

SUGARBYTES PRODUCTS - CRACK TUTORIAL

by Rank aka Tracer

PATCHING SUGARBYTES IN 3 STEPS

[IN THIS CASE 'GRAINDEAD' BUT NEAR ALL IS SAME ON EACH OTHER, IF YA CHECK THIS)

1/3

SEARCH: "NoSilentOpt" IN STRING REFERENCES
(NEAR IN EVERY RELEASE, ELSE BEGIN WITH 2/3 & READ AT THE END HOW TO *Alternate*)

0F9486 9A5B6401	SETB BYTE PTR DS:[RSI+1645B9A]	<----- [PATCH THIS, "94" TO "95"]
48:8B8D 580F0000	mov rcx,qword ptr ss:[rbp+F58]	
BA 80000000	mov edx,80	
E8 4D8D0500	call <graindad.sub_7FFFC6A9DCE0>	
C786 905B6401 00000000	mov dword ptr ds:[rsi+1645B90],0	
F3:0F1086 40640000	movss xmm0,dword ptr ds:[rsi+6440]	
F3:0F5905 EF1A1400	mulss xmm0,dword ptr ds:[7FFFC6B86]	
F3:0F2CC0	cvtts2si eax,xmm0	
3986 405B6401	cmp dword ptr ds:[rsi+1645B40],eax	
74 10	je graindad.7FFFC6A44FC9	
8986 405B6401	mov dword ptr ds:[rsi+1645B40],eax	
C786 445B6401 00000000	mov dword ptr ds:[rsi+1645B44],0	
C786 485B6401 E29A0A3	mov dword ptr ds:[rsi+1645B48],390	
48:8D86 C0640000	lea rax,qword ptr ds:[rsi+64C0]	
48:8986 505B6401	mov qword ptr ds:[rsi+1645B50],rax	
C785 9C0F0000 00000000	mov dword ptr ss:[rbp+F9C],0	
48:8D0D EE621A00	LEA RCX,QWORD PTR DS:[7FFFC6BEB2E0]	<----- [1st YOU LAND HERE] -> "NoSilentOpt"

2/3

SEARCH "serial" IN STRING REFERENCES (CHOOSE THE FIRST, CAUSE THERE ARE MORE) IF DONE:

55	push rbp	<----- [PATCH THIS, "55 56 57" TO "B0 01 C3"]
56	push rsi	
57	push rdi	
53	push rbx	
48:81EC 78010000	sub rsp,178	
48:8DAC24 80000000	lea rbp,qword ptr ss:[rsp+80]	
48:C785 F0000000 FEFF	mov qword ptr ss:[rbp+F0],FFFFFFFF	
48:89CB	mov rbx,rcx	
C785 EC000000 00000000	mov dword ptr ss:[rbp+EC],0	
48:C785 E0000000 0000	mov qword ptr ss:[rbp+E0],0	
48:B8 536F66747761726	mov rax,6572617774666F53	
48:8945 D0	mov qword ptr ss:[rbp-30],rax	
66:C745 D8 5C00	mov word ptr ss:[rbp-28],5C	
48:8D15 4FD91300	lea rdx,qword ptr ds:[7FFFC6BD87A0]	00007FFFC6BD87A0:"Sugar Bytes"
48:8D7D D0	lea rdi,qword ptr ss:[rbp-30]	
48:89F9	mov rcx,rdi	
E8 33810D00	call <graindad.sub_7FFFC6B72F90>	
48:89F9	mov rcx,rdi	
E8 7B830D00	call <graindad.sub_7FFFC6B731E0>	
66:C74405 D0 5C00	mov word ptr ss:[rbp+rax-30],5C	
48:89F9	mov rcx,rdi	
E8 6C830D00	call <graindad.sub_7FFFC6B731E0>	
C74405 D0 474C4F42	mov dword ptr ss:[rbp+rax-30],424F	
C74405 D3 42414C00	mov dword ptr ss:[rbp+rax-2D],4C41	
48:8D85 DC000000	lea rax,qword ptr ss:[rbp+DC]	
48:894424 40	mov qword ptr ss:[rsp+40],rax	
48:8D85 E0000000	lea rax,qword ptr ss:[rbp+E0]	
48:894424 38	mov qword ptr ss:[rsp+38],rax	
48:C74424 30 00000000	mov qword ptr ss:[rsp+30],0	
C74424 28 3F000F00	mov dword ptr ss:[rsp+28],F003F	
C74424 20 00000000	mov dword ptr ss:[rsp+20],0	
48:C7C1 01000080	mov rcx,FFFFFFFF80000001	
48:89FA	mov rdx,rdi	
45:31C0	xor r8d,r8d	
45:31C9	xor r9d,r9d	
FF15 75751500	call qword ptr ds:[<&RegCreateKeyE	
85C0	test eax,eax	
74 0D	je graindad.7FFFC6A9AEDC	
48:C785 E0000000 0000	mov qword ptr ss:[rbp+E0],0	
EB 45	jmp graindad.7FFFC6A9AF21	
48:8B8D E0000000	mov rcx,qword ptr ss:[rbp+E0]	
48:85C9	test rcx,rcx	
74 39	je graindad.7FFFC6A9AF21	
C745 D0 04000000	mov dword ptr ss:[rbp-30],4	
C785 DC000000 04000000	mov dword ptr ss:[rbp+DC],4	
48:897C24 28	mov qword ptr ss:[rsp+28],rdi	
48:8D85 EC000000	lea rax,qword ptr ss:[rbp+EC]	
48:894424 20	mov qword ptr ss:[rsp+20],rax	
48:8D15 CD121500	lea rdx,qword ptr ds:[7FFFC6BEC1DE]	<----- [YOU LAND HERE] "serial"
4C:8D8D DC000000	lea r9,qword ptr ss:[rbp+DC]	
45:31C0	xor r8d,r8d	
FF15 2F751500	call qword ptr ds:[<&RegQueryValue	
48:8D0D B6121500	lea rcx,qword ptr ds:[7FFFC6BEC1DE]	<----- [ANOTHER "serial" STRING]

