



**Universidade do Minho**  
Escola de Engenharia

# Engenharia de Serviços em Rede

Streaming de áudio e vídeo a pedido e em tempo real

Relatório Trabalho Prático 2

Grupo **PL43**

---

MEI - 1º Ano - 1º Semestre

**Trabalho realizado por:**

PG53601 - Afonso Xavier Cardoso Marques

PG55093 - Pedro Alexandre da Silva Oliveira

PG55097 - Vasco Rafael Barroso Gonçalves Rito

Braga, 26 de novembro de 2024

# Índice

<b>1. Introdução .....</b>	<b>4</b>
<b>2. Arquitetura da Solução .....</b>	<b>4</b>
2.1. Script .....	5
2.2. Servidores de Streaming .....	5
2.3. Points of Present (POP) .....	6
2.4. oNode .....	6
2.5. Cliente .....	6
<b>3. Especificação dos protocolos .....</b>	<b>6</b>
<b>4. Etapa 1 - Construção da Topologia Overlay e Underlay .....</b>	<b>6</b>
4.1. ....	6
<b>5. Etapa 2 - Streaming adaptativo sobre HTTP (MPEG-DASH) .....</b>	<b>8</b>
<b>6. Etapa 3 - Streaming RTP/RTCP unicast sobre UDP e multicast com anúncios SAP .....</b>	<b>9</b>
6.1. Perguntas e Respostas .....	9
<b>7. Conclusão .....</b>	<b>9</b>

## **Lista de Figuras**

<b>Figura 1: Arquitetura do Sistema .....</b>	<b>4</b>
<b>Figura 2: Script para correr Topologia .....</b>	<b>5</b>

## 1. Introdução

Neste trabalho prático foi-nos proposto o desenvolvimento de um serviço Over the Top para a entrega de conteúdos de multimédia. Tendo como objetivo desenvolver e prototipar um serviço de entrega de vídeo em tempo real, recorrendo à utilização de Points of Presence (POP) e uma rede de overlay. NO FINAL DO TRABALHO COMPLETAR !!!!!

## 2. Arquitetura da Solução

Neste trabalho é pretendido que se crie um sistema de entrega de multimédia em tempo real, partindo de um servidor de streaming para um conjunto de clientes. Para isso, os Points of Presence (POP), irão estar na fronteira da rede CDN e a rede de acesso, fazendo de ponto de contacto dos clientes com a CDN. Os POPs recebem o conteúdo via multicast e depois enviam-no via unicast para cada um dos clientes que pretendem aceder aquele conteúdo. A seguinte imagem mostra a arquitetura geral de sistema.

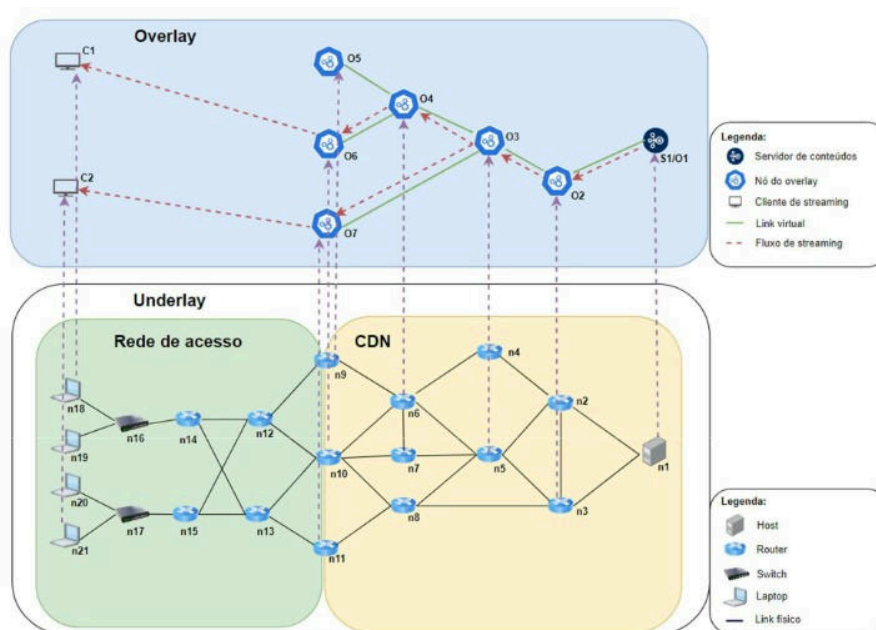


Figura 1: Arquitetura do Sistema

Para a implementação do trabalho prático utilizámos a linguagem de programação Python pois, além de possuir bibliotecas que facilitam o uso de sockets e threads, também é a linguagem com que o grupo se sente mais à vontade. Para a construção da rede de overlay decidimos utilizar um ficheiro de configuração JSON, onde está presente a informação que cada componente irá utilizar. Agora iremos proceder a explicar os argumentos necessários e a estrutura dos ficheiros de configuração.

