

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Alexandre Mariano Lopes de Deus

**CONFIGURAÇÃO E ANÁLISE DE TRÁFEGO DA
PLATAFORMA BIGBLUEBUTTON**

UBERLÂNDIA

2019

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Alexandre Mariano Lopes de Deus

Configuração e Análise de Tráfego da Plataforma BigBlueButton

Trabalho de Conclusão de Curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, Minas Gerais, como requisito exigido à obtenção do grau de Bacharel em Sistemas de Informação.

Universidade Federal de Uberlândia – UFU
Faculdade de Computação
Bacharel em Sistemas de Informação

Orientador: Prof. Dr. Luís Fernando Faina

Uberlândia, Brasil
2019

Dedico este trabalho a minha família, namorada, amigos e professores, que estiveram presentes nas etapas mais adversas de minha jornada acadêmica, tanto nos momentos bons e nos mais complexos.

Agradecimentos

Agradeço, primeiramente a Deus, que me proporcionou força e saúde para atingir meus objetivos e metas e tornou cada desafio cumprido muito mais gratificante.

Aos meus familiares, em especial minha avó, que já não está mais entre nós porém prestou todo apoio e fez o possível para que momentos como esse se realizassem em minha vida.

Aos meus professores que contribuíram imensamente provendo conhecimentos teóricos, práticos e foram um dos fatores cruciais para meu desenvolvimento profissional.

Aos meus amigos e colegas de trabalho que sempre estavam presentes nos momentos de superação.

Ao meu orientador Luís Fernando Faina, pelo total apoio prestado e clarificação nos momentos de dificuldade encontrados durante todo este período.

A todos, meus sinceros agradecimentos.

*“A menos que modifiquemos a nossa maneira de pensar, não seremos capazes de resolver os problemas causados pela forma como nos acostumamos a ver o mundo”.
(Albert Einstein)*

Resumo

Com o crescente avanço das tecnologias e as demandas cada vez mais altas por velocidade na transmissão da informação, o mundo se depara com uma necessidade de criar meios de comunicação que encurtem a distância entre as pessoas ao redor do globo. Quase todas as áreas existentes são reféns destes fatores citados, tanto na área acadêmica como na área corporativa entre outras. Um meio que veio para auxiliar e a suprir essa necessidade por comunicação a qualquer hora e em qualquer lugar foi a Conferência Web. Como objetivo de estudo, iremos explorar o BigBlueButton, uma ferramenta de Conferência Web que tem como objetivo principal a aprendizagem *online*. No meio acadêmico o BigBlueButton pode ser uma ferramenta muito poderosa reduzindo as distâncias entre discentes e docentes. O principal objetivo deste projeto é realizar a configuração e suceder análises de canais dos fluxo de mídia e algumas técnicas que o BigBlueButton emprega no controle de congestionamento de tráfego. Este que por vez cresce exponencialmente na medida que integrantes se juntam a uma Conferência Web. Espera-se que com o estudo realizado nos canais de comunicação da ferramenta, possa-se entender melhor o funcionamento de uma Conferência Web a nível de rede e como surge um possível congestionamento neste ambiente na medida que a quantidade de usuários aumenta.

Palavras-chave: Conferência Web. Comunicação. BigBlueButton. Congestionamento. Tráfego. Rede.

Abstract

With the increasing advancement of technologies and the ever-increasing demands for speed in information transmission, the world is faced with a need to create media that bridges the gap between people around the globe. Almost all existing areas are hostage to these factors, both in the academic area and in the corporate area, among others. One means that has come to assist and meet this need for communication anytime and anywhere has been the Web Conference. As a study goal, we will explore the BigBlueButton, a Web Conference tool that is primarily aimed at online learning. In the academic world the BigBlueButton can be a very powerful tool reducing the distances between students and teachers. The main purpose of this project is to perform the configuration and perform analysis of media flow channels and some techniques that the BigBlueButton employs in traffic congestion control. This one that at the same time grows exponentially as members join a Web Conference. It is hoped that with the study carried out in the communication channels of the tool, one can better understand the operation of a Web Conference at the network level and how it arises possible congestion in this environment.

