

2017 03



Sage 2.0

2017 03 20



.		3
.	Sage	4
2.1		4
2.2	-	4
2.3	-	4
.	Sage	

■

Sage

CryLocker

Sage

Cerber Locky Tesla Spora

Sage

zip

zip

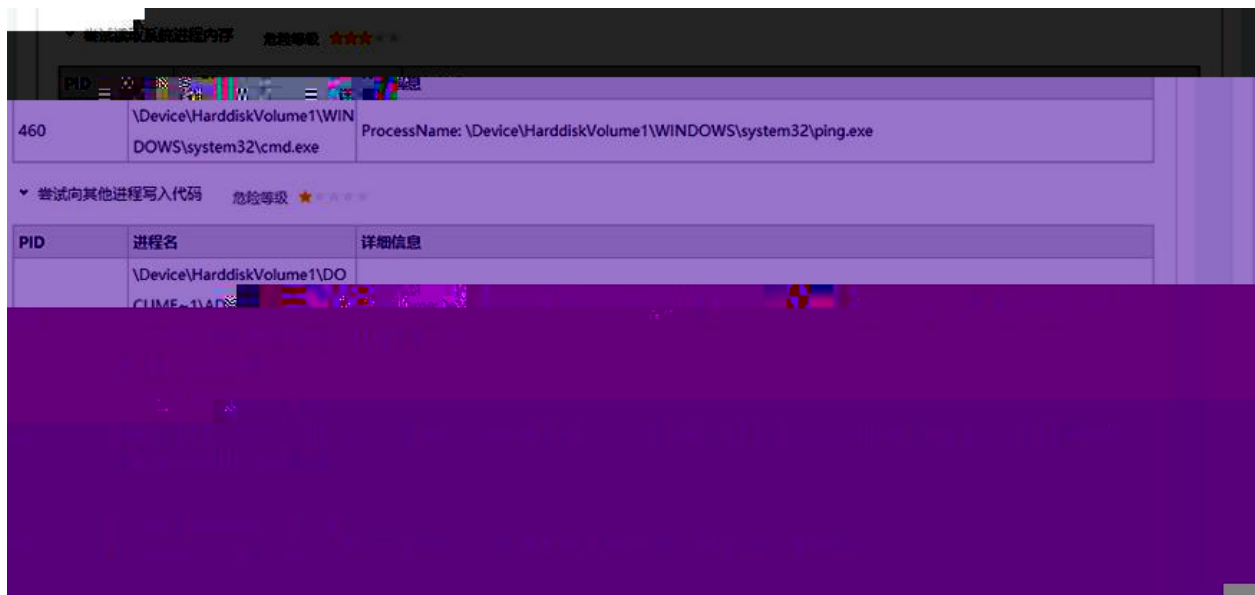
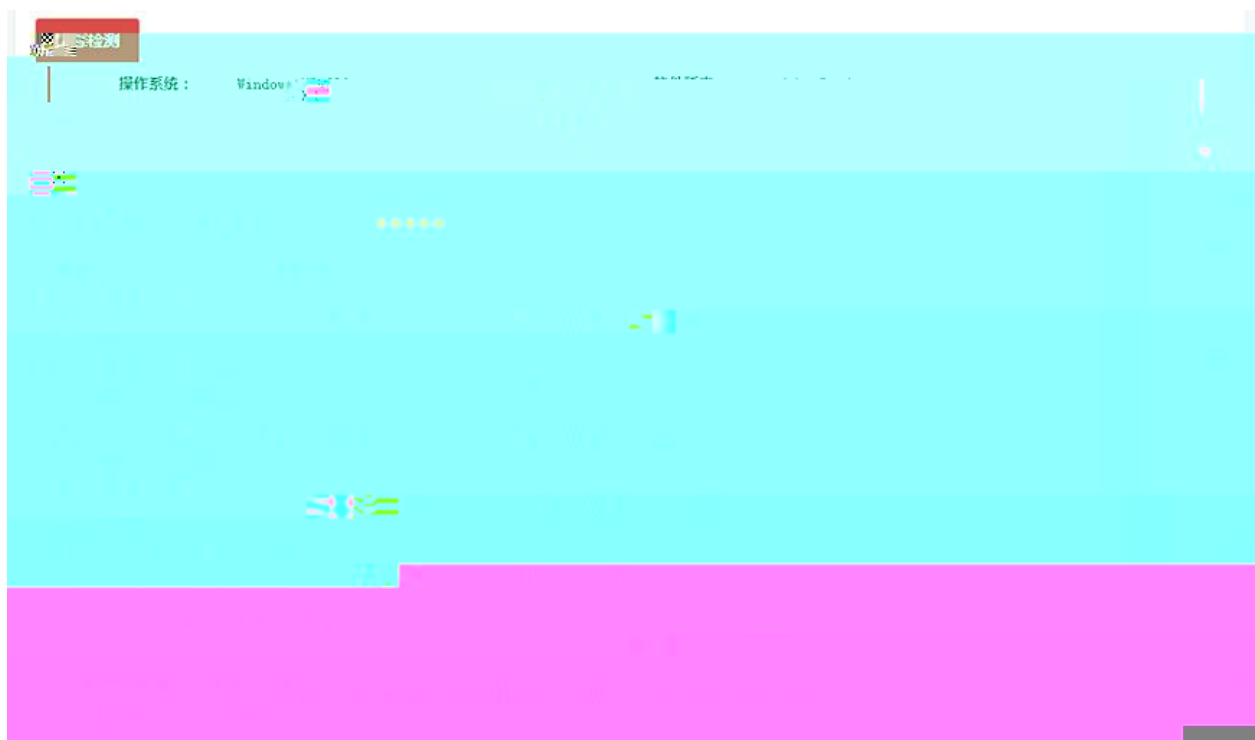
Word

· **Sage**

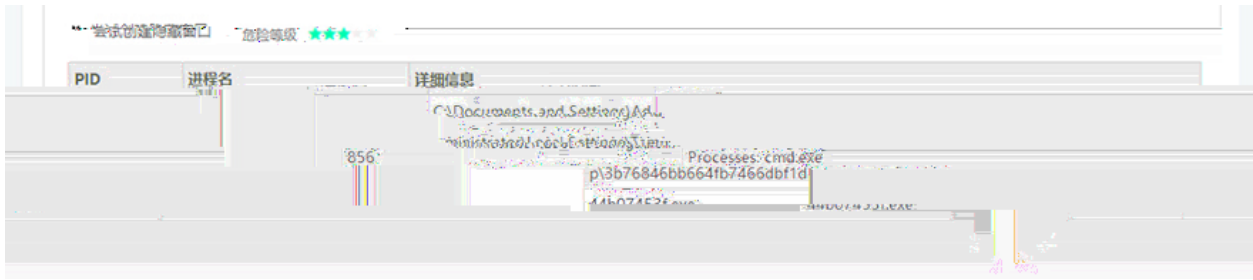
2.1

2.2

-





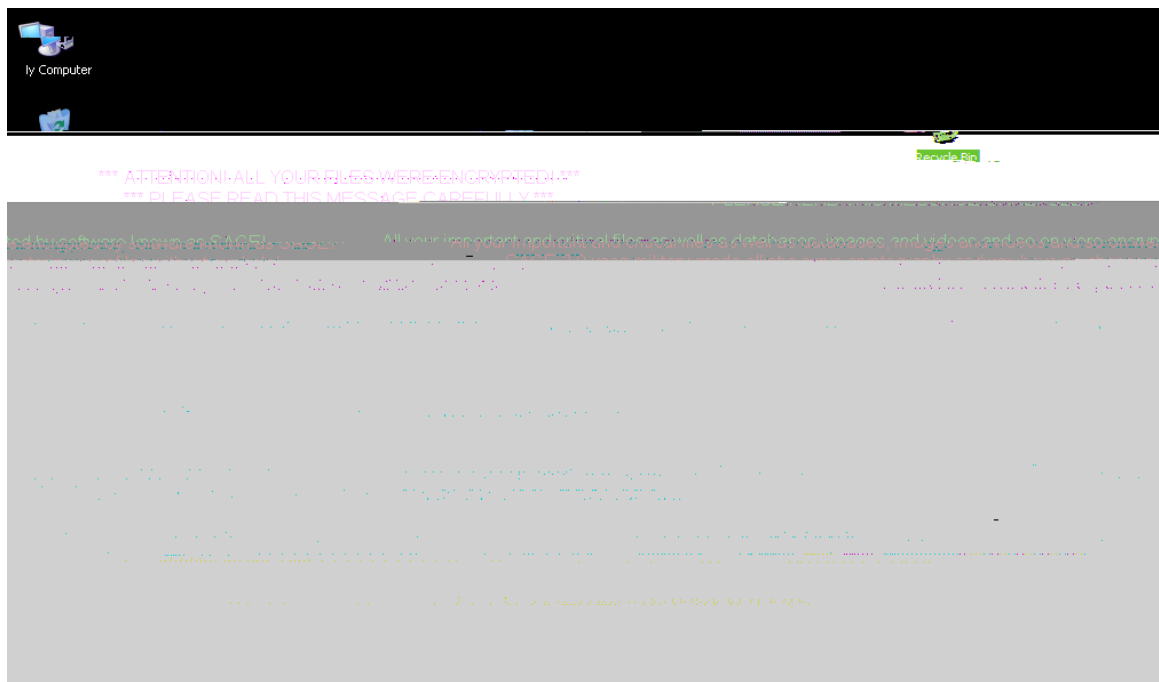


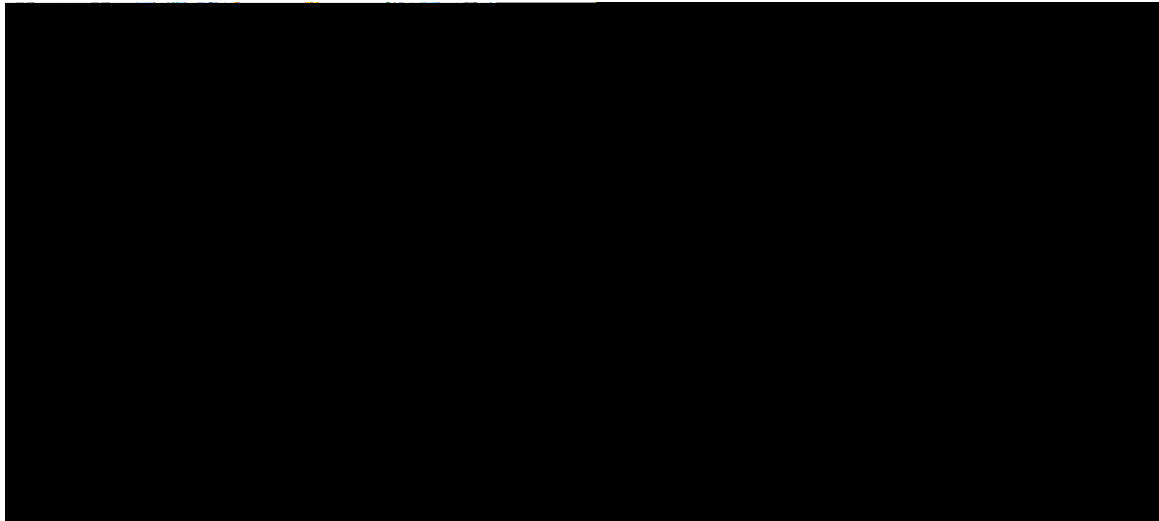
3. Sage

3.1

345KB

3.2





1 KB SAGE File

2 KB SAGE File

1 KB SAGE File

1 KB SAGE File

2 KB SAGE File

1 KB SAGE File

3.3

```

graph TD
    Entry(( )) --> Node1["call sub_4054A0  
test eax, eax  
jz short loc_405F95"]
    Node1 -- true --> Loop["sh esi, 2  
sub_407650  
v d, esi, ds:Sleep  
d esp, 4  
sh 300000 ; duMILLISECONDS  
esi ; Sleep  
sh 2  
sub_407650  
d esp, 4  
sh 600000 ; duMILLISECONDS  
esi ; Sleep  
sh 2  
sub_407650  
d esp, 4  
sh sub_40AC00  
r eax, esi  
p  
tn"]
    Node1 -- false --> Node2["push offset unk_40F1A0  
call sub_408490  
add esp, 4  
ret"]
    Node1 -- true --> Node3["call sub_405E9A  
call sub_4057E0  
test eax, eax  
jz short loc_405F95"]
    Node3 -- false --> Node4["xor eax, eax  
ret"]
    Node3 -- true --> Node5["push offset unk_40F1A0  
call sub_408490  
add esp, 4  
ret"]
    Node4 --> Exit(( ))
    Node2 --> Exit
    Node5 --> Exit
  
