Grupo 12 – Prioridades y colas

[1 NS3 ( Network Simulator) 2](#_Toc472993771)

[2 Objetivo 2](#_Toc472993772)

[3 Escenario 3](#_Toc472993773)

[4 Configuraciones 4](#_Toc472993774)

[5 Simulaciones 4](#_Toc472993775)

[6 Validez de los Resultados 4](#_Toc472993776)

[7 Conclusiones 5](#_Toc472993777)

BEATRIZ CARRETERO PARRA

ALVARO ESPEJO MUÑOZ

ALBERTO FRLORES PEREIRA

MIGUEL ALEJANDRO GARCIA BENITEZ

# NS3 ( Network Simulator)

NS3 es un simulador de redes escrito en C++, con un codigo fuente de más de 300.000 lineas con una licencia GNU General Public License v2, lo que garantiza un codigo libre. Pueda ser modificado y compartido de nuevo, pero solo de forma libre y bajo ningun concepto convertirlo en un software privativo.

Esta constituido por modulos que definen dispositivos, protocolos y utilidades reales. Es capaz de generar simulaciones de gran escala basadas en eventos discretos programados en una cola de tiempo de ejecución.

# Objetivo

Medir el funcionamiento de un flujo TCP por un cuello de botella, construyendo un sistema de prioridades, dando a cada paquete una prioridad determinada por el tipo de tráfico. Con un cuello de botella en un escenario creado estudiar las mejores opciones para un tráfico fluido en la red variando las prioridades de los distintos tráficos.

Estudiar el funcionamiento de las colas y las prioridades, no impartidas en clase y para las cuales hemos actualizado NS3 a su versión 3.26.

# Escenario

Cuello de Botella

Cliente VoIP

Cliente VoIP

Cliente Streaming

Servidor Streaming

HTTP

Servidor Echo

FTP

Router 2

Router 1

*Imagen 1*

Hemos creado un escenario con dos redes LAN, una simulando un cliente y otra siendo el otro extremo de la comunicación o servidor, unidas únicamente por un enlace p2p (imagen 1).

* Clientes:
  + VoIP
  + Streaming
  + HTTP
  + FTP
* Servidores:
  + VoIP
  + Streaming
  + Echo

El cliente VoIP se comunicará con su servidor, al igual que el servicio de Streaming. Los servicios HTTP y FTP se comunicarán con un servidor de eco, para un ejemplo simple de tráfico.

El escenario está planteado para que se genere un cuello de botella en el enlace entre el router 1 y el router 2 al usar todos los servicios a la vez, y estudiar el funcionamiento de la red ejecutando una cola con prioridades y modificando las prioridades de cada tipo de tráfico para su estudio.

# Configuraciones

Dependiendo de la programación final

|  |  |  |
| --- | --- | --- |
| Trafico | Nodo Destino | Prioridad |
| VoIP | Router 2 | 1 |
| Streaming | Router 2 | 2 |
| HTTP | Router 2 | 3 |
| FTP | Router 2 | 4 |
| VoIP | Router 2 | 2 |
| Streaming | Router 2 | 1 |
| HTTP | Router 2 | 3 |
| FTP | Router 2 | 4 |

# clases de la simulación

En las simulaciones hemos usado la clase PfifoFastQueueDisc[[1]](#footnote-1), que usa tres colas FIFO, acordes a las tres bandas que posee, basadas en prioridades dadas a los paquetes. Los paquetes de una banda con mayor prioridad siempre saldrán de la cola antes que los paquetes de otra banda menos prioritaria.

Hemos recreado el protocolo VoIP y simulado una comunicación real de este protocolo entre dos usuarios, intentando conseguir más del 99% de los paquetes correctos en la comunicación para lograr que ésta sea correcta.

También hemos recreado un servicio de streaming y un cliente que lo usa en la red, por lo que podemos estudiar cuando el usuario consigue más del 95% de paquetes correctos para una comunicación correcta

Y por último hemos intentado simular trafico HTTP y FTP enviando paquetes a un servidor de echo que los devuelve.

Con estos tres servicios queremos comenzar a simular modificando parámetros de la cola y asignando prioridades en paquetes.

Clases creadas para la simulación comentadas a continuación:

## VoIP.cc

## HTTP.cc

## FTP.cc

## Escenario.cc

## Simulacion.cc

## Observador.cc

# Simulaciones

Explicar proceso de simulación.cc

# Validez de los Resultados

Considerando los resultados obtenidos de las simulaciones, tenemos que validarlos según el criterio de simulación y verificar que tienen sentido en la simulación realizada y que hemos llegado a los valores esperados antes de realizarlas.

# Conclusiones

Comprobamos que aplicando una cola con prioridades en un escenario simple con cuello de botella logramos tener una red funcional para los servicios con mayor requisito de ancho de banda como puede ser VoIP y Streaming, dejando los demás servicios en segundo plano, pero no totalmente bloqueados por los prioritarios ya que si no se convertiría en un sistema en el que los servicios prioritarios toman el control del canal.

1. <https://www.nsnam.org/docs/release/3.26/doxygen/classns3_1_1_pfifo_fast_queue_disc.html> [↑](#footnote-ref-1)