Keywords: Web Conference. communication. BigBlueButton. Congestion. Traffic. Network.

Lista de ilustrações

Figura 1 – Arquitetura do BigBlueButton	19
Figura 2 – Tela de Login do BigBlueButton	20
Figura 3 – Explosão da Arquitetura do Red5	22
Figura 4 – Handshake do Protocolo RTMP	24
Figura 5 – Servidor Aguardando Conexões na Porta 1935	27
Figura 6 – Conexões na Porta 1935 do Servidor	27
Figura 7 – Canais de Comunicação Capturados pelo Wireshark	28
Figura 8 – Comparação do Payload do Pacote de Testes e do Pacote de Controle	28
Figura 9 – Payload do Pacote de Controle	29
Figura 10 – Payload do Pacote de Vídeo	30
Figura 11 – Payload do Pacote de Áudio SIP	32
Figura 12 – Tráfego no Canal de Vídeo	33
Figura 13 – Tráfego no Canal de Controle	35
Figura 14 – Tráfego no Canal de Áudio	36
Figura 15 – Pacotes SIP Capturados Durante Testes Realizados	36
Figura 16 – Tráfego de Compartilhamento de Tela	37

Lista de abreviaturas e siglas

EAD	Ensino a Distância
TCP	Transmission Control Protocol
IP	Internet Protocol
SIP	Session Initiation Protocol
ABED	Associação Brasileira de Educação a Distância
ACK	Acknowledgment
SYN	Synchronize
LAN	Local Area Network
RTMP	Real Time Messaging Protocol
API	Application Programming Interface
QOS	Quality of Service
RTMP	Real Time Monitoring Protocol
ICMP	Internet Control Message Protocol
VOIP	Voice Over Internet Protocol
MOODLE	Modular Object Oriented Distance Learning
SSL	Secure Socket Layer

Sumário

1	INTRODUÇÃO	12
1.1	Objetivos	13
1.2	Organização do Trabalho	14
2	REVISÃO BIBLIOGRÁFICA	15
2.1	Conferência Web no Contexto Educacional	15
2.2	Conferência Web e seus Protocolos	15
2.3	Uso do BigBlueButton no Ensino	17
2.4	Trabalhos Correlatos	18
3	ARQUITETURA DO BIGBLUEBUTTON	19
3.1	Interface do BigBlueButton	20
3.2	Módulos de Processamento de Informação	21
3.3	Servidor Red5	22
3.4	Protocolo RTMP	23
4	ANÁLISE DE TRÁFEGO NO BIGBLUEBUTTON	25
4.1	Instalação e Configuração do Servidor BigBlueButton	25
4.1.1	Repositório do BigBlueButton	25
4.1.2	Configuração do Ambiente para o BigBlueButton	25
4.1.3	Teste de Conexão e Abertura de Canais	26
4.1.4	Canal de Teste de Conexão	28
4.1.5	Canal de Controle	29
4.1.6	Canal de Compartilhamento de Vídeo	30
4.1.7	Canal de Compartilhamento de Áudio	31
4.1.8	Canal de Compartilhamento de Tela	31
4.2	Cenários para Análise de Tráfego no BigBlueButton	32
4.2.1	Análise do Tráfego de Vídeo	32
4.2.2	Análise do Tráfego de Controle	34
4.2.3	Análise do Tráfego de Áudio	35
4.2.4	Análise do Tráfego de Compartilhamento de Tela	37
5	CONCLUSÃO	39
5.1	Trabalhos Futuros	40
	Referências	41

APÊNDICES	43
APÊNDICE A – WIRESHARK	44

1 Introdução

A necessidade em agilizar a troca de informação ao redor do mundo e os crescentes avanços tecnológicos tem demandado uma carência cada vez maior nas pessoas de conseguirem meios eficazes de satisfazer tais necessidades. Embasado neste conceito um meio que vem ganhando muita força é a Conferência Web. Existem diversas ferramentas que provêm este serviço, algumas delas possuem seu código fechado e pertencem a empresas que vendem a Conferência Web como um serviço e, também existem as empresas que fazem uso do *software* livre para criarem seus próprios serviços de Conferência Web.

