORT IDMan.004F3FD9

Tenemos la comparación de ECX con 17 (23 en decimal) ECX en mi caso es 16 que es 22 en decimal porque el serial que puse fue 9898989898989898989898 por lo que no serán iguales y se activara el Jump que mandara a un mensaje de error demos Run,

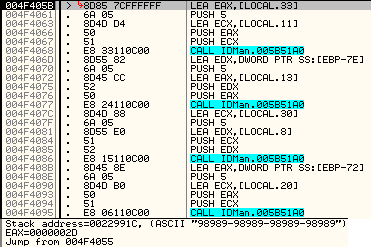


Bueno aquí nos dice que no confundamos el 0 con la O, la I con el 1 a la hora de escribir nuestro serial, que es mejor cortar y pegar, uffff todavía tiene el descaro de echarme la culpa eso mismo fue lo que hice ;-) mañana me le quejo a ese colega, me va a tener que escuchar yo le voy a decir unas cuantas cosas…… bueno sigamos.

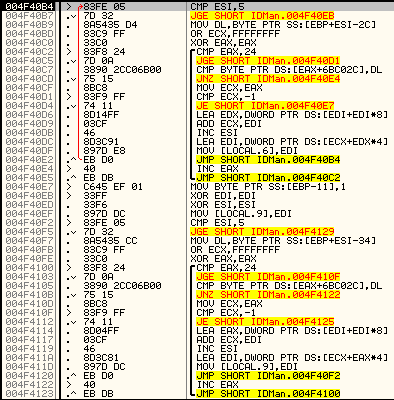
Vamos agregarle un 9 y veremos que venceremos ese primer obstáculo seguido vemos esto:



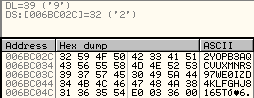
Lo 1er que hace es mover el 38 que es un 8 de nuestro serial para CL, 2 líneas abajo vemos que mueve 2D que es un guion a AL y seguido compara nuestro 38 con 2D, en el DUMP vemos que nuestro 38 es el 6to carácter de nuestro serial, pongamos un Bp y en ese JUMP y quitemos todos los demás, ahora nuestro serial va cogiendo forma para que venza este otro obstáculo tiene que ir así 98989-98989898989898989 demos Run, modificamos nuestro serial y Ok se detiene en el Bp y ya no nos amenaza con brincar pasemos con F8 y hace una comparación similar a la de arriba con otro 8 si lo buscamos en el DUMP y contamos la posición vemos que esta 6 caracteres después del guion que pusimos, uhmm me huele a que está verificando que el serial este formateado, deduciendo, dijimos que el 6to carácter al comenzar el serial tiene que ser un guion y que el 6to después tiene que ser otro guión siguiendo esta lógica vamos a hacer lo mismo con los otros caracteres que cumplan esa condición y nos quedaría así 98989-98989-98989-98989, escribámoslo y demos Run para en el Bp tracemos con F8 y ya no brinca, sigamos con F8 y vemos esto



Recuerden lo que les dije de observar bien para entender cómo funciona el programa y porque muchas líneas aquí se repiten en otros programas, bueno actualmente LOCAL.33 tiene la dirección de memoria que almacena nuestro serial, vayamos traceando y pasemos ese 1er CALL y observen que EAX tiene nuestros primeros 5 caracteres del serial (esto lo deduje porque en un momento probé con un serial ABCDE-FGHIJ… y me percate de ello) en los otros CALL hará lo mismo, tomara los otros 5 caracteres que están entre los guiones para guardarlos, sigamos traceando y llegamos a:



Si bajan con el mouse podrán ver que este patrón se repite 2 veces más, les cuento que automáticamente cuando vi ese CMP que es llamado desde abajo como podemos observar me dije esto es un ciclo y es que ya este patrón lo he visto en otros programas que he estudiado, ESI será el contador que se ira incrementado hasta ser mayor que 5 veamos las operaciones que realiza, me detendré a explicar las líneas que considero las más relevantes, empecemos, F8 llegamos a 004F40B9 MOV DL, BYTE PTR SS:[EBP+ESI-2C], mueve el 1er carácter de los 1eros 5 caracteres del serial después 004F40C7 CMP BYTE PTR DS:[EAX+6BC02C], DL compara nuestro 9 con un 2 sigamos a este último en el DUMP



Uhmm esto pinta raro una cadena 2YOPB3AQCVUXMNRS97WE0IZD4KLFGHJ8165T, continuemos a ver dónde llega todo esto, F8 salta e incrementa EAX vuelve a comparar pero esta vez con la Y repite lo mismo con la O entonces deducimos que cuando el valor de EAX se igual que 39 (9) no brincara, más abajo vemos el 9 en la 3er fila, sigamos hasta que lleguemos a él cuándo esto sucede EAX vale 10 (16) seguido hace un MOV ECX, EAX seguimos con un LEA que almacena cero y después una suma ADD ECX, EDI incrementamos ESI continuamos con LEA EDI, DWORD PTR DS:[ECX+EDX\*4] equivale a 10 + 0 \* 4 = 10, después lo guarda en LOCAL.6 y empieza el ciclo nuevamente.

Ahora buscara la posición del 8 en la cadena si contamos lo encontramos en la 31 (1F en hexadecimal), viene el LEA EDX, DWORD PTR DS: [EDI + EDI \* 8] recuerden que EDI se quedó valiendo 10 así que equivale a 10 + 10 \* 8 en el panel nos muestra ya el resultado 90, proseguimos con la suma de la 1era posición del 9 con la 2da posición del 8 equivale a 1F + 10 = 2F, incremento ESI, ahora LEA EDI, DWORD PTR DS:[ECX+EDX\*4] equivale a 2F + 90 del LEA pasado = 26F y repite hasta los 5 caracteres

Más abajo sucede parecido con cada parte del serial, cambia ligeramente los cálculos es importante que vean en donde guarda cada uno, para los primeros 5 lo hace en LOCAL.6, para los segundos LOCAL.9, tercero en EBX y cuarto en EDI.

