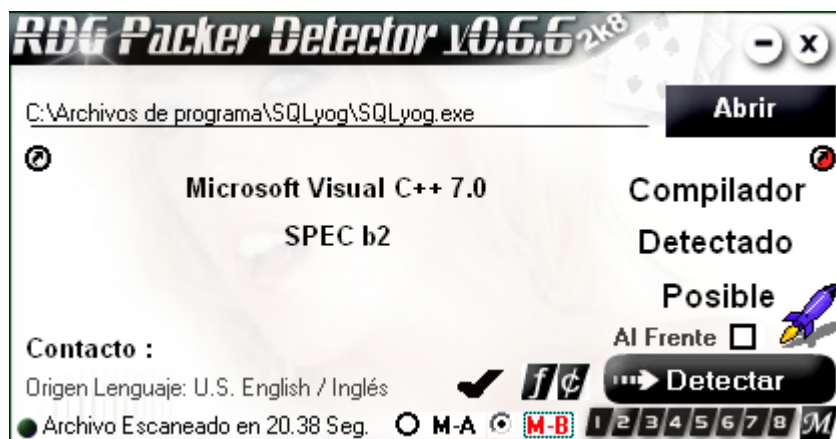


Victima:	Sqlyog
Version:	8.32
Descripción:	Frontd para administrar Mysql
Url:	http://www.webyog.com
Protección:	Serial
Objetivo:	Encontrar serial para activarlo
Dificultad:	Newbie
Herramientas:	RDG, Ollydbg
Cracker:	Dkool
Fecha:	22/04/10

Este es mi segundo tute, por la escuela y el trabajo pues estaba muy ocupado, pero bueno vamos al grano haber que sale jeje.

Como primer paso seria checar el programa si no esta empacado, como siempre lo hago lo abrimos con el RDG.



Y vemos que esta en Visual C++ y el RDG detecta SPEC b2. mmm pues no habia visto ese SPEC b2, pero sigamos a ver que pasa abrimos el programa y nos pide la ventana de registro, para ello ingresamos datos y haber que pasa pero pues como era esperarse nos manda error como se muestra.



Bien ahora le toca a ollydbg, lo abrimos y nos vamos alas referenced Strings no encontramos el mensaje de chico malo, mmmmm entonces vamos por otro metodo, bien damos F9 para correr el programa y cuando nos pida los datos de registro los llenamos y damos click en el boton de register nos sale el mensaje de error y antes de dar click en ok del chico malo nos vamos a ollydbg y presionamos pause y enseguida damos click en la letra k como se muestra.



Terminando esos 2 pasos caemos en los mensajes de error como se muestra.

Address	Stack	Procedure / arguments	Called from	Frame
0012F45C	7E399418	Includes ntdll.KiFastSystemCallRet	USER32.7E399416	0012F490
0012F460	7E3A770A	USER32.WaitMessage	USER32.7E3A7705	0012F490
0012F494	7E3A49C4	USER32.7E3A757B	USER32.7E3A49BF	0012F490
0012F4BC	7E3A4A06	USER32.7E3A490E	USER32.7E3A4A01	0012F4B8
0012F4DC	7E3A47EA	USER32.DialogBoxIndirectParamAorW	USER32.7E3A47E5	0012F4D8
0012F500	0044E2E3	USER32.DialogBoxParamW	SQLyog.0044E2D0	0012F4FC
0012F504	01840000	hInst = 01840000		
0012F508	0000038D	pTemplate = 38D		
0012F50C	001607D0	hOwner = 001607D0 (* Register SQL		
0012F510	0044E200	DlgProc = SQLyog.0044E200		
0012F514	0012F528	lParam = 0012F528		
0012F518	0044E3D6	SQLyog.0044E2C0	SQLyog.0044E3D6	0012F594
0012F558	0044E489	SQLyog.0044E2F0	SQLyog.0044E484	0012F594
0012F55C	001607D0	Arg1 = 001607D0		
0012F560	00000417	Arg2 = 00000417		

Y en la dirección 0012F518 damos doble click como muestra la imagen, dando dbl clic caemos en esa dirección como se muestra.

0044E3BF	E8 0C980B00	CALL 00507BD0	
0044E3C4	83C4 18	ADD ESP,18	
0044E3C7	EB 23	JMP SHORT 0044E3EC	
0044E3C9	85C0	TEST EAX,EAX	
0044E3CB	75 1F	JNZ SHORT 0044E3EC	
0044E3CD	8B4424 3C	MOV EAX,DWORD PTR SS:[ESP+3C]	
0044E3D1	50	PUSH EAX	
0044E3D2	8D4C24 0C	LEA ECX,DWORD PTR SS:[ESP+4C]	
0044E3D6	E8 E5FEFFFF	CALL 0044E2C0	
0044E3DB	EB 0F	JMP SHORT 0044E3EC	
0044E3DD	8B4C24 3C	MOV ECX,DWORD PTR SS:[ESP+3C]	Case 2 of switch 0044E32C
0044E3E1	6A 00	PUSH 0	
0044E3E3	51	PUSH ECX	
0044E3E4	E8 E7970B00	CALL 00507BD0	
0044E3E9	83C4 08	ADD ESP,8	
0044E3EC	8BD7	MOV EDI,EDI	Default case of switch 0044E32C

Y bien ya estando ahí vamos subiendo hasta donde inicia la función o en esta dirección 0044E2F0, y mas abajo esta una CALL 0044DFA0 y abajo de esta call esta una comparación CMP AX,1 como que suena bien esta call y le damos un bp con F2. Como se muestra.

Address	Hex dump	Disassembly	Comment
0044E2F0	6A FF	PUSH -1	
0044E2F2	68 10CB6700	PUSH 0067CB10	SE handler installation
0044E2F7	64:A1 000000	MOV EAX,DWORD PTR FS:[0]	
0044E2FD	50	PUSH EAX	
0044E2FE	64:8925 0000	MOV DWORD PTR FS:[0],ESP	
0044E305	83EC 24	SUB ESP,24	
0044E308	50	PUSH ESI	
0044E309	50	MOV ESI,ECX	
0044E30B	57	PUSH EDI	
0044E30C	8D4C24 10	LEA ECX,DWORD PTR SS:[ESP+10]	
0044E310	E8 9B071300	CALL 0057EAB0	
0044E315	C74424 34 00	MOV DWORD PTR SS:[ESP+34],0	
0044E31D	C74424 0C 00	MOV DWORD PTR SS:[ESP+C],0	
0044E325	8B7C24 40	MOV EDI,DWORD PTR SS:[ESP+40]	
0044E329	0FB7C7	MOVZX EAX,DI	
0044E32C	83E8 02	SUB EAX,2	Switch (cases 2..417)
0044E32F	C64424 34 01	MOV BYTE PTR SS:[ESP+34],1	
0044E334	0F84 A3000000	JE 0044E3D0	
0044E33A	2D 15040000	SUB EAX,415	
0044E33F	0F85 A7000000	JNE 0044E3EC	Case 417 of switch 0044E32C
0044E345	8BCE	MOV ECX,ESI	
0044E347	E8 E4FCFFFF	CALL 0044DFA0	
0044E34C	83F8 01	CMP EAX,1	
0044E34F	75 78	JNZ SHORT 0044E3C9	
0044E351	8B0D A8848000	MOV ECX,DWORD PTR DS:[8884A8]	
0044E357	8B46 18	MOV EAX,DWORD PTR DS:[ESI+18]	
0044E35A	81C1 38030000	ADD ECX,338	
0044E360	83F8 01	CMP EAX,1	
0044E363	75 12	JNE SHORT 0044E377	
0044E365	E8 F6851400	CALL 00659690	
0044E36A	50	PUSH EAX	
0044E36B	68 68506900	PUSH 00695068	ASCII "Thank you for registering SQLyog %s - MySQL 6

Bien ya que le dimos un bp a la call reiniciamos el olly CTRL + F2, y nuevamente damos F9 para correr el programa e ingresamos los datos de registro y le damos ok al

chico malo y caemos en la call que ingresamos el bp, enseguida de esto entramos a ella con F7, ya estando ahí vamos traceando con F8 hasta llegar a una call 00665C66 ya abajo un JNZ esa call significa junto con el salto si no ingresamos el numero de registro se habilita el salto y nos manda al call de error. Pero eso no pasara ya que nosotros ingresamos jeje, y bien pasemos ese JNZ y sigamos traceando hasta llegar esta CALL 0044DAD0. como se muestra.

0044E151	. E8 4A031300	CALL 0057E4A0
0044E156	. 53	PUSH EBX
0044E157	. 8D4C24 30	LEA ECX,DWORD PTR SS:[ESP+30]
0044E15B	. E8 30001300	CALL 0057E190
0044E160	. 50	PUSH EAX
0044E161	. 8BCE	MOV ECX,ESI
0044E163	. E8 68F9FFFF	CALL 0044DAD0
0044E168	. 8BF8	MOV EDI,EAX
0044E16A	. 83FF 05	CMP EDI,5
0044E16D	. 75 28	JNZ SHORT 0044E197

Entramos F7

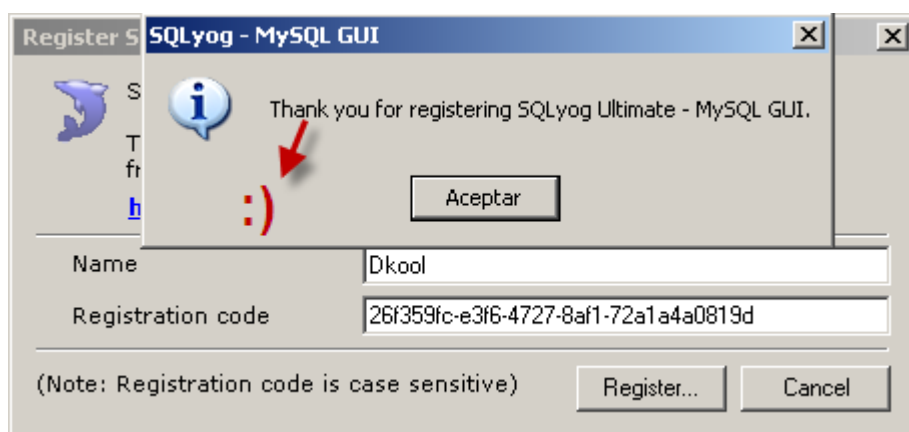
Entramos a es call con F7 y Woowww creo q encontramos el tesoro jejeje nos aparecen 3 seriales distintos como se muestra.

Address	Hex dump	Disassembly	Comment
0044DAD0	55	ENTER ESP,1	
0044DAD1	8B7424 08	MOV ESI,DWORD PTR SS:[ESP+8]	
0044DAD5	85F6	TEST ESI,ESI	
0044DAD7	75 09	JNZ SHORT 0044DAE2	
0044DAD9	B8 05000000	MOV EAX,5	
0044DADE	5E	POP ESI	
0044DADE	C2 0400	RET 4	
0044DAE2	68 58DB6800	PUSH 0068DB58	Unicode "26f359fc-e3f6-4727-8af1-72a1a4a0819d"
0044DAE7	56	PUSH ESI	
0044DAE8	E8 B8922100	CALL 00666DA5	
0044DAED	83C4 08	ADD ESP,8	
0044DAF0	85C0	TEST EAX,EAX	
0044DAF2	75 09	JNZ SHORT 0044DAFD	
0044DAF4	B8 01000000	MOV EAX,1	
0044DAF9	5E	POP ESI	
0044DAFA	C2 0400	RET 4	
0044DAFD	68 B0DB6800	PUSH 0068DBB0	Unicode "2e7d53db-9bd4-4673-a0ff-937e172d0a34"
0044DB02	56	PUSH ESI	
0044DB03	E8 9D922100	CALL 00666DA5	
0044DB08	83C4 08	ADD ESP,8	
0044DB0B	85C0	TEST EAX,EAX	
0044DB0D	75 09	JNZ SHORT 0044DB18	
0044DB0F	B8 02000000	MOV EAX,2	
0044DB14	5E	POP ESI	
0044DB15	C2 0400	RET 4	
0044DB18	68 10DC6800	PUSH 0068DC10	Unicode "17cb5c23-8653-418f-b81b-5582c7a5a2d7"
0044DB1D	56	PUSH ESI	
0044DB1E	E8 82922100	CALL 00666DA5	
0044DB23	83C4 08	ADD ESP,8	
0044DB26	F7D8	NEG EAX	
0044DB28	1BC0	SBB EAX,EAX	
0044DB2A	83E0 02	AND EAX,2	
0044DB2D	83C0 03	ADD EAX,3	
0044DB30	5E	POP ESI	
0044DB31	C2 0400	RET 4	