O *software* livre permite que usuários possam redistribuir, estudar e até mesmo modificar códigos e programas de computador. Tendo como única exigência que o *software* permaneça livre para outros usuários após a redistribuição (COSTA, 2014).

O foco deste trabalho é voltado para a inserção da Conferência Web no meio universitário onde considera-se um cenário em que turmas com vários alunos podem se conectar a fim de estabelecerem uma comunicação audiovisual para a troca de conhecimento. Para isso, é preciso abordarmos o conceito do EAD (Ensino a Distância). Segundo Aretio (1994 apud Guarezi; Matos, 2009) citado por KICH (2015, p.13), "EAD é um sistema em que a interação pessoal na sala de aula é substituída pelo uso de uma organização tutorial e por recursos didáticos, propiciando, em vista disso, uma aprendizagem autônoma."

Segundo Sandhiltz citado por DOTTA (2009, p.02), "o uso da Conferência Web em cursos a distância justifica-se pela necessidade de se fazer uma transição de cursos centrados em conteúdos para cursos centrados no diálogo." A utilização dos EAD's traz benefícios perceptíveis como o aumento da interação entre educador e o aluno, também estimula a motivação e interesse do aluno para com as aulas, e além de tudo ajuda a desenvolver as habilidades de interação social (ABED,2016). Para REIS (2015) o EAD vem sendo muito bem cotado pois ele possibilita e reduz as barreiras de desigualdade social, também permite dar oportunidade e capacitação profissional a muitas pessoas.

Ao adentrarmos no meio da Conferência Web é necessário selecionar um ou alguns dos *softwares* de videoconferência conforme citados, sejam eles de código livre ou não. Neste trabalho iremos adotar como meio de estudo o BigBlueButton, que é uma ferramenta de código livre desenvolvida na Universidade de Toronto em 2007. Os idealizadores do BigBlueButton acreditam que uma experiência de aprendizado *online* de qualidade deve estar acessível a cada aluno que possui um navegador de internet (BIGBLUEBUTTON INC,2015).

Como problema principal a ser abordado neste trabalho, iremos estudar o comportamento da rede durante uma conferência e os congestionamentos gerados durante o tráfego. "Em gerenciamento de redes, um dos grandes problemas enfrentados são os colapsos de congestionamento"(BARCELAR, 2012, p.01). O fluxo de uma conferência além de ser

bidirecional, a informação chega e sai para ambos os lados (transmissor e receptor) simultaneamente, ele também acaba por transportar mídias de áudio e vídeo, entre outros dados (FONSECA e STANTON, 2001).

É possível notar a grande importância de prevenir o congestionamento no tráfego de rede proveniente dos pacotes de uma Conferência Web, uma vez que na medida que o congestionamento cresce acaba-se por impactar não só o próprio serviço mas também outras aplicações que estejam fazendo uso da rede em questão, a mitigação deste problema acaba por permitir uma maior escalabilidade dos serviços, uma vez que é possível integrar mais usuários com qualidade e satisfação trazendo sincronia em áudio e vídeo para obtenção do resultado esperado (FONSECA e STANTON, 2001).

1.1 Objetivos

Analisar e estudar a ferramenta BigBlueButton, a fim de compreender como funcionam os canais de comunicação de mídia presentes em uma Conferência Web e, assim entender como surge o congestionamento neste ambiente e, quais medidas a ferramenta agrega para tentar contornar e garantir a qualidade dos serviços fornecidos.

