REG NO : 3122235002128

NAME : SOWKANDHA RAGHAV G DEPT : IT – C

NETWORK PROGRAMMING LAB EX NO : 3

PING AND TRACEROUTE

Write a code simulating PING and TRACEROUTE commands using Twisted Python. CODING:

from twisted.internet import reactor,defer

from twisted.internet.protocol import DatagramProtocol import random

import time import sys

class PingClient(DatagramProtocol): def init (self,host,count=4):

self.host=host self.count=count self.seq=0 self.delays=[]

def startProtocol(self):

print(f"Pinging {self.host} with {self.count} packets...") self.transport.connect(self.host,33434) self.sendPing()

def sendPing(self):

if self.seq<self.count: self.seq+=1

self.transport.write(b"PING") print(f"Sent PING {self.seq}") self.sent\_time=time.time()

reactor.callLater(1,self.sendPing) else:

reactor.callLater(2,self.stop)

def datagramReceived(self,data,addr): rtt=(time.time()-self.sent\_time)\*1000 self.delays.append(rtt)

print(f"Reply from {addr[0]}: seq={self.seq} time={rtt:.2f} ms") def stop(self):

if self.delays:

print(f"Ping statistics for {self.host}: min={min(self.delays):.2f} ms, " f"max={max(self.delays):.2f} ms, avg={sum(self.delays)/len(self.delays):.2f} ms")

reactor.stop()

class TracerouteClient(DatagramProtocol): def init (self,host,max\_hops=30):

self.host=host self.ttl=1

self.max\_hops=max\_hops def startProtocol(self):

self.transport.connect(self.host,33434) self.sendPacket()

def sendPacket(self):

if self.ttl>self.max\_hops:

print("Traceroute completed.") reactor.stop()

return

self.transport.write(b"TRACE") start\_time=time.time()

reactor.callLater(1,self.checkResponse,start\_time) def checkResponse(self,start\_time):

rtt=(time.time()-start\_time)\*1000 if random.random() > 0.3:

print(f"{self.ttl}\t\* \* \* Request timed out.") else:

print(f"{self.ttl}\t{self.host} {rtt:.2f} ms") self.ttl+=1

self.sendPacket()

if name ==' main ': if len(sys.argv) < 3:

print("Usage: python script.py <ping|traceroute> <host>") sys.exit(1)

command,host=sys.argv[1],sys.argv[2] if command.lower()=="ping":

reactor.listenUDP(0,PingClient(host)) elif command.lower()=="traceroute":

reactor.listenUDP(0,TracerouteClient(host)) else:

print("Invalid command. Use 'ping' or 'traceroute'.") sys.exit(1)

reactor.run()

OUTPUT:

