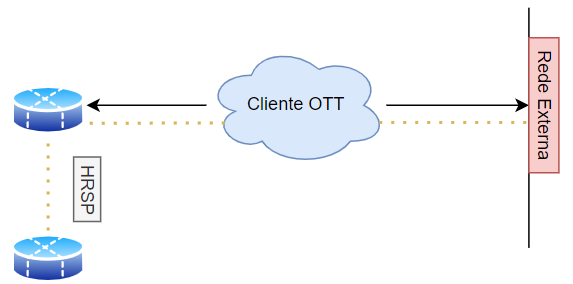
**Passo 1**: Comunicação routers de entrada



**Fig1. Construção da rede - RT e HRSP**

**Nota 1:** Esta configuração não está restrita a versões específicas de software e hardware.

**OBS**: Hot Standby Router Protocol (**HSRP**) – é um protocolo de redundância proprietário do cisco para estabelecer um gateway padrão tolerância a falhas.

Uma maneira de atingir quase 100% de tempo de atividade da rede é usar o **HSRP (Hot Standby Router Protocol)**, que fornece redundância de rede para redes IP, garantindo que o tráfego do usuário se recupere de forma imediata e transparente de falhas de primeiro salto em dispositivos de borda de rede ou circuitos de acesso.

Ao compartilhar um endereço IP e um endereço MAC (Camada 2), dois ou mais roteadores podem atuar como um único roteador "virtual". Os membros do grupo de roteadores virtuais trocam continuamente mensagens de status. Dessa forma, um roteador pode assumir a responsabilidade de roteamento de outro, caso fique fora de serviço por motivos planejados ou não. Os hosts continuam a encaminhar pacotes IP para um endereço IP e MAC consistente, e a troca de dispositivos que faz o roteamento é transparente.

**Passo 2**: Clientes OTT

A tecnologia OTT (Over-the-top) permite que provedores de conteúdos e emissoras entreguem vídeo em vários dispositivos usando a internet.

O OTT envolve serviço de mídia de Streaming que distribui conteúdo via internet, ou seja, o conteúdo é entregue através de outra plataforma (média) “por cima” de outros dispositivos.

1. Os clientes fazem primeiro pedido ao DNS para saber aonde podem chegar dentro da “rede” a um determinado serviço ex: pedido de Stream;
2. O DNS responde sempre com 4 (quatro) IPs a cada novo pedido (configuração Round Robin), ex: 1234. 2341, 3412, 4123 sucessivamente.

**002.frontends.conf**

frontend cdn\_fe

bind 10.151.129.70:80

bind 10.151.129.71:80

bind 10.151.129.72:80

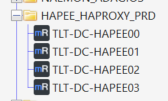
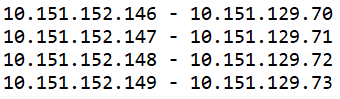
bind 10.151.129.73:80

**Nota**: Verificar ficheiro de configuração: **happee-lb.cfg.d / 002.frontends.conf**

**Nota**: Cada happee tem um VRRP amarado: ex: happee 00: x70, happee 01: x71, happee 02: x72, happee03: x73

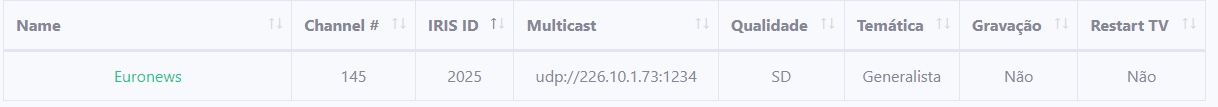
O **VRRP** trabalha de forma virtual em cada um dos Load Balancers, se um LB ficar offline o IP do LB correspondente passa para outro assumindo-o.

Ex: LB1\_VRRP\_x70 -> Offline, LB2: assume VRRP\_x70 do LB1 e o seu próprio VRRP. Se três LBs ficarem offline e tiver um Online, os seus respetivos VRRPs passam para o LB ativo assumindo assim os seus VRRPs de forma a responder por eles até que estes estejam ativos.

**Fig2. IP HAPROXY e VRRP**

1. Conforme linha do ficheiro frontend do cliente, é feito o pedido do canal que se encontra dentro do ficheiro **003.backends.conf** pede o canal requisitando ao host onde fica amarrado. No parâmetro sublinhado amarelo fica amarrado o número do canal ex: \_**2025**\_ para verificar de que canal se trata devemos verificar no **CLU DIGITAL.**



use\_backend channel**\_%[req.hdr(host),lower,map(/etc/hapee-2.2/channels\_be\_map,error)]\_be**

1. O canal esta a ser servidor por cinco servidores onde o Load Balancers faz a sua distribuição nos Transcoder e agarrasse no pedido de um determinado cliente.

backend channel\_2025\_be

balance source

option httpchk

http-check send meth GET uri /ZAP/2025.stream\_zap/manifest.mpd

http-check expect status 200

default-server inter 10s fall 2 rise 3

http-response del-header Server

http-response del-header X-AspNet-Version

stick-table type ip size 5m expire 30m peers ZAP-peers

stick on src

http-request set-header Host 2025.cdn.iptv.zap

server TLT-DC-WTFARM01 10.151.137.17:1935 check

server TLT-DC-WTFARM03 10.151.137.19:1935 check

server TLT-DC-WTFARM05 10.151.137.21:1935 check

server TLT-DC-WTFARM07 10.151.137.23:1935 check

server TLT-DC-WTFARM09 10.151.137.25:1935 check

Nota: Verificar ficheiro de configuração: **happee-lb.cfg.d / 004.backends.conf**

**Nota**: O Load Balancers, faz constantemente helfcheck nos servidores, verifica a Stream e pede o manifesto nos HLS. No manifesto conseguimos verificar a origem dos pedidos.

**helfcheck nos servidores**:

http-check send meth GET uri /ZAP/2042.stream\_zap/manifest.mpd

**Código 200 de entrega**:

http-check expect status 200

**Se o servidor falhar por 10s duas vezes fica fora do cluster e Offline:**

default-server inter 10s fall 2 rise 3

**HLS** é o protocolo de transmissão ao vivo.

1. Antes disso quem se encarrega de direcionar os pedidos é os load balancers os tais hapee. Temos 4 (quatro) LBs que tem a missão de assumir as tolerâncias a falha dependendo dos impactos que possa vir a surgir dentro da rede.

Ex: um servidor que faz um reboot inesperado no momento do pedido a “Stream” o pedido é reencaminhado para outro e processo de entrega continua.

**Passo 3**: WOWZA Transcoding Farm

Os transcoders e os happee estão montados dentro de um enclosure da DELL “Blade server”. Os Transcoder fazem duas coisas: Transcoder e package. A Zap não reduz as Stream pois o acordo com a emissora é para aumentar a qualidade e não para reduzi-la.

A codificação depende da tecnologia de compactação codecs “(coder – decoder) ou compressor – decompressor)” áudio codec: AAC e vídeo H.264.

A transcodificação de vídeo pode envolver alterações menores, como adicionar marcas d´água e gráfico. E depois de transcodificar o conteúdo para torna-lo compatível com tipos de protocolos que os dispositivos do publico usamos os protocolos: HLS e MPEG-DASH.



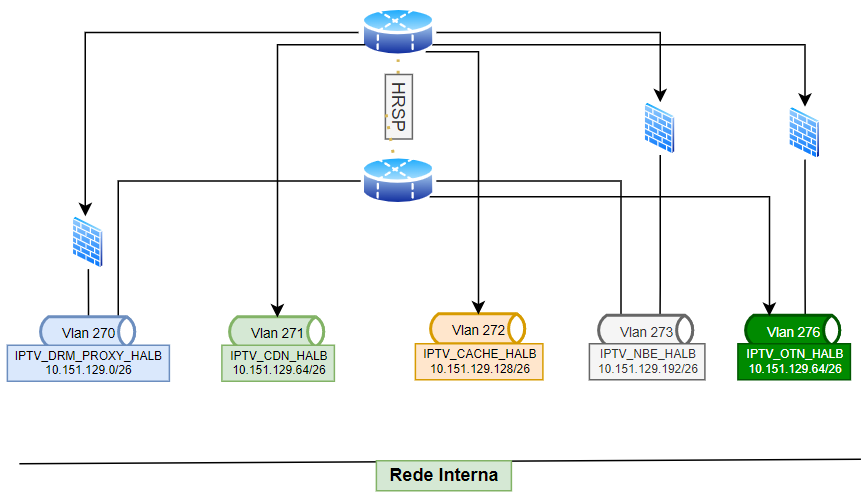
**Passo 4**: Cach

Usamos o “Cach” na frente para reduzir o número de pedidos aos transcoding. As máquinas WTFARMs recebem os pedidos para fazer Transcoder e enviar aos CDNs.