Para alcançar os objetivos gerais citados, serão estabelecidos alguns objetivos específicos conforme abaixo:

- montar um ambiente de testes que simule a comunicação entre o servidor do BigBlueButton e seus clientes;
- capturar por meio de ferramentas de medição os fluxos de comunicação entre ambos os participantes de uma conferência (clientes e servidor);
- analisar e demonstrar por meio das coletas realizadas como ocorre o congestionamento de rede dentro do ambiente de Conferência Web;
- identificar quais meios e técnicas foram utilizadas na ferramenta BigBlueButton para mitigar tais problemas.

1.2 Organização do Trabalho

O Capítulo seguinte tem como objetivo, especificar o protocolo base de uma Conferência Web e o motivo pelo qual uma Conferência embasada em tal protocolo é ideal para o uso no Ensino; O capítulo 3 irá descrever a Arquitetura do BigBlueButton e especificar cada um de seus módulos mais importantes; Já no capítulo 4 mostra os testes e análises que foram realizados nos canais de mídia do BigBlueButton; Por fim, o capítulo 5 contém os resultados que foram obtidos em virtude das análises realizadas.

2 Revisão Bibliográfica

2.1 Conferência Web no Contexto Educacional

Métodos como a Conferência Web, que visam facilitar a disseminação do conhecimento e melhoria do aprendizado são estudados há bastante tempo. "A partir dos anos 90 e, sobretudo, em virtude da rápida expansão e evolução tecnológica nos meios de informação e comunicação, intensificaram-se as discussões acerca da alteração dos modos de acesso ao saber"(SILVA e PEREIRA, 2002, p.01). Utilizando-se dos meios de transmissão de conteúdo audiovisual é possível criar um leque de possibilidades no meio de estudo como por exemplo, formação de salas *online* para estudo, apresentação de materiais didáticos sem a necessidade de o aluno estar com o conteúdo impresso em mãos e até mesmo na apresentação de trabalhos de conclusão de curso sem uma banca avaliadora fisicamente presente.

Ao confrontar a apresentação da webconferência com os conceitos apresentados pelos teóricos, podemos perceber que através dessa webconferência eles se materializam tendo em vista a forma como foi direcionado, proporcionando um efetivo conhecimento aos alunos, permitindo a eles uma maior interação com os especialistas, tirando suas dúvidas e podendo solicitar mais ferramentas de estudo como livros e dicionários. No entanto, mesmo se não tivesse sido aberto o espaço para interação com os alunos, a forma como foi apresentada proporcionou a transmissão de conhecimento (SILVA e PEREIRA, 2002, p.09).

2.2 Conferência Web e seus Protocolos

Quando se fala de Conferência Web é válido salientar que existe toda uma infraestrutura de rede por trás dos serviços fornecidos, uma vez que o modelo padrão de rede é o TCP/IP(Transmission Control Protocol/Internet Protocol), nos deparamos com um cenário onde é preciso contornar ou pelo menos amenizar os problemas já conhecidos e mapeados acerca deste protocolo que possam trazer conseqüências negativas para o pleno funcionamento da ferramenta utilizada. Como um dos grandes problemas deste protocolo, iremos nos deparar principalmente com o congestionamento de tráfego na rede durante uma Conferência Web.

O congestionamento é um dos problemas mais pesquisados na área de redes de computadores. Ele é um estado atingido pela rede que degrada a qualidade de serviço das conexões estabelecidas e, por isso, os algoritmos de controle de congestionamento nas implementações do TCP tentam minimizar e controlar esse estado da rede (CAVALCANTI, 2005, p.11).

