

# Actividad Fragmentación

## 1. router.py

Este módulo contiene la clase `Router`, la cual representa un router con las funcionalidades para manejar y parsear los paquetes IP, asimismo maneja el forwarding de estos paquetes chequeando las rutas de la tabla de rutas otorgada. Finalmente también maneja el fragmentado y posterior re-ensamblaje de paquetes en caso de que la ruta posea un MTU menor al tamaño del paquete. Esta clase se encuentra totalmente documentada por lo que no requiere entrar en detalle respecto a los métodos implementados.

## 2. router\_test.py

Este código crea un router con dirección IP según los parámetros entregados en la ejecución de este, como se muestra abajo, adicionalmente se encarga de manejar un diccionario con los paquetes/fragmentos que tengan destino la dirección asociada, en tal caso se realiza el manejo adecuado de estos mediante los correspondientes métodos de la clase `Router`.

```
$ python3 router_test.py router_IP router_puerto router_rutas.txt
```

## 3. Pruebas

- **Pruebe sus routers usando un paquete IP de tamaño total 150 bytes. Envíe dicho paquete varias veces desde R1 a R5 usando netcat como en la prueba anterior. Verifique que la fragmentación y el re-ensamblaje son exitosos ¿Cómo cambia el comportamiento si los envía desde otros routers? ¿Su código logra fragmentar un fragmento en fragmentos más pequeños en caso de ser necesario? Describa brevemente sus observaciones en el informe.**

El envío del paquete se realiza satisfactoriamente, llegando por completo al router 5. Al cambiar la dirección desde donde se envía el paquete este vuelve a llegar por completo al router 5 (se probó a enviar el paquete desde los routers 1, 2, 3 y 4) teniendo en cuenta que las rutas que toman los fragmentos varían en cada envío, pues se alternan las rutas que se utilizan debido al uso de round-robin. Respecto al proceso de fragmentación de fragmentos, esto es posible y se observa en el código al momento de la ejecución, asimismo en la entrega se agrega un código para probar este proceso individualmente el cual se encuentra en el archivo `test.py`.