**Passo 2**: Construção das Vlans e Firewall

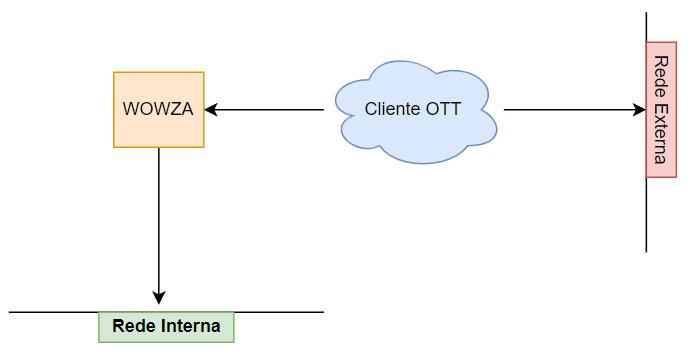
As **Vlans** permitem a segmentação das redes físicas, sendo que a comunicação entre máquinas de Vlans diferentes terá de passar obrigatoriamente por um router ou outro equipamento capaz de realizar encaminhamento (routing), que será responsável por encaminhar o tráfego entre redes (Vlans) distintas.

**Firewall** é uma solução de segurança baseada em hardware ou software (mais comum) que, a partir de um conjunto de regras ou instruções, analisa o tráfego de rede para determinar quais operações de transmissão ou recepção de dados podem ser executadas. "Parede de fogo", a tradução literal do nome, já deixa claro que a firewall se enquadra em uma espécie de barreira de defesa. A sua missão, por assim dizer, consiste basicamente em bloquear tráfego de dados indesejado e liberar acessos bem-vindos.



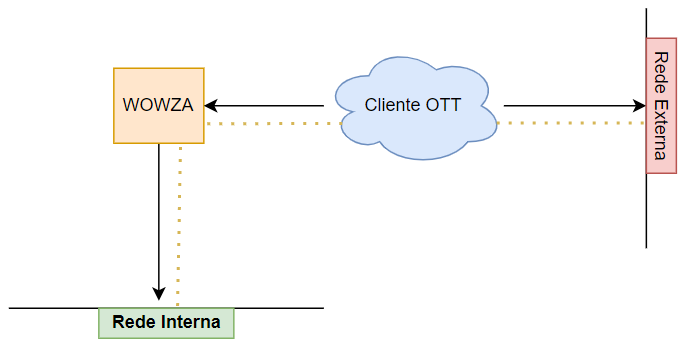
**Fig2. Construção das Vlans e Firewall**

**Passo 3**: Construção do Wowza



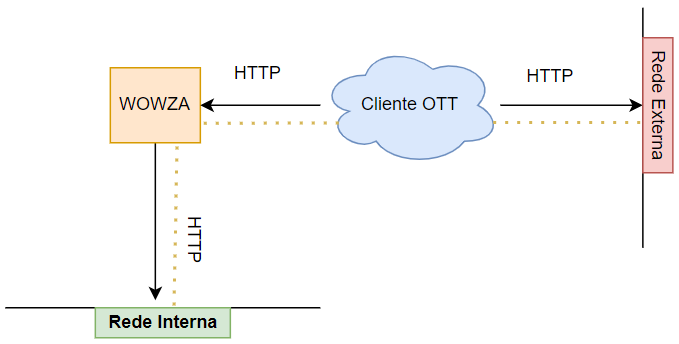
**Fig1. Entrega de conteúdo direto ao cliente**

**OTT** (Over The Top/A cima do Topo) é um termo que descreve a distribuição de conteúdos através da internet, sob demanda e de forma direta, sem a curadoria de intermediários, como os meios de comunicação de massa.



**Fig2. Intermediário na internet “Zap Fibra”**

A Zap Fibra funciona como intermediário na internet (provedor de serviços) colaborando para o uso da rede, de forma ampla. Ou seja, conecta outras empresas e/ou o usuário final à rede mundial de computadores, permitindo desde o acesso à web até a contratação de determinada velocidade para navegação, hospedagem de sites, compartilhamento de informações, etc.



**Fig3. Protocolos e Conexão**

O Alvo de uma requisição HTTP é chamada de “**Resources**”, ou recursos em português, com a natureza ainda não definida; isso pode ser um documento, uma foto, um vídeo ou qualquer outra coisa qualquer. Cada recurso é identificado por uma Identificação de recursos uniforme, do inglês **Uniform Resource Identifier** (**URI**) usada pelo HTTP para identificar recursos.

**Nota**: A identidade e a localização de recursos na Web são fornecidas, principalmente por um único URL (**Uniform Resource Locator**, um tipo de URI).

A URI (**Uniform Resource Identifier**, ou Identificador Uniforme de Recursos) é uma “**string”** (sequência de caracteres) que se refere a um recurso.

[http://www.exemplo.com:80/pasta/para/meuarquivo.html?chave1=valor1&chave2=valor2#AlgumaCoisaNoDocument](http://www.exemplo.com:80/pasta/para/meuarquivo.html?chave1=valor1&chave2=valor2#AlgumaCoisaNoDocumento)

* **http://** é o protocolo.
* **www.example.com** é o nome de domínio
* **:80** é a porta nesta instância
* **/path/to/myfile.html** é o caminho para o recurso no servidor Web
* **?key1=value1&key2=value2** são parâmetros extras fornecidos ao servidor Web
* **#SomewhereInTheDocument** é uma âncora para outra parte do próprio recurso.

**Wowza Streaming Engine** é a ferramenta usada para fazer a transmissão, é uma ferramenta paga por instância mensal é relativamente cara, mas entrega uma performance excecional e é bastante completa. Ela é usada para transmissões ao vivo, literalmente desde a emissora de TV pode-se configurar dentro do Wowza até um serviço de live stream com o Instagram até o próprio Youtube tem um serviço de live.

**Atenção**: Para que seja instalado na sua **VPS** a licença poderá a ter um custo aproximado de **2.500,00 USD**

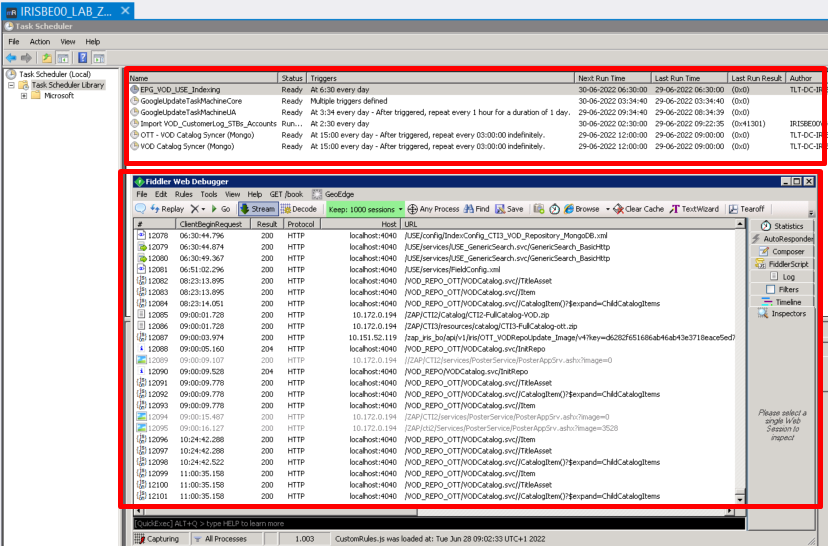
**Nota: VPS** (Virtual Private Server**)**, ou seja, Servidor Virtual Privado é um servidor físico que é dividido entre outros virtuais, porém é diferente do compartilhado, já que por meio da virtualização ele proporciona uma hospedagem com recursos dedicados e personalizáveis.

Com o **VPS** conseguimos acessar a raiz **Apache**, o **PHP**, instalar o **certificado SSL**, entre outras funções, sem a necessidade de mudar para o plano dedicado.