Na medida que a quantidade de participantes conectados a uma Conferência Web cresce, o número de conexões TCP a fim de estabelecer comunicação entre esses diversos participantes também cresce exponencialmente, para entender o conceito de estabelecimento de conexão no TCP é preciso primeiramente entender o que é o Handshake, que segundo

DANTAS (2006) basicamente um computador origem envia um pacote de abertura de sessão com o bit SYN (Synchronize) ligado ao computador de destino. Por sua vez ao receber o pacote enviado da origem, responde com um novo pacote agora contendo os bits SYN e ACK (Acknowledgment) ligados. Por fim o computador origem ao receber este novo pacote do destino, retorna um pacote apenas com o bit ACK ligado confirmando a chegada do pacote para início da sessão de comunicação.

Tendo em mente esse processo de estabelecimento de comunicação, observa-se que em uma Conferência Web esse número de sessões iniciadas cresce exponencialmente na medida que cada usuário é obrigado a fechar uma conexão para cada um dos participantes e além disto enviar uma quantidade considerável de mídias de áudio e vídeo a todos estes canais.

Ao analisar o protocolo base que uma Conferência Web utiliza para seu funcionamento, começa a notar-se que em diversos casos e também no caso do BigBlueButton que é o alvo dos estudos deste trabalho, que estas aplicações necessitam de serviços de entrega confirmada para garantirem seu funcionamento. Esta confirmação de entrega não ocorre apenas para canais de controle e sim para todos os canais que a aplicação utiliza, uma vez que esta ferramenta fornece diversos recursos aos usuários no decorrer de uma conferência, isto será mostrado nos capítulos a seguir.

Levando em consideração que os serviços de Conferência Web são construídos e providos em uma rede com alicerce na implementação dos protocolos TCP/IP e tendo em mente que o objetivo deste trabalho é principalmente analisar e entender como funcionam os canais de comunicação e de tratativa de congestionamento na rede oriundo de uma Conferência Web, não se pode desprezar o conceito de camadas de rede de acordo com o modelo TCP. Este modelo é composto basicamente por 4 níveis. Segundo TANENBAUM (2002) o modelo TCP é formado pelas camadas de aplicação, transporte, inter-redes, host/redes.

Basicamente todas as camadas serão abordadas, desde a camada física partindo da configuração das máquinas para o ambiente de testes, até as análises da camada de aplicação, contudo o foco principal será nas camadas de rede e aplicação. O controle no congestionamento feito na camada de aplicação acaba por fornecer uma maior interação com o usuário uma vez que é possível definir parâmetros que podem ser alterados de acordo com a necessidade de cada usuário em seus cenários específicos.

A camada de redes implementa medidas mais específicas que serão observadas e também sugeridas como configuração de QOS(Quality of Service) para priorizar tráfegos específicos. A garantia da qualidade de serviços pode ser entendida como um requisito mínimo que as aplicações exigem para que consigam prover seus serviços de maneira satisfatória segundo PINHEIRO (2004), ainda complementa que a qualidade de serviços não é um parâmetro tão trivial de ser ajustado, uma vez que não depende de apenas um equipamento de rede e sim de equilibrar e calibrar as configurações de diversos equipamentos que compõem uma topologia.

Um outro conceito importante que será abordado é o controle de congestionamento

que *a priori* foi proposto por Van Jacobson em 1997 e tem suma importância pois receptor e emissor são fontes de potência variável o que por sua vez pode levar ao problema de esgotamento dos buffers de alguma das partes para com a outra, conforme FERREIRA (2008). Existem diversos algoritmos de controle de congestionamento que podem ser aplicados de acordo com os casos e sintomas que a rede possa apresentar. É possível citar alguns exemplos descritos por FERREIRA (2008), dentre estes destacam-se os algoritmos de controle que predominam nas tratativas do congestionamento, são eles o *Fast Retransmit*, *Fast Recovery*, *Slow Start* e *Congestion Avoidance* (realizam o ajuste da taxa de transmissão e também identificam perdas de segmentos).

Assim como os conceitos citados acima, mais conceitos surgirão no decorrer do desenvolvimento deste trabalho uma vez que a Conferência Web é uma área que envolve diversos fatores e apêndices integrados em sua composição e aplicabilidade.

