**Escola de Engenharia**

Mestrado em Engenharia Informática

Engenharia de Serviços em Rede

TP2 - Streaming de áudio e vídeo a pedido e em tempo real

# Grupo 37

# Pedro Veloso (pg47578)

# Carlos Preto (pg47089)

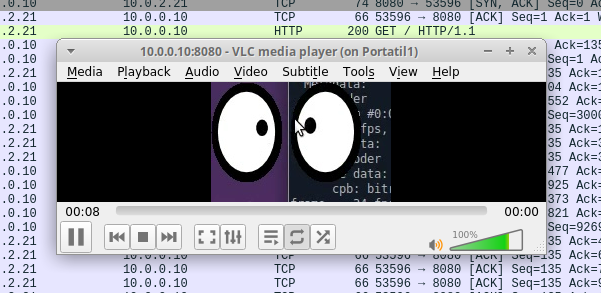
Braga, Novembro de 2021

# Questões e Respostas

**Questão 1: Capture três pequenas amostras de trágefo no link de saída do servidor, respetivamente com 1 cliente (VLC), com 2 clientes (VLC e Firefox) e com 3 clientes (VLC, Firefox e ffmeg). Identifique a taxa em bps necessária (usando o ffmpeg -i video1.mp4 e/ou o próprio *wireshark*), o encapsulamento usado e o número total de fluxos gerados. Comente a escalabilidade da solução. Ilustre com evidências da realização prática do exercício (ex: capturas de ecrã).**

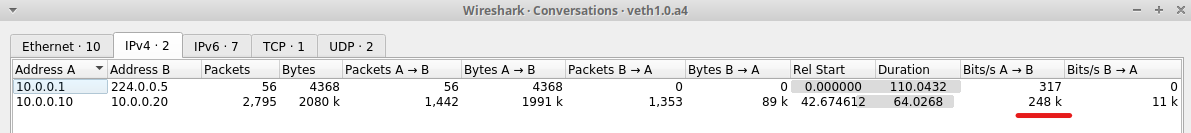
Para realizar o *streaming* de um vídeo, recorreu-se ao VLC, que permite de uma forma simples realizar estas operações. Contudo, apesar das transmissões geradas funcionarem bem para clientes que utilizem o VLC ou o ffmpeg, realizou-se o *transcoding* do vídeo com o *codec* “*Theora*” que permite utilizar o browser para a sua reprodução, empacotando-o posteriormente num contentor do tipo OGG.

Assim, com 1 cliente (VLC), verificou-se que foi necessária uma taxa de 248 Kbps de modo a transmitir o vídeo para o cliente. Para este cliente foi gerado apenas um fluxo (como se visualiza ao inserir o filtro “tcp.stream eq 0” no *wireshark*)

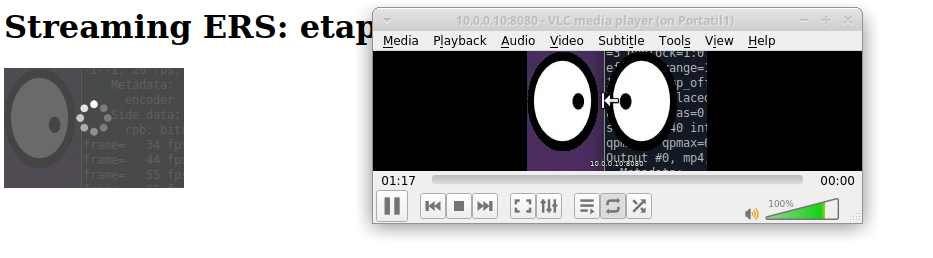


Uma imagem com mesa

Descrição gerada automaticamente

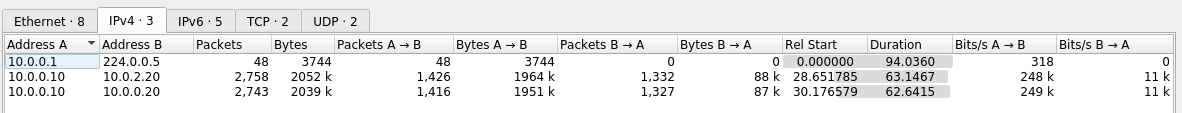


Para a utilização de 2 clientes, um a utilizar o VLC (portátil 1, com ip=10.0.0.20) e outro a utilizar o *Firefox* (portátil 2, com ip=10.0.2.20) verificou-se a necessidade de uma velocidade de 249 Kbps para o cliente 1 e de 248 Kbps para o cliente 2, totalizando desta forma na necessidade de 497 Kbps. Neste caso, foram gerados 2 fluxos.

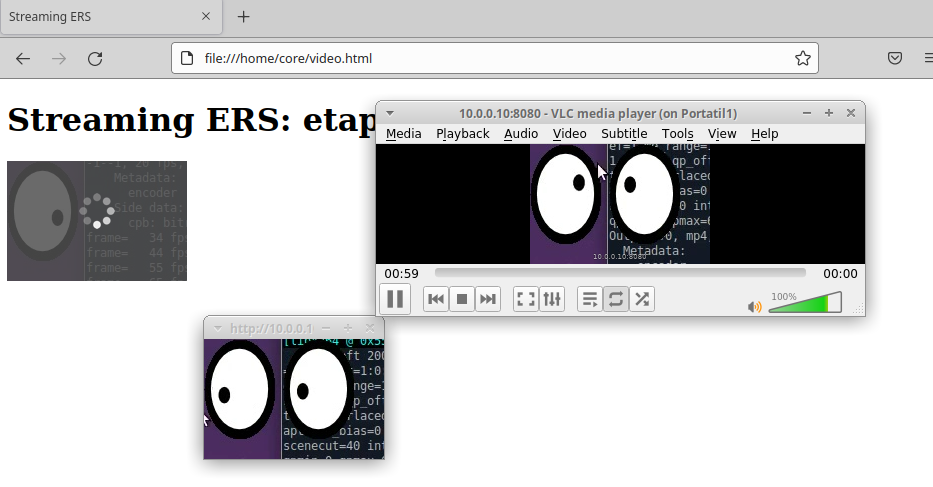


Uma imagem com mesa

Descrição gerada automaticamente

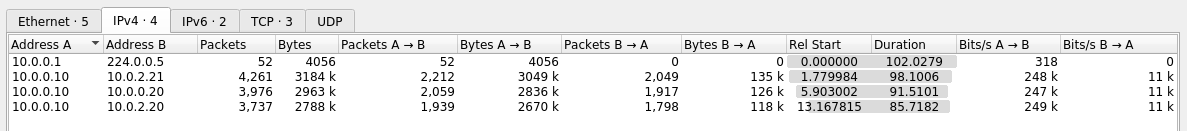


Com a adição de mais um cliente (portátil 3, ip=10.0.2.21), verificou-se a necessidade de 744 Kbps para realizar o *streaming* do vídeo para os 3 clientes. Assim, neste teste verificou-se a existência de 3 fluxos gerados.



Uma imagem com mesa

Descrição gerada automaticamente



**Questão 2: Diga qual a largura de banda necessária, em bits por segundo, para que o cliente de *streaming* consiga receber o vídeo no *firefox* e qual a pilha protocolar usada neste cenário.**

A largura de banda necessária para realizar o *streaming* de dados, nesta captura, foi de aproximadamente 3.6 Mbps, tal como se observa na seguinte imagem:

Uma imagem com mesa

Descrição gerada automaticamente

Por fim, como principais camadas da pilha protocolar utilizadas considerou-se a camada de transporte e a camada de aplicação.

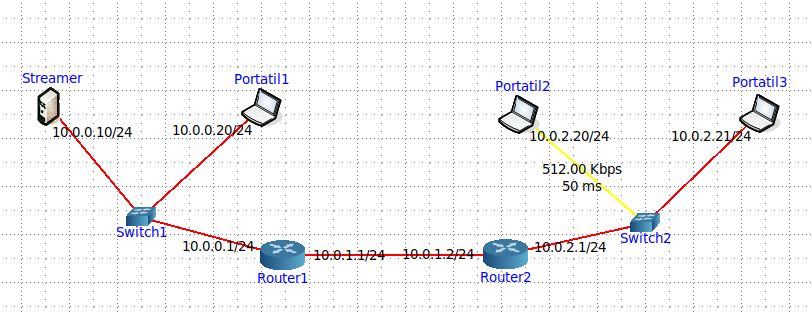
Para a camada de transporte identificou-se o uso de dois protocolos, o TCP para a transferência dos dados dos ficheiros e o UDP para contactar o servidor de DNS. Para a camada de aplicação utilizou-se o HTTP.

Uma imagem com mesa

Descrição gerada automaticamente

**Questão 3: Ajuste o débito dos links da topologia de modo que o cliente no portátil 2 exiba o vídeo de menor resolução e o cliente no portátil 1 exiba o vídeo com mais resolução. Mostre evidências.**

De modo a garantir que o portátil 1 tem uma melhor resolução do que o portátil 2, limitou-se a velocidade do link, que liga ao portátil 2, a 512 Kbps. Assim obteve-se a seguinte topologia:

****

Posteriormente, verificou-se que o ficheiro pedido pelo portátil 1 é o vídeo de melhor qualidade. Assim, ao analisar o tráfego com o *wireshark* verificou-se que era realizado um pedido HTTP GET do vídeo “video2\_540\_360\_1000k\_dash.mp4” (vídeo de maior qualidade).

Uma imagem com mesa

Descrição gerada automaticamente

Por outro lado, o portátil 2 realiza um pedido do vídeo de maior qualidade no início, baixando de imediato para o vídeo de menor qualidade (“video2\_180\_120\_200k\_dash.mp4”). Este processo é dado devido à capacidade adaptativa do protocolo DASH e é evidenciado na seguinte figura:

Uma imagem com mesa

Descrição gerada automaticamente

**Questão 4: Descreva o funcionamento do DASH neste caso concreto, referindo o papel do ficheiro MPD criado.**

O protocolo DASH foi desenvolvido com o objetivo de garantir que um cliente seja capaz de visualizar um vídeo com o mínimo de interrupções possíveis.

Neste exemplo, devido à menor capacidade do *link* que liga o computador 2, é possível perceber que, tal como evidenciado na questão anterior, a qualidade de vídeo recebida neste computador é menor. Este facto dá-se devido à capacidade adaptativa do DASH, que ao perceber que a capacidade do link não é suficiente para visualizar o vídeo com a melhor qualidade diminuiu-a, de forma automática, para uma qualidade inferior.

Para realizar este processo, o DASH utiliza um ficheiro MDP. Este ficheiro é representado no formato XML e possui as seguintes informações importantes para adaptar dinamicamente o conteúdo do vídeo:

* *Period*: apresenta a duração total do vídeo;
* *AdaptationSet*: apresenta o conjunto de todos os segmentos do vídeo/áudio a serem transmitidos;
* *Represention*: apresenta os dados referentes aos vídeos com diferentes qualidades. Nesta *tag* são representadas informações como, por exemplo, resolução do vídeo, largura de banda necessária para transmitir o segmento ou *frameRate*.
* *BaseURL*: apresenta a diretoria do vídeo a ser utilizado para os segmentos seguintes;
* *SegmentURL*: apresenta informações referente ao ponto inicial e final do conteúdo do segmento sobre o conteúdo global do vídeo.

Após adquirir o ficheiro MDP (através de HTTP), o cliente é capaz de perceber qual é a melhor qualidade para iniciar o *streaming*. Posteriormente, o cliente continua a avaliar as condições da rede e consoante os resultados pode decidir aumentar ou diminuir a qualidade atual do vídeo.

Uma imagem com texto

Descrição gerada automaticamente

**Questão 5: Compare o cenário unicast aplicado com o cenário multicast. Mostre vantagens e desvantagens na solução multicast ao nível da rede, no que diz respeito a escalabilidade (aumento do nº de clientes) e tráfego na rede. Tire as suas conclusões.**

O cenário *unicast* é usado para enviar pacotes IP para um único *host* na rede, ou seja, apenas irá existir um remetente e um recetor. Nesta questão o remetente é o servidor e o recetor é o portátil 3.

Uma imagem com mesa

Descrição gerada automaticamente

Como se pode observar na figura anterior, a origem e o destino são sempre os mesmos, sendo os endereços identificados de forma inequívoca.

O cenário *multicast*, ao contrário do *unicast*, irá transmitir os pacotes IP para mais que um *host* na rede, simultaneamente, originando uma topologia com orientação *‘1-to-many’*. Durante uma transmissão *multicast*, o remetente irá enviar os pacotes uma única vez, ficando ao cargo dos recetores captarem e reproduzirem a transmissão entre si.

Uma imagem com mesa

Descrição gerada automaticamente

Analisando a imagem, percebe-se que ao contrário do que acontecia no *unicast*, o endereço de destino não representa nenhum *host* da rede, mas antes o endereço de grupo 224.0.0.100.

Neste tipo de transmissão, o aumento do número de clientes não irá provocar um congestionamento na rede, uma vez que será da responsabilidade dos clientes replicar e enviar os pacotes uns para os outros.

Porém, como o cenário *multicast* utiliza UDP, não há controlo de fluxo nem correção de erros, podendo assim haver duplicação de pacotes, pelo que pode originar um congestionamento na rede.

# Conclusão

Durante a realização deste trabalho, verificou-se técnicas para realizar o *streaming* de vídeos, sejam elas a pedido ou em tempo real.

Primeiramente, utilizando o VLC, experimentou-se o *streaming* de um vídeo, para diferentes plataformas, como por exemplo outro *player* VLC ou o Firefox (*browser*). Para cada umas das soluções analisou-se a taxa em bps necessária para realizar o *streaming*.

Posteriormente, recorreu-se ao DASH para realizar a transmissão de vídeos a pedido. A utilização desta técnica, permite ao cliente adaptar a qualidade de vídeo consoante a qualidade da sua ligação, permitindo assim assisti-lo sem um número muito elevado de *freezes*, uma vez que, de forma automática, a qualidade do vídeo baixa, no caso de a qualidade da ligação piorar.

Por fim, verificou-se as diferenças entre os dois tipos de transmissão – *unicast* ou *multicast* – referentes ao congestionamento e escalabilidade.