**Passo 4:** Fiddler Web Debugger

Captura pacotes com erros em **http** e **https.** A Zap desenvolveu scripts de reconstrução para a correção das string. Serviços: **EPG\_VOD\_Syncer** e sua instalação é feita no servidor Windows. Todos os dias do lado dos **FE [IRISFE00, EVTFE00, AAFE00]** existe uma ligação a **NOS** para ir a busca dos **EPG**.

* O fornecedor ZAP de **EPG** é a **NOS**.
* **IRIS** – utiliza STV RF Overlay e através das antenas recebe os “**lives**”. Os Servidores **WTFARM00** á 09 são os responsáveis pela gestão, transcodificação e sincronização dos lives. Recebe também neste ponto os pedidos de “**RestartTV**” pelo RF.
* **TIME WORK** – é Restart, pause dos arquivos já passados.
* **VOD CLIENT** – Ponto onde esta ligado os IPs. É o primeiro contacto das STVs e feito com a **NOS**.
* **NDS** – são as máquinas que contém a parte de catálogos e onde os **EPG** e **VOD** são carregados nível de IRIS. Atualmente existe 3 (três) servidores de NDS [**TLT-DC-IRISFE00**, **TLT-DC-IRISFE01**, **TLT-DC-IRISFE02**]



Temos os serviços do fiddler instalados em 4 (quatro) servidores: **AAFE00v4**, **IRISBE00v4**, **NPVRBE00v4** e **PZFFE00v4**.

**Passo 4:** MONGO vs INVENTA

**MongoDB** é um software de banco de dados orientado a documentos livre, de código aberto e multiplataforma, escrito na linguagem C++. Classificado como um programa de banco de dados NoSQL, o MongoDB usa documentos semelhantes a JSON com esquemas.

O **INVENTA** é usado para troca de mensagens entre os vários serviços.

O Mongo e Inveta geram os bugs a nível de repositório:

* EPG
* VODs
* SQL/Front End

**Servidores Virtuais – VPS**

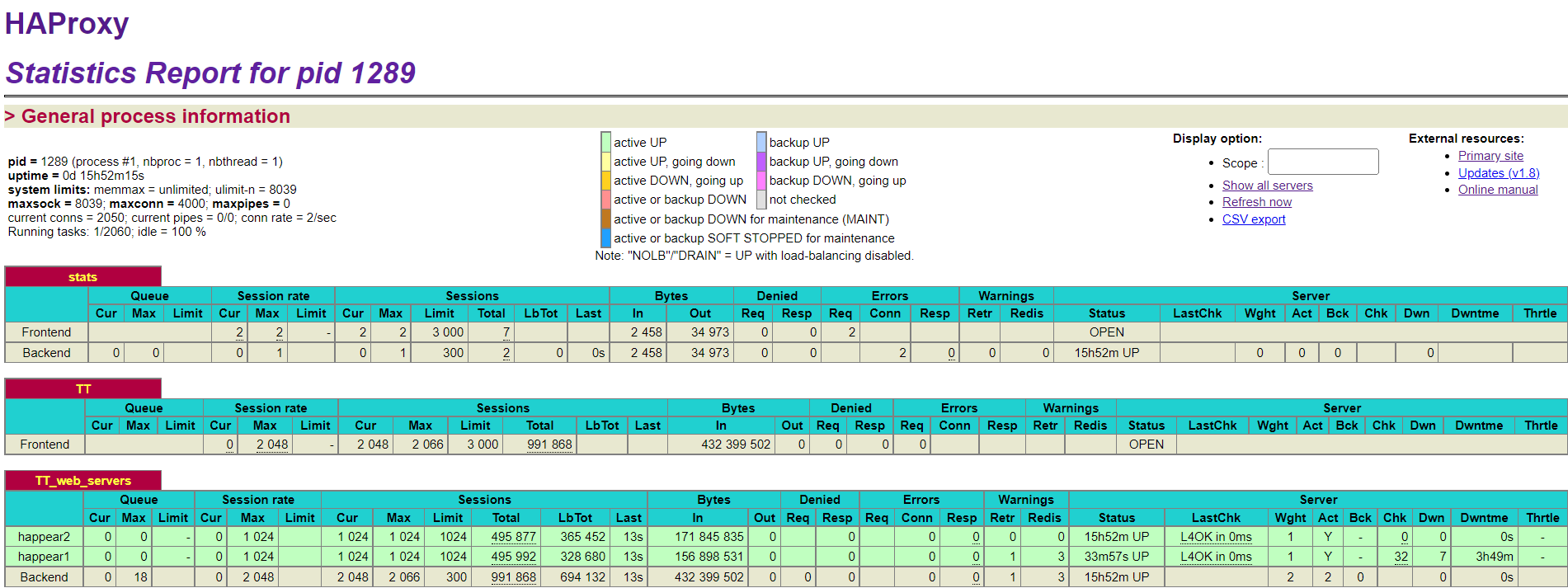
Tecnicamente os Servidores Virtuais ou VPS funcionam em Servidores Dedicados de elevada capacidade e redundância que são depois divididos em pequenas partes. Cada uma destas partes (VPS) está isolada das demais, tem recursos alocados (RAM, CPU e HDD), bem como o seu próprio sistema operativo.

Da perspetiva da redundância e do failover, embora falemos de um serviço estável e flexível, não podemos esquecer que os VPS dependem de um único hardware físico e não de vários servidores, o que numa situação extrema poderá originar falhas de serviço quer seja por intervenções programadas ou falhas inesperadas e incluírem backups diários para storage remoto.



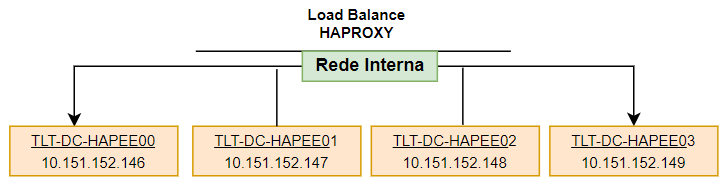
**Fig4. VPS em “Servidores dedicados”**

**Passo 4**: Construção do HAPROX (Load Balance)



**Fig4. Load Balance” Statistics Report”**

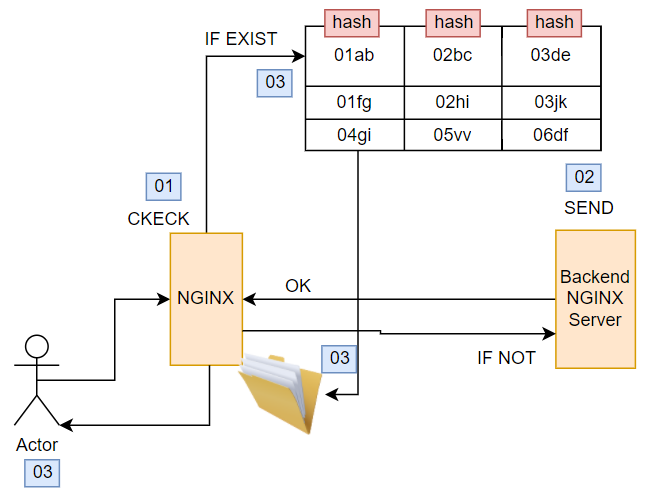
O **HAProxy** é um **proxy reverso** gratuito, muito rápido e confiável que oferece alta disponibilidade, balanceamento de carga e proxy para aplicativos baseados em TCP e HTTP.



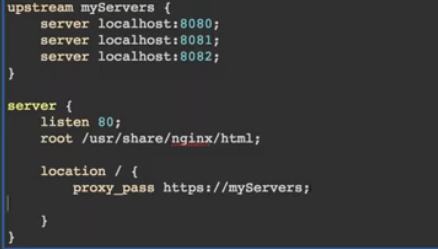
**Fig6. Estrutura Load Balanced**

**Passo 5**: Construção do NGINX

**NGINX**, pronunciado “***engine-ex***,” é um software de código aberto para servidores web lançado originalmente para navegação HTTP. Porém, ele também funciona como proxy reverso, balanceador de carga HTTP, e proxy de email para os protocolos IMAP, POP3 e SMTP.