2.3 Uso do BigBlueButton no Ensino

Conforme visto acima existem algumas entrelinhas que devem ser analisadas ao utilizar uma ferramenta de Conferência Web, pensando nesse ponto chega-se a conclusão de que, para exploração e tentativa de amenizar os problemas citados seria mais fácil a utilização de uma ferramenta de código livre, onde é possível ter acesso a todas as linhas de código para que assim seja possível compreender melhor toda a estrutura da aplicação.

O BigBlueButton é uma ferramenta que foi construída com a premissa de que todo estudante com um Navegador Web deve ter acesso a uma experiência de aprendizagem *online* de qualidade (BIGBLUEBUTTON INC,2015). É possível encontrar os repositórios da ferramenta disponíveis no GitHub (<https://github.com/bigbluebutton/bigbluebutton>) e uma documentação bem especificada de como utilizar a ferramenta no site do próprio BigBlueButton(<http://docs.bigbluebutton.org/>).

Esta ferramenta possui uma diversidade de funcionalidades que auxiliam por exemplo uma aula de EAD, nela é possível compartilhar não apenas áudio e vídeo entre todos os participantes como também exibir documentos compartilhados em tempo real e editá-los transmitindo as modificações a todos os participantes conectados.

A vantagem do código livre possibilita integrar à ferramenta melhorias que outros desenvolvedores disponibilizaram de acordo com suas experiências de usabilidade e também compartilhar suas próprias melhorias com os mesmos. Desta maneira é possível sempre dispor de novos recursos e ajustes para oferecer aos usuários finais da ferramenta.

Além das utilizações citadas nos parágrafos anteriores, o BigBlueButton é uma ferramenta que vem servindo de base para aplicações de Conferência Web que são utilizadas em diferentes áreas e locais, um exemplo desta utilização é a ferramenta Mconf que foi desenvolvida na Universidade Federal do Rio Grande do Sul, sua aplicabilidade vai desde ambientes acadêmicos até meios corporativos.

2.4 Trabalhos Correlatos

Existem diversos trabalhos que podem se correlacionar com este trabalho, pois, serão executadas pesquisas na área da Conferência Web, trabalhos na área do ensino a distância uma vez que o BigBlueButton tem o foco principal neste segmento, por fim também existem trabalhos aplicados às áreas de redes, uma vez que os fluxos da ferramenta e o congestionamento serão analisados neste trabalho.

Correlaciona-se citar a obra de MORAES (2002) Educação a distância: Fundamentos e práticas que se relaciona com o trabalho abordado uma vez que ela também faz referência aos conceitos do ensino a distância, sua importância para os tempos atuais, formas de emprego do EAD na formação do discente entre outros assuntos cruciais.

Um trabalho que também se contextualiza muito bem com os temas que serão abordados é o de BARCELAR (2012), Congestionamento em Redes TCP, que explica como ocorrem os congestionamentos em uma rede TCP., explica a importância da utilização de mecanismos de controle de congestionamento e também faz citação de alguns mecanismos para o controle do congestionamento.

A fim de entendimento do funcionamento e integração da Conferência Web com o mundo acadêmico, o trabalho de KICH (2015), Integração de Ferramenta de videoconferência ao Moodle, também foi estudado, uma vez que este trabalho engloba o BigBlueButton como uma das ferramentas para integração acadêmica ao Moodle (Modular Object Oriented Distance Learning).

3 Arquitetura do BigBlueButton

Conforme mostrado na introdução, apesar de o BigBlueButton poder ser empregado de diversas formas, ele foi projetado com design voltado ao aprendizado *online*. Com isso antes que fosse apresentada qualquer melhoria, foi necessário entender o funcionamento da ferramenta e quais os pontos mais importantes a serem analisados com a finalidade de afunilar o foco dos estudos.

