Ampliación de Redes

Final de Septiembre (Práctica), 05 de septiembre de 2005



Este examen consta de 5 ejercicios con un total de 100 puntos. Utilice letra clara y escriba únicamente en el espacio reservado. Cada 10 errores ortográficos restan 5 puntos a la nota total.

Apellidos:	Nombre:	Grupo:
1. (10p) ¿Con qué programa se puede av	eriguar el RTT(Round-Trip Time) a un host?	
\square a). traceroute	\square c). nmap	
\square b). ping	d). netstat	
2. (10p) ¿En qué se basa el funcionamier	nto del traceroute estándar?	
☐ a). Consultar las tablas de rutas de	los enrutadores intermedios.	
□ b). Enviar paquetes IP con valores	de TTL incrementales.	
c). Enviar paquetes ICMP broadca	st.	
☐ d). Cualquiera de las anteriores.		
· · ·		
3. (10p) ¿Cómo podríamos hacer un ping	<mark>g estándar a una máquina local</mark> de l <mark>a que</mark> sólo conoc	emos su dirección física?
☐ a). No se puede.		
□ b). Modificando la tabla ARP de n	uestro host e inventándonos una IP.	
\Box c). Usando <i>ping</i> directament con la		
☐ d). Ninguna de las anteriores.		



4. (35p) Modifica el siguiente servidor TCP para convertirlo en UDP. Indica cual es la sintaxis de llamada al programa.

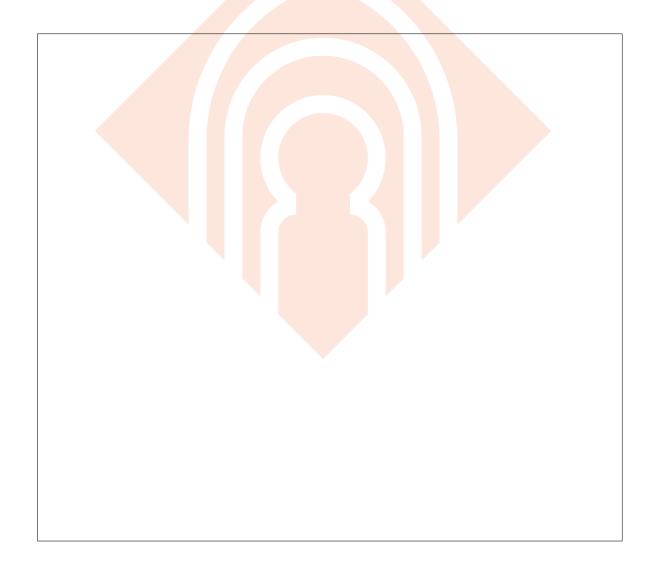
```
from sys import argv, exit
from socket import *
import time

def handle(sock, client):
    while 1:
        data = sock.recv(32)
        if not data: break
        time.sleep(1) # some job
        sock.sendall(data.upper())
        sock.close()

if len(argv) != 2:
    print __doc__ % argv[0]
    exit(1)

sock = socket(AF_INET, SOCK_STREAM)
sock.bind(('', int(argv[1])))
sock.listen(5)

while 1:
    child_sock, client = sock.accept()
handle(child_sock, client)
```





5. (35p) Modifica el siguiente cliente UDP para convertirlo en TCP. Indica cual es la sintaxis de llamada al programa.

```
from sys import argv, exit
from socket import *

if len(argv) != 3:
    print __doc__ % argv[0]
    exit(1)

sock = socket(AF_INET, SOCK_DGRAM)

while 1:
    data = raw_input()
    if not data: break

    sock.sendto(data, (argv[1], int(argv[2])))
    msg, server = sock.recvfrom(1024)
    print "La respuesta es '%s'" % msg

sock.close()
```