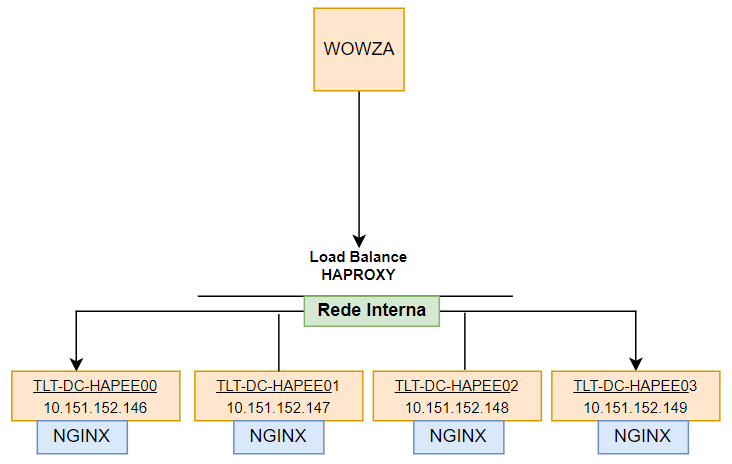
**Fig5. NGINX funcionamento do cache**



* Configurar o Load Balancing [**upstream** – Pool Servers];

**Passo 6**: Construção do NGINX como cache Wowza

O servidor Wowza pode transcodificar e servir seu fluxo HLS/DASH, mas às vezes servir os arquivos para muitos usuários da própria origem pode sobrecarregar o servidor Wowza, que já está ocupado transcodificando o vídeo. Para aumentar a quantidade de usuários que você pode atender, uma abordagem comum é configurar servidores de cache (um ou mais) para reduzir a carga de servir os arquivos do servidor Wowza para os servidores de cache.



**Fig5. NGINX como cache quente para Wowza**

No entanto temos como exemplo, uma configuração padrão que pode ser compilada a partir dos servidores:

* 1 x transcodificador Wowza
* 4 x servidores de cache NGINX (Rocky Linux)

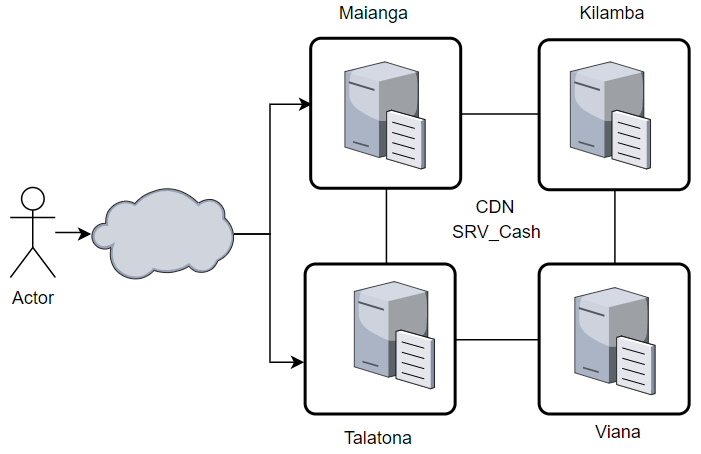
Neste exemplo, os servidores estão conectados à mesma rede local, embora também possam ser remotos e o endereço IP deva ser alterado, respetivamente.

**Passo 7**: Construção Server Cash

**CDN** [Stream que já passaram] – Responsável pelos conteúdos já passados (Servidor de conteúdos passados). **CDN**: **TLT-DC-CDN00**, **TLT-DC-CDN01** e **TLT-DC-CDN02** – Bate nos CDN quando o programa esta completamente gravado.

**CDN** é projetada para contornar o congestionamento da rede e ser resiliente contra a interrupção do serviço. Possui a característica de manter o conteúdo do site on-line diante dos problemas comuns de rede, incluindo falhas de hardware e congestionamento da rede.

Uma CDN também usa balanceamento de carga para fazer alterações com rapidez e eficiência quando a disponibilidade dos recursos do servidor aumenta ou diminui. No caso de um servidor falhar e ocorrer failover, um balanceador de carga redirecionará o tráfego alocado para o servidor com falha e o distribuirá proporcionalmente pelos servidores restantes.



**Fig5. CDN – Rede de Conteúdos “Pops”**

**Passo 8**: Construção Wowza Streaming Transcoder benchmarks: Transrate 720p “Cenário”

**Transcodificação** é a conversão direta de digital para digital de uma codificação para outra, como arquivos de dados de filmes, arquivos de áudio (por exemplo, MP3, WAV) ou codificação de caracteres (por exemplo, UTF-8, ISO/IEC 8859) Isso geralmente é feito nos casos em que um dispositivo de destino (ou fluxo de trabalho) não suporta o formato ou possui capacidade de armazenamento limitada que exige um tamanho de arquivo reduzido, ou para converter dados incompatíveis ou obsoletos em um formato moderno ou com melhor suporte.

**OBS**: A transcodificação é geralmente um processo com perdas, apresentando perda de geração; no entanto, a transcodificação pode ser sem perdas se a saída for compactada ou descompactada sem perdas. O processo de transcodificação para um formato com perda introduz vários graus de perda de geração, enquanto a transcodificação de com perda para sem perda ou sem compressão é tecnicamente uma conversão sem perda porque nenhuma informação é perdida; no entanto, o processo é irreversível e é mais corretamente conhecido como destrutivo.

A **transcodificação**, em alto nível, está pegando conteúdo já compactado (ou codificado); descompactá-lo (decodificá-lo); e então de alguma forma alterando e recomprimindo-o. Como exemplo, você pode alterar o formato de áudio e/ou vídeo (codec) de um para outro, como converter de uma **fonte MPEG2** (comumente usada em transmissão de televisão) para **vídeo H.264** e **áudio AAC** (os codecs mais populares para transmissão). Outras tarefas básicas podem incluir adicionar marcas d'água, logotipos ou outros gráficos ao seu vídeo.

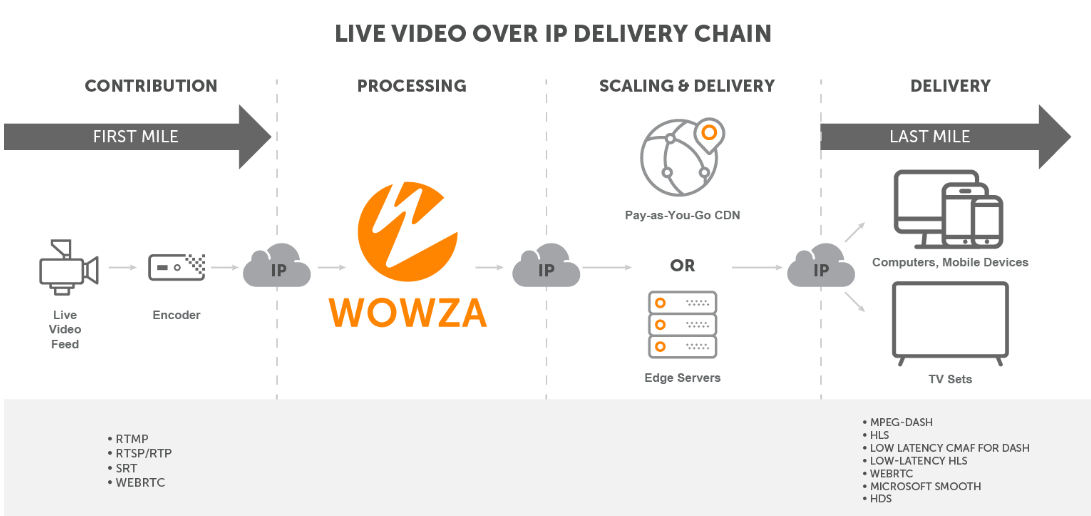
**Transrating** refere-se especificamente à alteração das taxas de bits, como pegar um fluxo de entrada de vídeo 4K a 13 Mbps e convertê-lo em um ou mais fluxos de taxa de bits mais baixa (também conhecidos como renderizações): HD a 6 Mbps ou outras renderizações a 3 Mbps, 1,8 Mbps, 1 Mbps, 600 Kbps, etc.

**Transsizing** refere-se especificamente ao redimensionamento do quadro de vídeo, digamos, de uma resolução de 3840×2160 (4K UHD) para 1920×1080 (1080p) ou 1280×720 (720p).

O processo de conversão de vídeo (**transcodificação**) é computacionalmente intensivo, portanto, a transcodificação geralmente requer recursos de hardware poderosos, incluindo CPUs mais rápidas ou recursos de aceleração de gráficos. A transcodificação é feita usando um transcodificador ao “vivo” – que provavelmente assumirá a forma de um servidor de Streaming ou plataforma de Streaming baseada em nuvem.