```

```

39 v12 = -1665/92991;
40 LoadLibraryA("wlanapi.dll");
41 LoadLibraryA("ntdll.dll");
42 LoadLibraryA("mpr.dll");
43 LoadLibraryA("iphlpapi.dll");
44 sub_405640(&v13, 9, &v9, 4);
45 WSASStartup(2u, &WSAData);
46 CoInitialize(0);
47 v13.v13 = 0;
48 v13.v13 = 0;
49 sub_409490(v13, 0);
50 sub_4092E0(v13, 0, 0, 0);
51 sub_407320(v13);
52 v13.v13 = 0;
53 v13.v13 = 0;
54 v13.v13 = 0;
55 v13.v13 = 0;
56 if (v13.v13 != (HANDLE)-1)
57 {
58     WSAData.lpVendorInfo = 0;
59     SetFilePointer(v5, -10240, 0, 2u);
60     sub_407320(v13);
61     sub_409430(v13);
62     sub_409430(v13);
63     sub_409430(v13);
64     sub_409430(v13);
65     sub_409430(v13);
66     sub_409430(v13);
67     sub_409430(v13);
68     sub_409430(v13);
69     sub_409430(v13);
70     sub_409430(v13);
71     sub_409430(v13);
72     sub_409430(v13);
73     sub_409430(v13);
74     sub_409430(v13);
75     sub_409430(v13);
76     sub_409430(v13);
77     sub_409430(v13);
78     sub_409430(v13);
79     sub_409430(v13);
80     sub_409430(v13);
81     sub_409430(v13);
82     sub_409430(v13);
83     sub_409430(v13);
84     sub_409430(v13);
85     sub_409430(v13);
86     sub_409430(v13);
87     sub_409430(v13);
88     sub_409430(v13);
89     sub_409430(v13);
90     sub_409430(v13);
91     sub_409430(v13);
92     sub_409430(v13);
93     sub_409430(v13);
94     sub_409430(v13);
95     sub_409430(v13);
96     sub_409430(v13);
97     sub_409430(v13);
98     sub_409430(v13);
99     sub_409430(v13);
100    sub_409430(v13);
101    sub_409430(v13);
102    sub_409430(v13);
103    sub_409430(v13);
104    sub_409430(v13);
105    sub_409430(v13);
106    sub_409430(v13);
107    sub_409430(v13);
108    sub_409430(v13);
109    sub_409430(v13);
110    sub_409430(v13);
111    sub_409430(v13);
112    sub_409430(v13);
113    sub_409430(v13);
114    sub_409430(v13);
115    sub_409430(v13);
116    sub_409430(v13);
117    sub_409430(v13);
118    sub_409430(v13);
119    sub_409430(v13);
120    sub_409430(v13);
121    sub_409430(v13);
122    sub_409430(v13);
123    sub_409430(v13);
124    sub_409430(v13);
125    sub_409430(v13);
126    sub_409430(v13);
127    sub_409430(v13);
128    sub_409430(v13);
129    sub_409430(v13);
130    sub_409430(v13);
131    sub_409430(v13);
132    sub_409430(v13);
133    sub_409430(v13);
134    sub_409430(v13);
135    sub_409430(v13);
136    sub_409430(v13);
137    sub_409430(v13);
138    sub_409430(v13);
139    sub_409430(v13);
140    sub_409430(v13);
141    sub_409430(v13);
142    sub_409430(v13);
143    sub_409430(v13);
144    sub_409430(v13);
145    sub_409430(v13);
146    sub_409430(v13);
147    sub_409430(v13);
148    sub_409430(v13);
149    sub_409430(v13);
150    sub_409430(v13);
151    sub_409430(v13);
152    sub_409430(v13);
153    sub_409430(v13);
154    sub_409430(v13);
155    sub_409430(v13);
156    sub_409430(v13);
157    sub_409430(v13);
158    sub_409430(v13);
159    sub_409430(v13);
160    sub_409430(v13);
161    sub_409430(v13);
162    sub_409430(v13);
163    sub_409430(v13);
164    sub_409430(v13);
165    sub_409430(v13);
166    sub_409430(v13);
167    sub_409430(v13);
168    sub_409430(v13);
169    sub_409430(v13);
170    sub_409430(v13);
171    sub_409430(v13);
172    sub_409430(v13);
173    sub_409430(v13);
174    sub_409430(v13);
175    sub_409430(v13);
176    sub_409430(v13);
177    sub_409430(v13);
178    sub_409430(v13);
179    sub_409430(v13);
180    sub_409430(v13);
181    sub_409430(v13);
182    sub_409430(v13);
183    sub_409430(v13);
184    sub_409430(v13);
185    sub_409430(v13);
186    sub_409430(v13);
187    sub_409430(v13);
188    sub_409430(v13);
189    sub_409430(v13);
190    sub_409430(v13);
191    sub_409430(v13);
192    sub_409430(v13);
193    sub_409430(v13);
194    sub_409430(v13);
195    sub_409430(v13);
196    sub_409430(v13);
197    sub_409430(v13);
198    sub_409430(v13);
199    sub_409430(v13);
200    sub_409430(v13);
201    sub_409430(v13);
202    sub_409430(v13);
203    sub_409430(v13);
204    sub_409430(v13);
205    sub_409430(v13);
206    sub_409430(v13);
207    sub_409430(v13);
208    sub_409430(v13);
209    sub_409430(v13);
210    sub_409430(v13);
211    sub_409430(v13);
212    sub_409430(v13);
213    sub_409430(v13);
214    sub_409430(v13);
215    sub_409430(v13);
216    sub_409430(v13);
217    sub_409430(v13);
218    sub_409430(v13);
219    sub_409430(v13);
220    sub_409430(v13);
221    sub_409430(v13);
222    sub_409430(v13);
223    sub_409430(v13);
224    sub_409430(v13);
225    sub_409430(v13);
226    sub_409430(v13);
227    sub_409430(v13);
228    sub_409430(v13);
229    sub_409430(v13);
230    sub_409430(v13);
231    sub_409430(v13);
232    sub_409430(v13);
233    sub_409430(v13);
234    sub_409430(v13);
235    sub_409430(v13);
236    sub_409430(v13);
237    sub_409430(v13);
238    sub_409430(v13);
239    sub_409430(v13);
240    sub_409430(v13);
241    sub_409430(v13);
242    sub_409430(v13);
243    sub_409430(v13);
244    sub_409430(v13);
245    sub_409430(v13);
246    sub_409430(v13);
247    sub_409430(v13);
248    sub_409430(v13);
249    sub_409430(v13);
250    sub_409430(v13);
251    sub_409430(v13);
252    sub_409430(v13);
253    sub_409430(v13);
254    sub_409430(v13);
255    sub_409430(v13);
256    sub_409430(v13);
257    sub_409430(v13);
258    sub_409430(v13);
259    sub_409430(v13);
260    sub_409430(v13);
261    sub_409430(v13);
262    sub_409430(v13);
263    sub_409430(v13);
264    sub_409430(v13);
265    sub_409430(v13);
266    sub_409430(v13);
267    sub_409430(v13);
268    sub_409430(v13);
269    sub_409430(v13);
270    sub_409430(v13);
271    sub_409430(v13);
272    sub_409430(v13);
273    sub_409430(v13);
274    sub_409430(v13);
275    sub_409430(v13);
276    sub_409430(v13);
277    sub_409430(v13);
278    sub_409430(v13);
279    sub_409430(v13);
280    sub_409430(v13);
281    sub_409430(v13);
282    sub_409430(v13);
283    sub_409430(v13);
284    sub_409430
```