55	push rbp	<-----[(B)...TILL HERE\CHECK REF. TO THE
56	push rsi	RELATED CALL (IS JUST 1) E8 A6BEF4FF]
57	push rdi	
53	push rbx	
48:83EC 68	sub rsp,68	
48:8D6C24 60	lea rbp,qword ptr ss:[rsp+60]	
48:C745 00 FFFFFFFF	mov qword ptr ss:[rbp],FFFFFFFF	
48:89CB	mov rbx,rcx	
48:8B45 50	mov rax,qword ptr ss:[rbp+50]	
48:8D35 052F1500	lea rsi,qword ptr ds:[<&sub_7FFFC6	
48:8931	mov qword ptr ds:[rcx],rsi	
4C:8D81 B0120000	lea r8,qword ptr ds:[rcx+12B0]	
48:8D3D 14321500	lea rdi,qword ptr ds:[<&sub_7FFFC6	
48:89B9 B0120000	mov qword ptr ds:[rcx+12B0],rdi	
48:894424 20	mov qword ptr ss:[rsp+20],rax	
C74424 30 00000000	mov dword ptr ss:[rsp+30],0	
48:C74424 28 00000000	mov qword ptr ss:[rsp+28],0	
E8 BA010000	call <graindad.sub_7FFFC6A36230>	
48:8933	mov qword ptr ds:[rbx],rsi	
48:89BB B0120000	mov qword ptr ds:[rbx+12B0],rdi	
C783 B8120000 01000000	mov dword ptr ds:[rbx+12B8],1	
48:C783 C6120000 0000	mov qword ptr ds:[rbx+12C6],0	
48:C783 C0120000 0000	mov qword ptr ds:[rbx+12C0],0	
66:C783 CE120000 0100	mov word ptr ds:[rbx+12CE],1	
48:895D F8	mov qword ptr ss:[rbp-8],rbx	
C683 D0120000 00	mov byte ptr ds:[rbx+12D0],0	
48:833D B4EC1D00 10	cmp qword ptr ds:[7FFFC6C14D70],10	
72 09	jb graindad.7FFFC6A360C7	
48:8B15 93EC1D00	mov rdx,qword ptr ds:[7FFFC6C14D58	
EB 07	jmp graindad.7FFFC6A360CE	
48:8D15 8AEC1D00	lea rdx,qword ptr ds:[7FFFC6C14D58	
48:8D4D E0	lea rcx,qword ptr ss:[rbp-20]	
45:31C0	xor r8d,r8d	
E8 26120600	call <graindad.sub_7FFFC6A97300>	
48:8B45 F8	mov rax,qword ptr ss:[rbp-8]	
4C:8D80 C8120000	lea r8,qword ptr ds:[rax+12C8]	
48:8D15 F2601B00	lea rdx,qword ptr ds:[7FFFC6BEC1DE	
48:8D4D E0	lea rcx,qword ptr ss:[rbp-20]	
E8 CB4A0600	call <graindad.sub_7FFFC6A9ABC0>	
48:8D4D E0	lea rcx,qword ptr ss:[rbp-20]	
E8 52130600	call <graindad.sub_7FFFC6A97450>	
C745 F4 00000000	mov dword ptr ss:[rbp-C],0	
48:833D 63EC1D00 10	cmp qword ptr ds:[7FFFC6C14D70],10	
72 09	jb graindad.7FFFC6A36118	
48:8B15 42EC1D00	mov rdx,qword ptr ds:[7FFFC6C14D58	
EB 07	jmp graindad.7FFFC6A3611F	
48:8D15 39EC1D00	lea rdx,qword ptr ds:[7FFFC6C14D58	
48:8D4D E0	lea rcx,qword ptr ss:[rbp-20]	
45:31C0	xor r8d,r8d	
E8 D5110600	call <graindad.sub_7FFFC6A97300>	
48:8D15 FE6F1B00	lea rdx,qword ptr ds:[7FFFC6BED130	<-----[(A) YOU LAND HERE "serialhide" :
		GO ABOVE...
^^ IF DONE THIS^..^^		
E8 A6BEF4FF	call <graindad.sub_7FFFC6A36020>	<-----[(C) THEN WE LAND IN THIS RELATED CALL]
48:8B85 90440000	mov rax,qword ptr ss:[rbp+4490]	
48:8B8D 58440000	mov rcx,qword ptr ss:[rbp+4458]	
48:8988 200E0000	mov qword ptr ds:[rax+E20],rcx	
48:8B80 50010000	mov rax,qword ptr ds:[rax+150]	
83B8 945B6401 00	cmp dword ptr ds:[rax+1645B94],0	
74 0A	je graindad.7FFFC6AEA1A9	<-----[(D) WE PATCH THIS FROM "74" TO "75"]
48:8B01	mov rax,qword ptr ds:[rcx]	
31D2	xor edx,edx	
FF50 58	call qword ptr ds:[rax+58]	
EB 0B	jmp graindad.7FFFC6AEA1B4	
8B95 28440000	mov edx,dword ptr ss:[rbp+4428]	
E8 3CC2F4FF	call <graindad.sub_7FFFC6A363F0>	
48:8BBD 90440000	mov rdi,qword ptr ss:[rbp+4490]	

PATCHING DONE - FINALLY ADD A FAKE SERIAL FOR EACH PLUGIN INSIDE REGISTRY AS EXAMPLE:
 Registry\HKEY_CURRENT_USER\Software\Sugar Bytes\Graindad\serial=2f951f6

MAY ALSO SET THIS VALUE TO ZERO:

Registry\HKEY_CURRENT_USER\Software\Sugar Bytes\Graindad\isDemo=0

YOU'RE TOTALLY DONE.

EXTRA : IF THE 1ST PATCH (1/3) NOT WORX - HERE'S THE ALTERNATE :

SEARCH: "isDemoVersion" , if done :

80BF 33D0AB00 00	cmp byte ptr ds:[rdi+ABD033],0	<----- right click and 'Find references to Constand'
74 18	je effectrix2.7FFE7A5899B0	
48:8D15	lea rdx,qword ptr ds:[7FFE7A908848]	00007FFE7A908848:"isDemoVersion" <----- YOU LAND HERE

...that list's any related, choose the one who contains 'SET BYTE PTR ...' & change byte from "94" TO "95"

NOW YOU'RE ALSO TOTALLY DONE.