**Atenção**: A transcodificação não deve ser confundida com **transmuxing**, que também pode ser chamada de **repackaging**, **packetizing** ou **rewrapping**. Transmuxing é quando você pega áudio e vídeo compactados e - sem alterar o conteúdo real de áudio ou vídeo - (re)empacota em diferentes formatos de entrega.

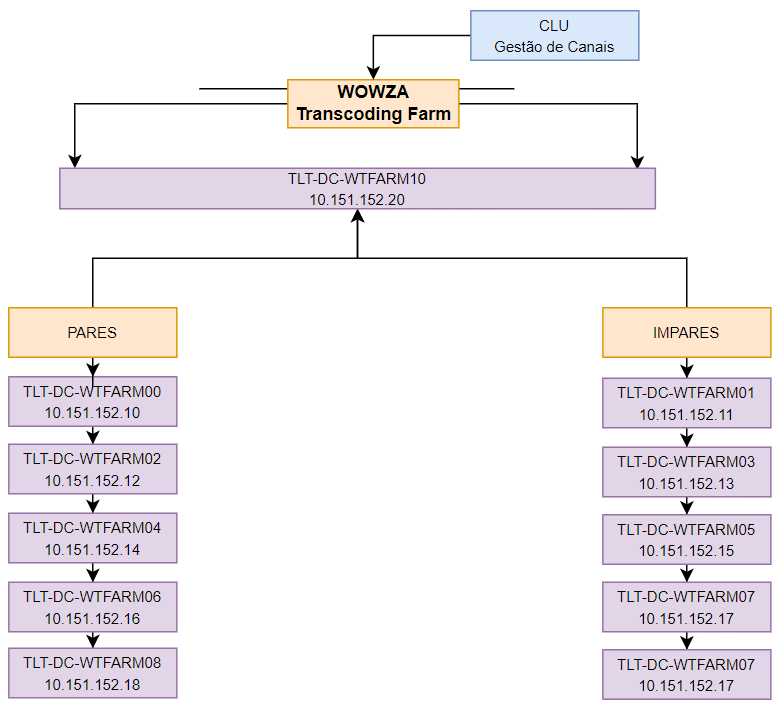
Por exemplo, você pode ter conteúdo **H.264**/**AAC** e, alterando o container em que está empacotado, pode entregá-lo como **HTTP** Live Streaming (**HLS**) ou Dynamic Adaptive Streaming Over HTTP (**DASH**). A sobrecarga computacional para transmuxing é muito menor do que para transcodificação.

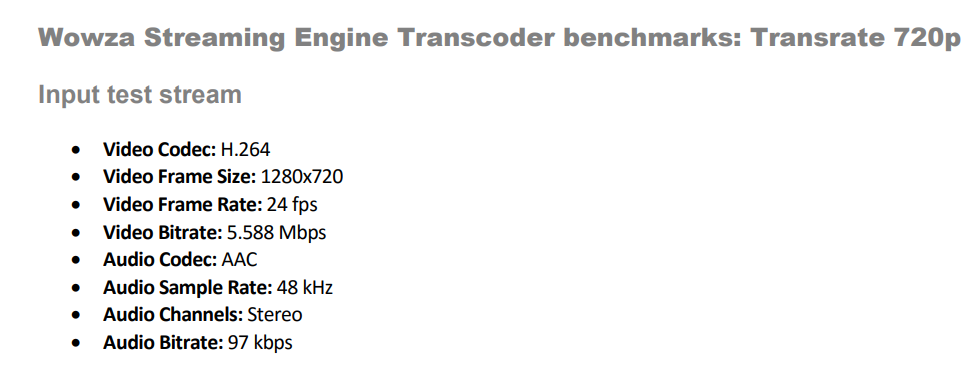


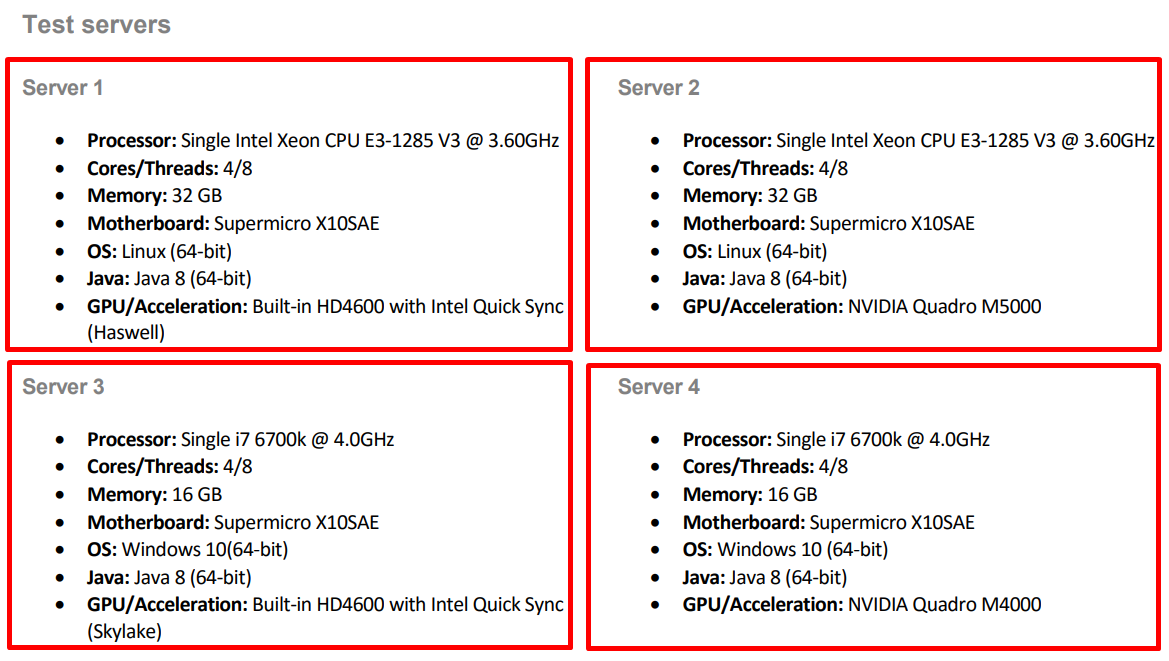
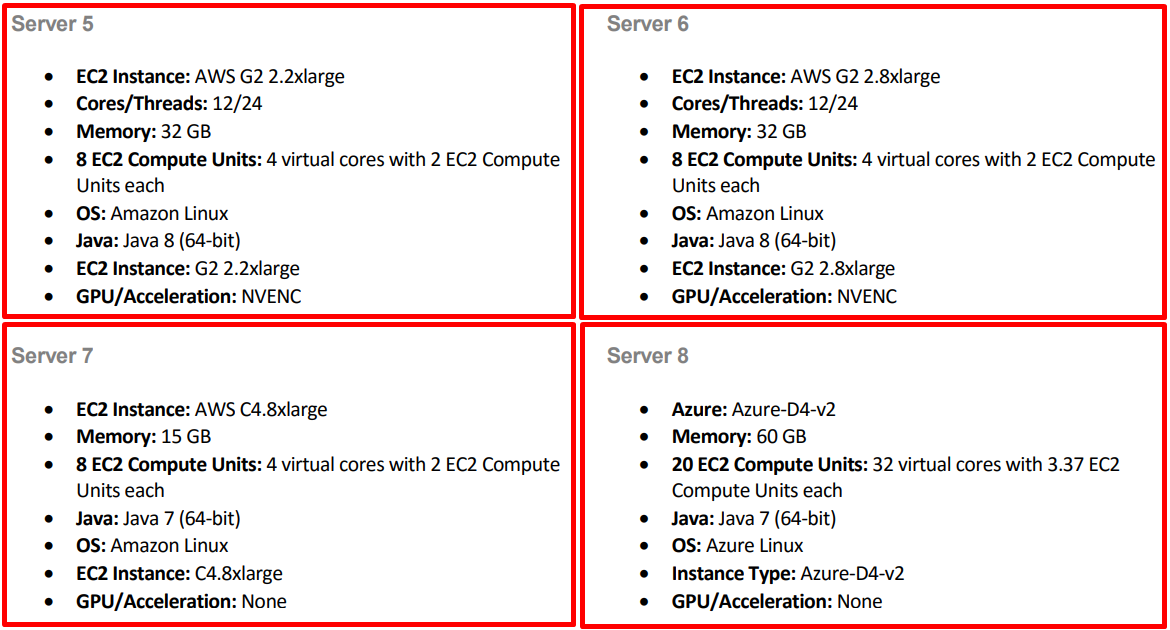
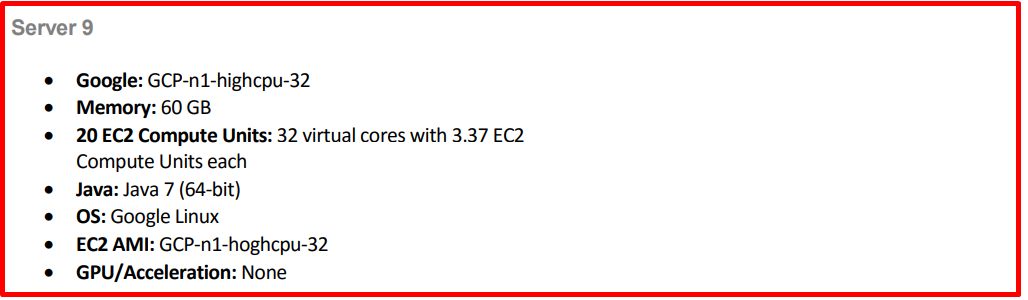
**Fig5. Processo transcodificação**

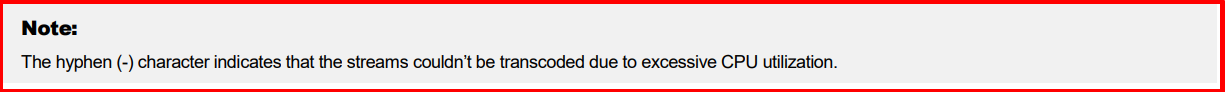
**Passo 8**: Servidores WTFARM

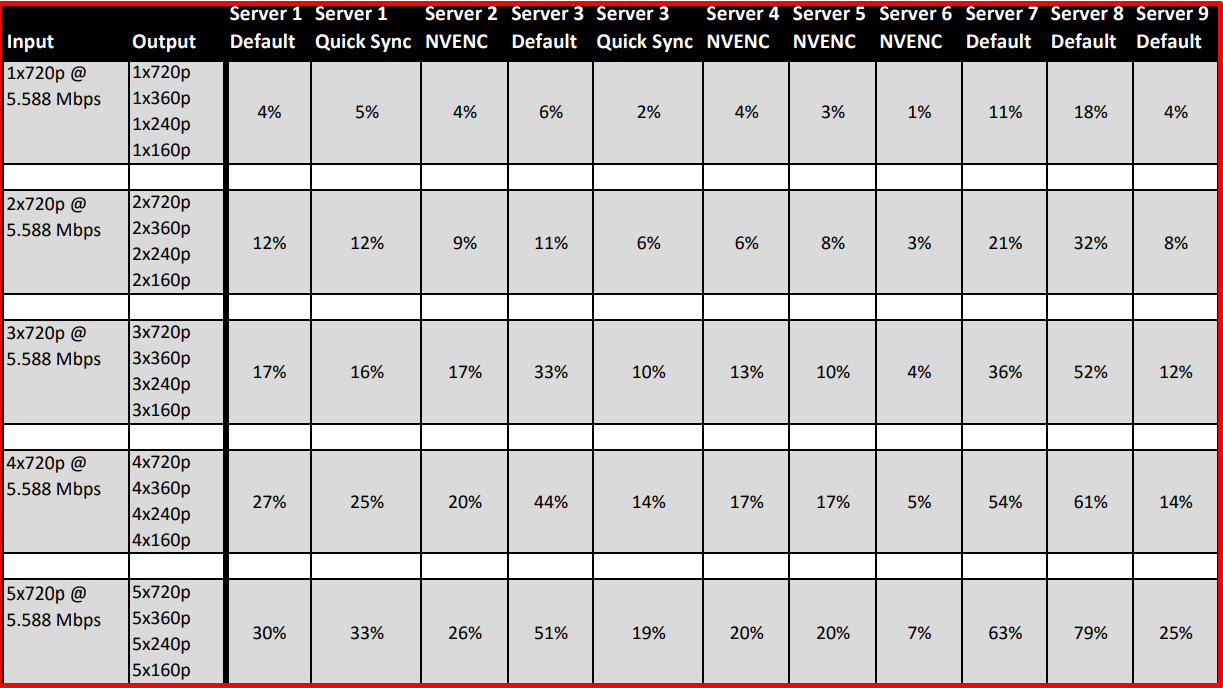
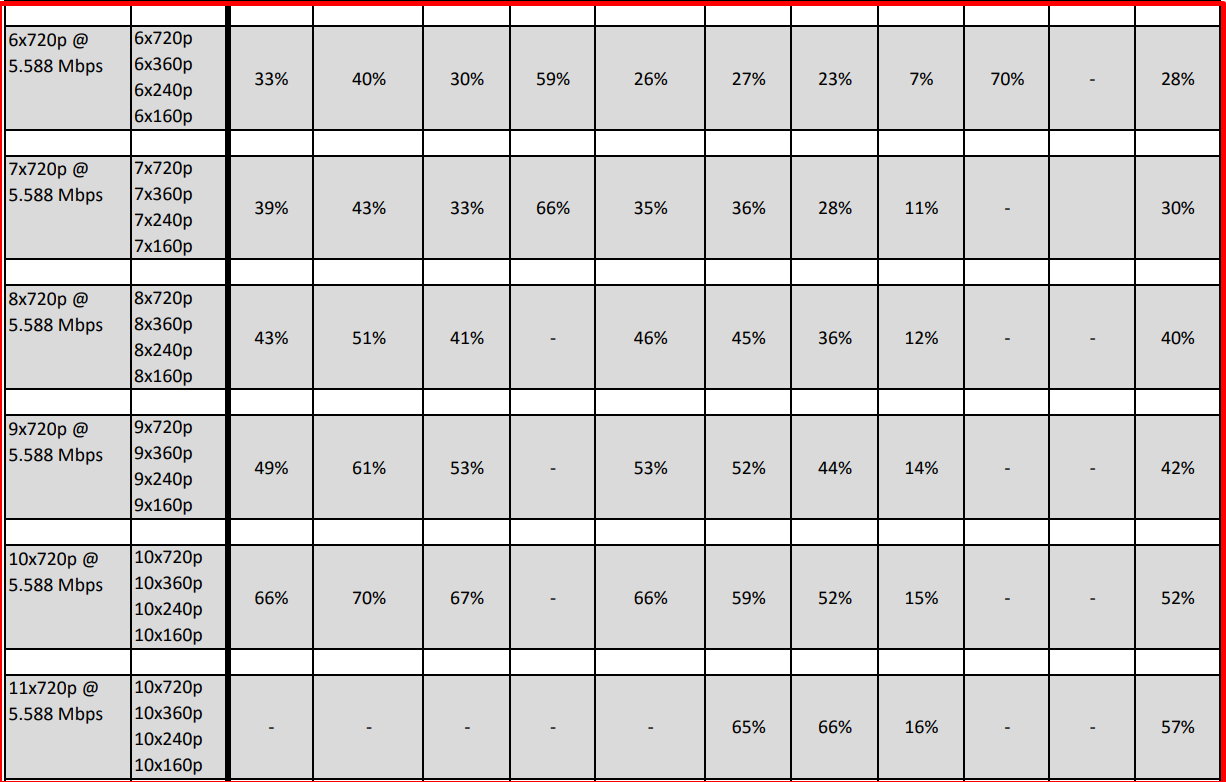
**WTFARM** [Stream Live] – Conteúdos que esta a ser transmitido [Load Balanced – faz a divisão do pedido de forma a não sobrecarregar os servidores]. TLT-DC-WTFARM00 á WTFARM09 – A sua distribuição passa devido aos canais que estão a ser gravados no **ZAPR**.

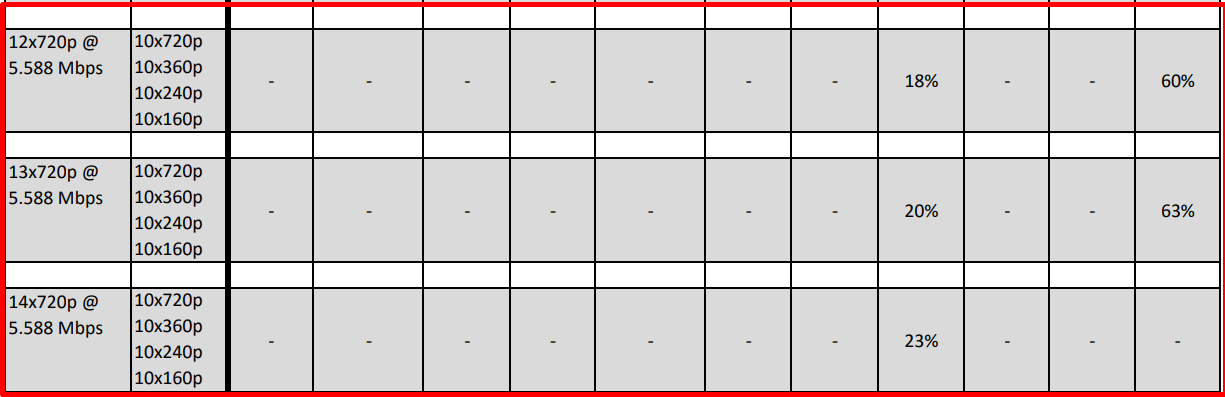


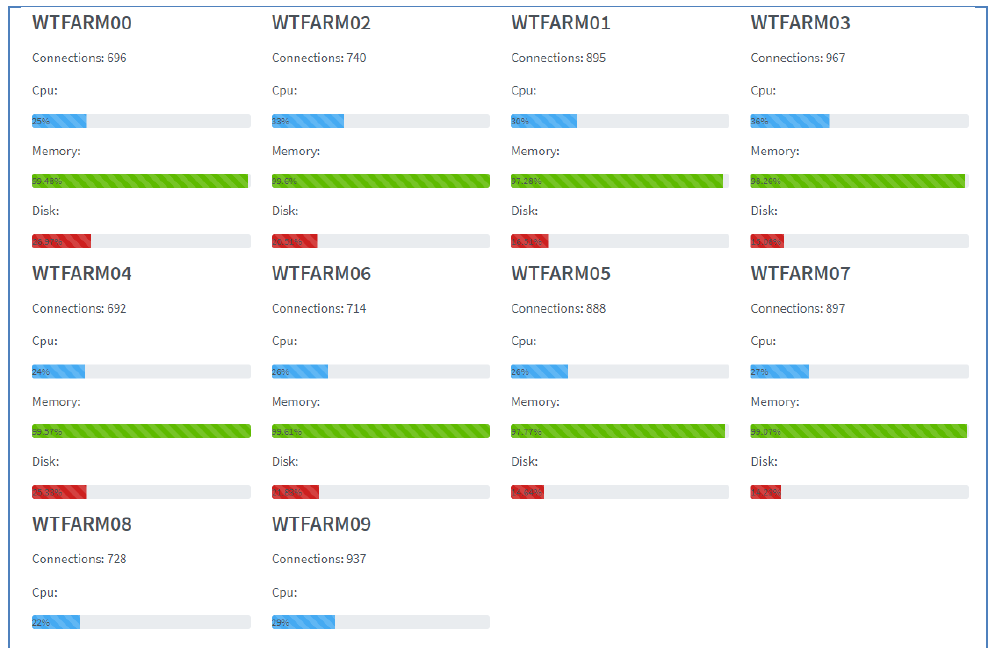




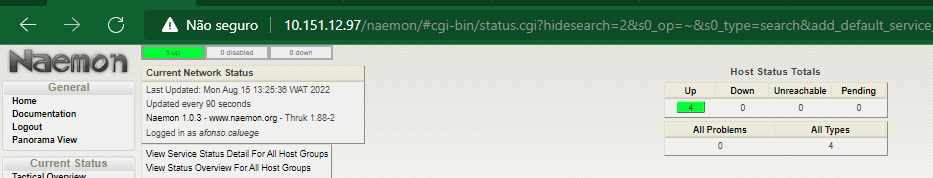
 

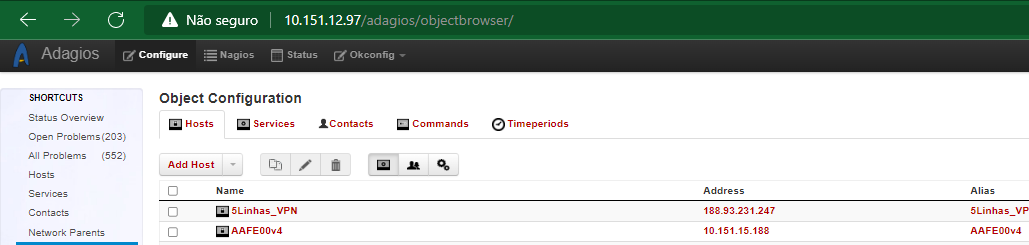




**Passo 9**: Naemon e Adágios

Naemon é uma ferramenta de monitorização e alarmística, responsável pela gestão de todos os equipamentos e serviços na rede. Esta linkada ao Adágios ferramenta da adição das hosts, serviços, contactos, Logs etc.





**Passo 10**: Pastas partilhadas

As pastas partilhadas estão inseridas no SRV 10.151.58.50:8080 designado por Alfresco onde guarda os ficheiros de todas as equipes inclusive a documentação geral.

**Passo 11**: Servidores de Laboratório

Na estrutura ZAP existem 11 servidores para laboratório dentre os quais 3(três) Linux e 8 (Oito) Windows. No Linux temos um servidor que armazena a base de dados **MONGO DB** e um **RTSP DB.**

**Passo 12**: NBE [Direitos/Nordija]

O servidor NBE - TLT-DC-NBE00, TLT-DC-NBE01 e TLT-DC-NBE02 é o responsável pelos direitos onde visualiza se a pessoa tem ou não direitos para visualizar os conteúdos e busca das imagens.

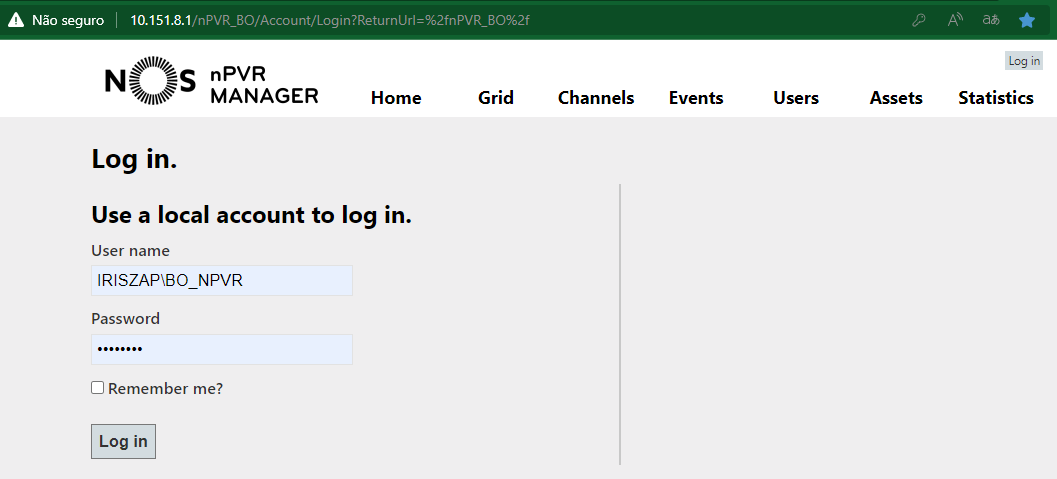
**Passo 13**: Focus\_. Zone [Nordija]

O Focus Zone é o responsável pela gestão das Boxs e visualização dos vídeos.

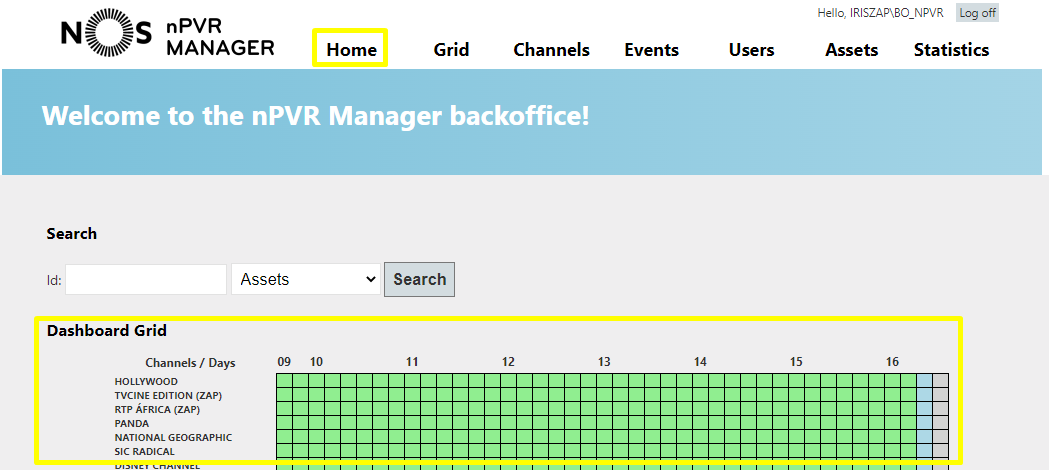
**Passo 14**: nPVR Manager

**NPVR** – Trata da gestão das gravações dos clientes, os 7 (sete) dias gravados e grava a cota do cliente.

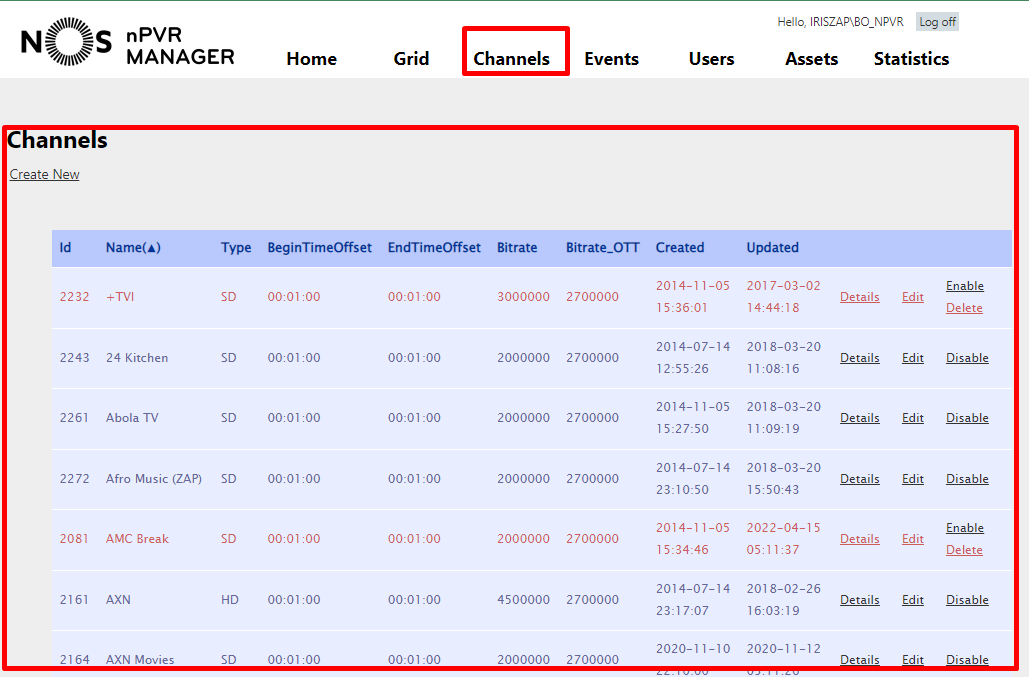
Nela encontramos menu: grelha de canais, gestão da cota dos clientes, eventos, estatísticas de importação de EPG etc.



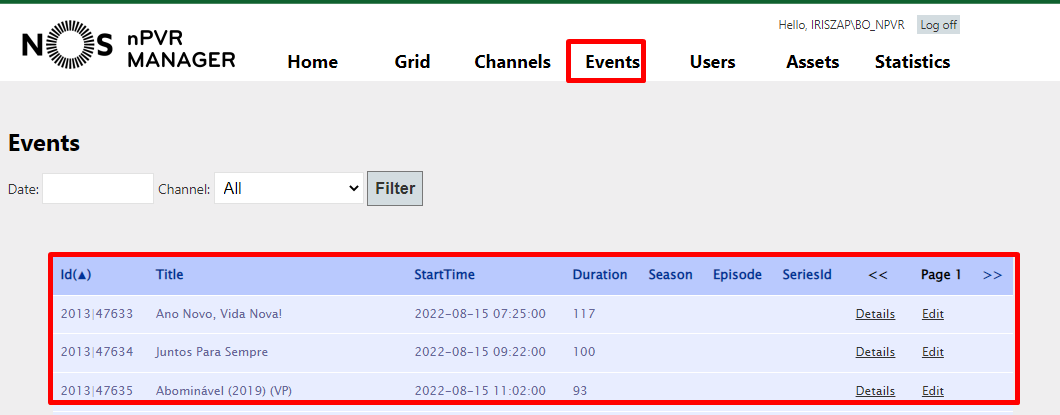
**Fig5. nPVR acesso**



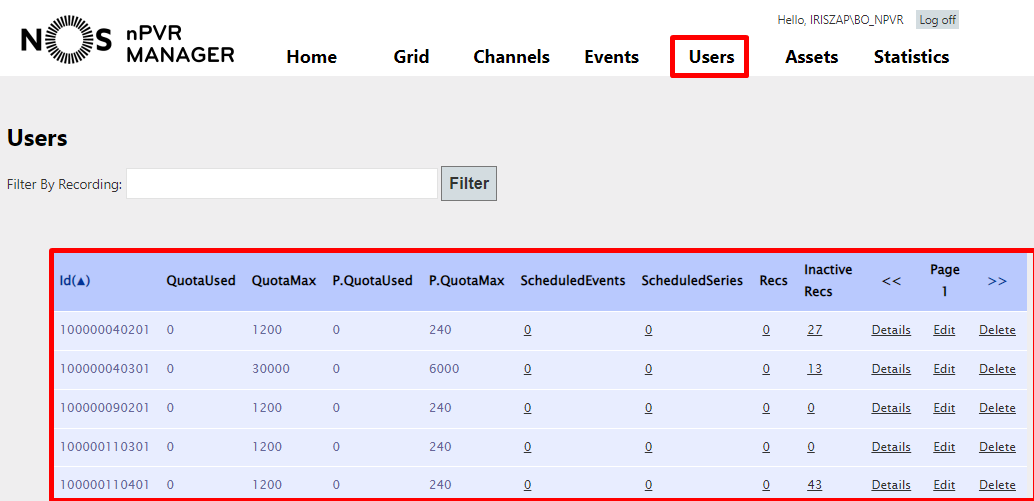
**Fig5. Home**



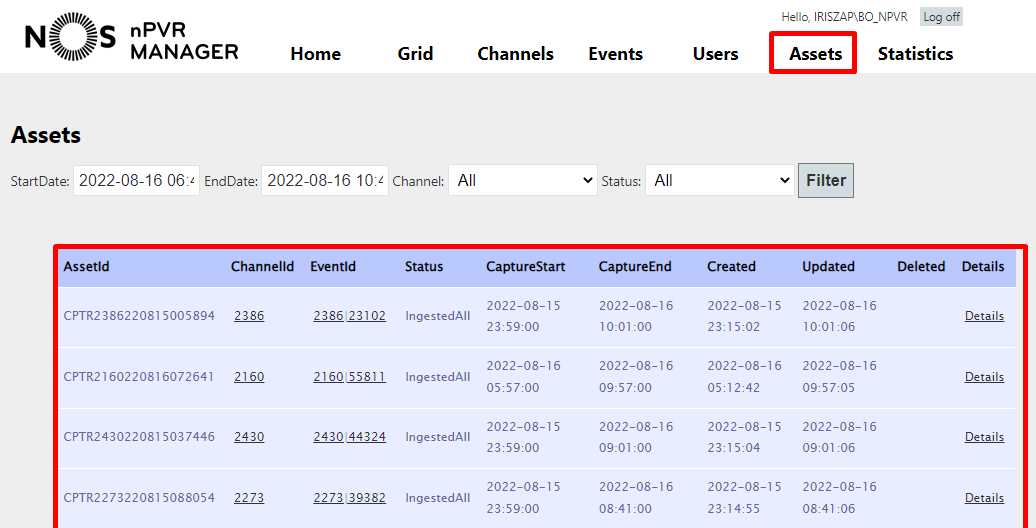
**Fig5. Canais**



**Fig5. Eventos**



**Fig5. Users**



**Fig5. Ativos**