The diagram illustrates a function call stack for the function `GetKeuhnardList`. The stack contains a single frame for this function, which is currently executing. The function's parameters are `id` (value 10) and `HRK` (value 0). The return value is `0`. The function's body is shown with a `result` variable initialized to `0`. The stack is managed by the `OS` (Operating System). The stack is shown as a vertical container with a `result` variable at the top, followed by the function's parameters and return value. The stack is managed by the `OS` (Operating System). The stack is shown as a vertical container with a `result` variable at the top, followed by the function's parameters and return value. The stack is managed by the `OS` (Operating System).

```

12 v1 = u0;
13 u2 = 0;
14 sub_409DD0((int)&v8);
15 v3 = sub_40B1C0((int)&v8);
16 if ( v3 >= 0 )
17 {
18     sub_409DD0((int)&duNumberOfBytesAvailable);
19     while ( os7b_kentg9z69HBBX%wN == -0x6bba-0x23-0x3 )
20         ;

```

```

jtesAvailable :
jFBytesAvailable, v7::
jtesAvailable)

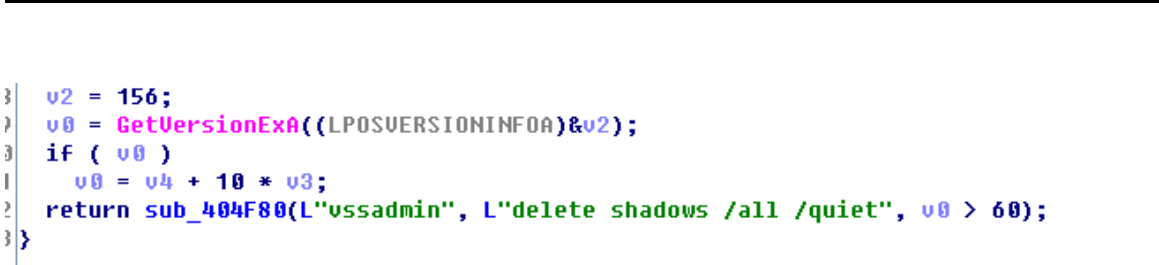
```

```

16 v3 = sub_4072A0(v2);
17 v4 = (const CHAR *)sub_40AAB0("%s\\__config%u.bat", v3);
18 v5 = CreateFileA(v4, 0x40000000u, 7u, 0, 2u, 0x102u, 0);
19 if ( v5 == (HANDLE)-1 )
20 {
21     result = 0;
22 }
23 else
24 {
25     v7 = (const CHAR *)sub_40AAB0(
26         ":abx\r\n"
27         "ping 127.0.0.1 -n 2 > nul\r\n"
28         "del /A /F /Q \"%s\\\" \r\n"
29         "if exist \"%s\" goto abx\r\n"
30         "del /A /F /Q \"%s\\\" \r\n",
31         v1,
32         v1,
33         v4);
34
35     v8 = v7;
36     v9 = strlenA(v7);
37     WriteFile(v5, v8, v9, &NumberOfBytesWritten, 0);

```

.dat .mx0 .cd .pdb .xqx .old .cnt .rtp .qss .qst .fx0 .fx1 .ipg .ert .pic .img.cur .fxr .slk .m4u .mpe .mov .wmv .mpg
.vob .mpeg .3g2 .m4v .avi .mp4 .flv.mkv .3gp .asf .m3u .m3u8 .wav .mp3 .m4a .m .rm .flac .mp2 .mpa .aac .w
ma .djv.pdf .djvu .jpeg .jpg?www.2cto.com .bmp .png?www.2cto.com .jp2 .lz .rz .zipx .gz .bz2 .s7z .tar .7z .tgz.r
ar .zip .arc .paq .bak .set .back .std .vmx .vmdk .vdi .qcow .ini .accd .db.sqli .sdf .mdf .myd .frm .odb .myi .dbf .i
ndb .mdb .ibd .sql .cgn .dcr .fpx.pcx .rif .tga .wpg .wi .wmf .tif .xcf .tiff .xpm .nef .orf .ra .bay .pcd .dng.ptx .r3d .
raf .rw2 .rwl .kdc .yuv .sr2 .srf .dip .x3f .mef .raw .log .odg .uop.potx .potm .pptx .rss .pptm .aaf .xla .sxd .pot .e
ps .as3 .pns .wpd .wps .msg.pps .xlam .xll .ost .sti .sxi .otp .odp .wks .vcf .xltx .xltn .xlsx .xlsm.xlsb .cntk .xlw .xl



```

v3 = sub_4087D0(8u, 102);
// 4087D0: sub_4087D0(8u, 102) = 102
// 4087D0: sub_4087D0(8u, 102) = 102

```

[illegible]

. **APT**

4.1 **APT**

APT

APT