## 2.1. Script

De forma a tornar mais rápida e simples a execução da topologia decidimos desenvolver um script.



```
/home/core/Desktop/ESR-2425/TP2/code/script.sh - Mous
File Edit Search View Document Help

#!/bin/bash

# Comandos iniciais para configurar o ambiente

cd /home/core/Desktop/ESR-2425/TP2/code

boot_ip="10.0.25.10"
topologia="topologias/topo_overlay.json"

# Verifique o nome do nó para decidir o comando específico
case $1 in
    "c")
        export DISPLAY=:0.0
        python3 oClient.py "$boot_ip" "$2"
        ;;
    "n")
        python3 oNode.py -n "$boot_ip"
        ;;
    "pop")
        python3 oNode.py -pop "$boot_ip"
        ;;
    "s")
        python3 oNode.py -s "$topologia" "$2"
        ;;
    *)
        echo "Node not recognized"
        ;;
esac
```

Figura 2: Script para correr Topologia

## 2.2. Servidores de Streaming

Os servidores de streaming são responsáveis por enviar o conteúdo inicial para a rede. Estes servidores utilizam como input um ficheiro JSON contendo a configuração da topologia, além de uma flag que define se o servidor será ou não o bootstrapper. O bootstrapper é o nó inicial que coordena a descoberta da rede.

O servidor pode ser inicializado utilizando o comando:

```
python3 oNode.py -s "topologia.json" 1
```

- A flag '-s' indica que o nó é do tipo servidor.
- O número '1' no final, especifica que o servidor será inicializado como bootstrapper.

Se a flag for o número '0', o servidor será inicializado como não bootstrapper.

## 2.3. Points of Present (POP)

Os Points of Presence (POP) são responsáveis por retransmitir o conteúdo recebido pelos servidores para os clientes. Eles funcionam como intermediários na rede, permitindo escalabilidade e eficiência na transmissão.

Os POPs utilizam como input o endereço IP do bootstrapper (boot IP), que é o primeiro nó da rede. A partir deste endereço, os POPs conseguem descobrir e conectar-se aos seus vizinhos na topologia.

Para inicializar um POP, utiliza-se o comando:

```
python3 oNode.py -pop "endereço_bootstrapper"
```

- A flag '-pop' indica que o nó é um POP.
- "endereço\_bootstrapper" deve ser substituído pelo IP do bootstrapper.

## 2.4. oNode

O oNode é um componente genérico que pode atuar como diferentes tipos de nós na rede overlay, dependendo das flags fornecidas na inicialização. Ele pode ser configurado para funcionar como:

- Servidor '-s': Um nó responsável por enviar o conteúdo inicial para a rede.
- Nó intermédio '-n': Um nó que atua como intermediário na transmissão de dados.
- POP '-pop': Um nó que retransmite o conteúdo dos servidores para os clientes.

O comportamento de cada oNode é definido pelos argumentos passados no momento da execução, permitindo flexibilidade na configuração da rede.

## 2.5. Cliente

Os clientes são os nós finais da rede que solicitam e consomem o conteúdo transmitido. Cada cliente conecta-se à rede utilizando o endereço do bootstrapper (boot IP) para localizar os servidores ou POPs responsáveis por fornecer o conteúdo solicitado.

Antes de executar o cliente, é necessário configurar a variável de ambiente DISPLAY, que define o ecrã onde será exibido o conteúdo.

Para inicializar um cliente, utilizam-se os seguintes comandos:

```
export DISPLAY=:0.0
```

```
python3 oClient.py "endereço_bootstrapper" "$2"
```

- "endereço\_bootstrapper" deve ser substituído pelo IP do bootstrapper.
- '\$2': Representa o nome do conteúdo que será transmitido.

## 3. Especificação dos protocolos

O protocolo aplicacional para o streaming foi o Real-time Transport Protocol (RTP), usámos como base a implementação que foi fornecida pela equipa docente. Oferecendo simplicidade e baixa latência, que são características fundamentais para transmissões em tempo real. A transmissão dos

## 4. Etapa 1 - Construção da Topologia Overlay e Underlay

### 4.1.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura.

## 5. Etapa 2 - Streaming adaptativo sobre HTTP (MPEG-DASH)

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguere possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura.



## **6. Etapa 3 - Streaming RTP/RTCP unicast sobre UDP e multicast com anúncios SAP**

### **6.1. Perguntas e Respostas**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aequale doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguere possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensiva et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura.

## **7. Conclusão**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aequale doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguere possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensiva et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura.