**Developing anonymous FTP Scanner using flib module**

EXP NO: 8

DATE: 19/9/25

**AIM:**

To build anonymous FTP Scanner using flib module

**PROCEDURE:**

Install Python (3.8+) and ensure internet access for targets you are authorized to scan.

 Create a text file targets.txt with one hostname or IP per line (only targets you own or have permission to test).

 Save the scanner code below as anon\_ftp\_scanner.py.

 Run the script: python anon\_ftp\_scanner.py — it will read targets.txt, try anonymous login, and print results.

 Review the printed output and any saved logs. **Important:** Only scan hosts you have permission to test.

**CODE:**

# Simple Anonymous FTP Scanner using ftplib (built-in)

# NOTE: Only scan hosts you are authorized to test.

import ftplib

import socket

import sys

from time import sleep

TARGETS\_FILE = "targets.txt" # one host per line

TIMEOUT = 5 # seconds

RETRY\_DELAY = 0.5

def check\_anonymous\_ftp(host):

"""Try anonymous FTP login on host. Return dict with results."""

result = {"host": host, "reachable": False, "anonymous": False, "message": "", "listing": None}

try:

# resolve host

ip = socket.gethostbyname(host)

except Exception as e:

result["message"] = f"Resolve error: {e}"

return result

try:

ftp = ftplib.FTP()

ftp.connect(host=host, timeout=TIMEOUT)

result["reachable"] = True

# Attempt anonymous login

resp = ftp.login(user='anonymous', passwd='anonymous@')

result["message"] = resp

# If login succeeded, try to list root directory

try:

listing = []

ftp.retrlines('LIST', callback=listing.append)

result["anonymous"] = True

result["listing"] = listing

except Exception as e:

result["message"] += f" | Listing failed: {e}"

finally:

try:

ftp.quit()

except Exception:

ftp.close()

except ftplib.error\_perm as e:

result["message"] = f"Permission/FTP error: {e}"

except (socket.timeout, TimeoutError):

result["message"] = "Connection timed out"

except ConnectionRefusedError:

result["message"] = "Connection refused"

except Exception as e:

result["message"] = f"Error: {e}"

return result

def main():

# Read targets

try:

with open(TARGETS\_FILE, 'r') as f:

targets = [line.strip() for line in f if line.strip()]

except FileNotFoundError:

print(f"Targets file '{TARGETS\_FILE}' not found. Create it with one hostname/IP per line.")

sys.exit(1)

print("Starting Anonymous FTP Scanner...")

print(f"Scanning hosts from list: {TARGETS\_FILE}\n")

scanned = 0

anon\_count = 0

for i, host in enumerate(targets, start=1):

print(f"{i}) {host}:")

res = check\_anonymous\_ftp(host)

if not res["reachable"]:

print(f" -> Reachable: NO ({res['message']})")

else:

if res["anonymous"]:

anon\_count += 1

print(f" -> Anonymous login: SUCCESS")

print(f" -> Welcome: {res['message']}")

if res["listing"] is not None:

sample = res['listing'][:5] # show first few entries

print(f" -> Root listing: {sample}")

else:

print(f" -> Anonymous login: FAILED ({res['message']})")

scanned += 1

sleep(RETRY\_DELAY)

print(f"\nScan complete: {scanned} hosts checked, {anon\_count} with anonymous access.")

if \_\_name\_\_ == "\_\_main\_\_":

main()

**OUTPUT:**

****

**RESULT:**

Thus anonymous FTP Scanner using flib module built successfully.