

LABORATORIO 5 – TRANSMISIÓN DE DATOS SOBRE UDP

Laboratorio de infraestructura de comunicaciones

4.3

Durante la ejecución de las pruebas propuestas utilice el comando netstat para identificar las conexiones UDP activas. Presente en una tabla la información obtenida. ¿todas las aplicaciones cliente utilizan el mismo puerto local? Explique.

R: No. De hecho, en cada uno de los clientes en los que se ejecutó la aplicación el puerto local fue diferente. De la misma manera, el puerto TCP utilizado para enviar la cantidad de objetos al servidor es distinto al puerto UDP para enviar los respectivos objetos. Estos puertos no sólo varían para los diferentes cliente, sino que cada vez que el programa cliente se ejecuta estos puertos varían (aún cuando el cliente ya terminó la ejecución anterior). Cabe recordar que el mecanismo para la asignación de puertos es dependiente del sistema operativo.

El código fuente y las capturas se encuentran en UDPLab\4.1-4.2 y UDPLab\4.3\capturas, respectivamente.

Para los puntos 8 y 9

Hubo problemas con los servicios en la nube. Tanto AWS como Azure no permiten ingresar tráfico por el puerto 4001 y 12345 a pesar de que se establecieron reglas de excepción específicas para que se dejara pasar el tráfico de esos puertos (se permitió tráfico desde cualquier IP). Así pues, no se pudo realizar estos puntos.

Aún así, respecto al punto 9, no todos los clientes tienen el mismo puerto. Al momento de aceptar la conexión, el servidor le da un puerto disponible al cliente. De este forma, múltiples clientes pueden estar conectados al servidor en simultáneo en puertos distintos.

4.4

En todas las pruebas los archivos fueron recibidos correctamente. Las capturas por los diferentes casos de prueba se encuentran en UDPLab\4.4-UDPfile\capturas. El tamaño de los paquetes oscila entre 60 y 400 bytes.

4.5. Intercambio de archivos en Bittorrent

Verifique la captura realizada en Wireshark para cada uno de los escenarios y cada

una de las pruebas, identifique el tamaño y la cantidad de paquetes intercambiados entre cliente y servidor.

Archivo 2.12MB: 136 partes de 15.59KB cada una. Tiempo de descarga: 8s

Archivo 8.88MB: 569 partes de 15.60KB cada una. Tiempo de descarga: 17s

Archivo 47.2MB: 756 partes de 62.43KB cada una. Tiempo de descarga: 47s

Para todas las transferencias ocurre lo mismo: unos pocos paquetes bajo el protocolo TCP al iniciar la descarga. Después, paquetes de 1425 bytes del servidor al cliente y paquetes de 20 bytes del cliente al servidor, ambas bajo UDP. Esto se evidencia en las capturas que se encuentran en la carpeta UDPLab\4.5-bittorrent\capturas.

4.5. Intercambio de archivos en TCP

Verifique la captura realizada en Wireshark para cada uno de los escenarios y cada una de las pruebas, identifique el tamaño y la cantidad de paquetes intercambiados entre cliente y servidor.

Tiempos de transferencia:

Archivo pequeño (2.12 MB): 0.21s

Archivo mediano (8.88 MB): 0.85s

Archivo grande (47.2 MB): 4.17s

En los tres casos los archivos se recibieron correctamente. También se validó que el hash enviado por el servidor corresponde con el hash calculado en el cliente. El código fuente y las capturas se encuentran en UDPLab\4.5-TCP.

Repositorio con el código ejecutado:

<https://github.com/mneira10/UDPLab.git>