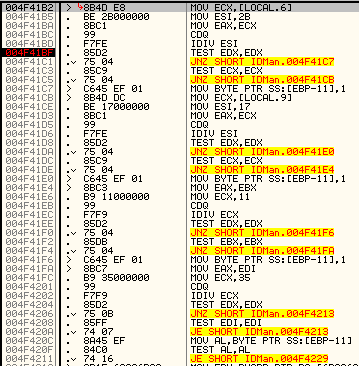
Cuando terminamos todos los cálculos llegamos aquí



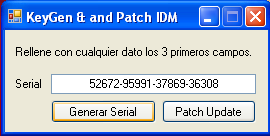
Lo comente para hacerlo más fácil de entender, en ECX está el resultado final del cálculo con los primeros 5 caracteres en mi caso 01E1DEEE (31579886 en decimal), o sea que dividirá este valor entre 2B (43 decimal) y el resto que se guardara en EDX tiene que ser 0 para que no salte y continúe con los demás, por supuesto aquí me dio como resultado 29 (41 decimal).

Con los otros es lo mismo solo que cambia el dividendo pero al final siempre tiene que dar como resto cero y esa es la magia de todo este asunto.

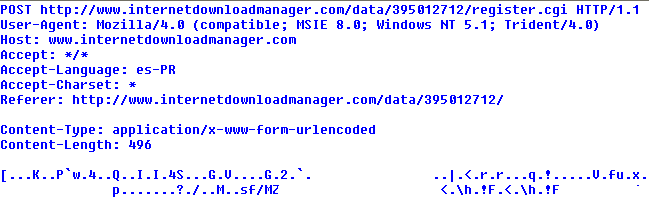
Más abajo verán una llamada a las API’s RegCreateKeyExA y RegSetValueExA crea las claves y guarda los valores con nuestros datos en el registro.



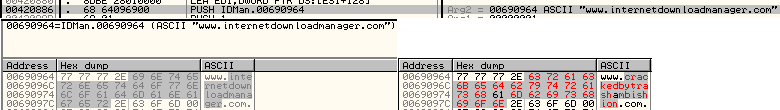
Les dejo un simple KeyGen que me genera serial numéricos me daba FIACA uno más completo, si quieren hagan uno que sea alfanumérico. Lo que hice fue generar números aleatorios entre 10000-99999 dividirlo entre cada dividendo y cuando me daba cero lo voy guardando y concatenando hasta hacer un serial valido.



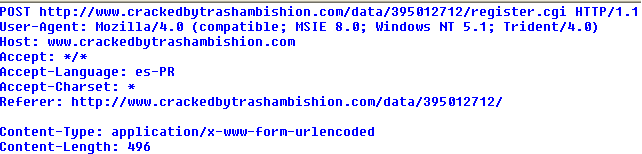
Ahora tenemos la otra problemática que cuando a él le da la gana verifica si hay nuevas versiones y de paso mira si nuestra licencia es legítima, lo que haremos será averiguar la dirección HTTP con la que él se comunica con su servidor para enviarle los datos una vez que la tengamos vamos a modificarla, esto para hacerlo interesante porque en verdad hay maneras más sencillas. Usaremos la herramienta SmartSniff para capturarla y ahora tendremos que iniciar el IDM puede que no se salga al momento pero luego de un rato



Ok la meta aquí será cambiar el [www.internetdownloadmanager.com](http://www.internetdownloadmanager.com) por [www.crackedbytrashambishion.com](http://www.crackedbytrashambishion.com) para nada orgulloso, jajaja XD, veamos busquemos la URL a modificar en la 3era coincidencia caigo aquí



Ya tengo seleccionado lo que voy a cambiar y al lado como queda, guardemos los cambios y probémoslo a ver qué sucede, activemos el Sniffer para ver que captura.



Wuala hemos vencido a medias, ya sabemos lo que tenemos que modificar ahora hagamos nuestro KeyGen & Patch 100% redistribuible y completo, bueno ahora nos apoyaremos del UltraEdit vamos a buscar [www.internetdownloadmanager.com](http://www.internetdownloadmanager.com) marcamos la casilla FindAscii



Llegamos rápidamente a la zona caliente

C:\Documents and Settings\Aismel\Mis documentos\Mis imágenes\1.bmp

Este es el que buscamos con seguridad, porque? si se fijan en el DUMP del Olly verán que después de la URL está el nombre de la página testing.html al igual que en la imagen, bueno vamos a empezar a modificar desde 290968 para nada mas modificar lo que está entre los puntos, así me evito escribir de más aquí ya les invito que revisen el proyecto que está en .Net para ver la modificación, les enseño una captura ya modificado del programa.



Ojala y les haya gustado, tanto como a mi hacerlo.

PD: Algo curioso haciendo las pruebas para capturar la solicitud POST y GET tuve que configurar un Proxy el CCPROXY la última versión quizás lo hayan utilizado, lo hice para engañar al IDM de que estaba en una RED real, el punto es que ese programa también hay que registrarlo, sino nada más podrán conectarse 3 usuarios, será mi próximo tutorial si gustan, ustedes digan.

Salu2, critiquen…. ;-) de mas esta decir los agradecimientos.