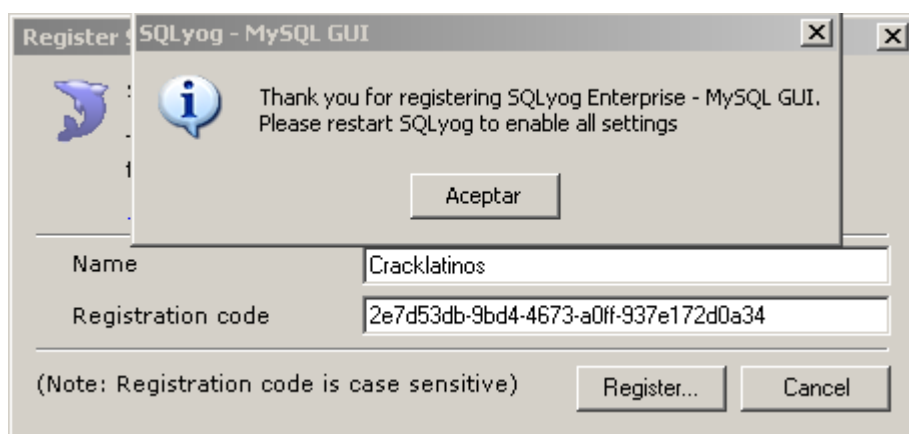
Bien si queremos podemos ir traceando para ver como se compara el serial que ingresamos por ejemplo.. 1234567890 con los 3 de ahí. Lo podemos ver en el stack como se muestra.

0012F1A4	01684FF8	Unicode "1234567890"
0012F1A8	0068DB58	Unicode "26f359fc-e3f6-4727-8af1-72a1a4a0819d"
0012F1AC	0012F95C	
0012F1B0	0044E168	RETURN to SQLyog.0044E168 from SQLyog.0044DAD0
0012F1B4	01684FF8	Unicode "1234567890"
0012F1B8	00000417	
0012F1BC	0012F95C	
0012F1C0	00000000	
0012F1C4	00150000	
0012F1C8	00000000	
0012F1CC	01CB0CA8	ASCII "Dkool"

Ahora viene probar esos seriales sin son los correctos para ello copiare el primero, bien vamos haber, abrimos ejecutamos el programa y probemos.



Bien si funciona jeje muy bien. Pues como observamos el name no importa para los seriales, mmm ingresemos con otro name y ahora probemos el segundo serial jeje vamos haber si funciona.



Perfecto si funciona con cualquier name y con el segundo serial jejee era de esperarse jeje. Ahora mi programa queda registrado a **CRACKLATINOS**. Me parece excelente que quede con ese nombre registrado por que sin ellos no estaría haciendo estos tutoriales, sencillos pero efectivos jeje. Bueno un saludo a todos los Cracklatinos a ZELT@ y al buen master ricardo narvaja.. Bien me despido y prometo hacer otro tutorial mientras sigo tomando unas cervezas jeje por que esta duro la calor jeje.