Pensando nisto, foram estudados todos os módulos da plataforma e suas especificidades integradas. A arquitetura do BigBlueButton é composta por diversos módulos de *software* livre que trabalham em conjunto, dentre estes *softwares* destacam-se alguns como, tomcat7, redis, red5, FreeSWITCH, nginx entre outros.

A imagem abaixo (Figura 1) extraída da documentação do BigBlueButton mostra uma visão geral da arquitetura e como os componentes presentes se comunicam:

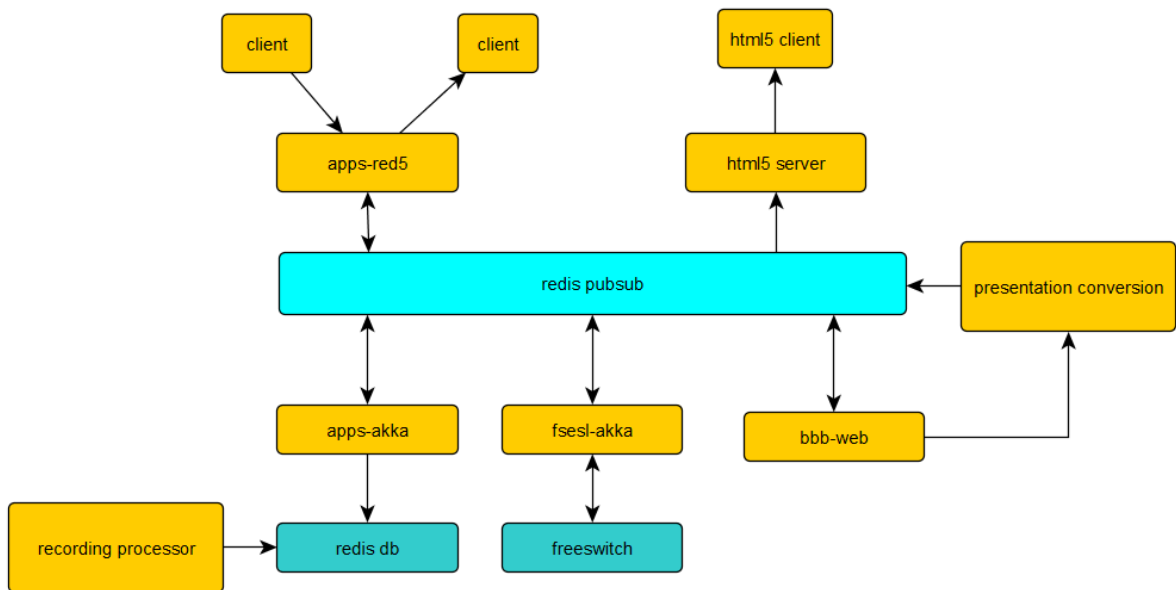


Figura 1 – Arquitetura do BigBlueButton

No BigBlueButton o cliente implementa uma aplicação em Flash que é suportada por um *Browsers*. A comunicação deste cliente com a plataforma é feita através dos protocolos RTMP(Real Time Monitoring Protocol) na porta 1935. Este protocolo foi desenvolvido pela Macromedia com a finalidade de fornecer uma transmissão de áudio, vídeo e dados com alta performance. A fim de contextualização os componentes do BigBlueButton serão descritos resumidamente, uma vez que os estudos foram voltados aos elementos que dizem respeito a melhoria da conexão e do congestionamento de uma Conferência Web.

3.1 Interface do BigBlueButton

O BigBlueButton possui um front-end proprietário que foi nomeado de GreenLight, este *front-end* por sua vez permite que os usuários acessem o sistema e utilizem suas principais funcionalidades como por exemplo, criar reuniões, ingressar em reuniões já existentes e convidar outros usuários para ingressar em uma reunião. Existem outras funcionalidades que também são providas pelo sistema como a gravação de conferências e manutenção das mesmas. Porém como o front-end tem o intuito de ser minimalista e simplificado para facilitar o uso, normalmente se destacam as funções principais que foram citadas (Figura 2).

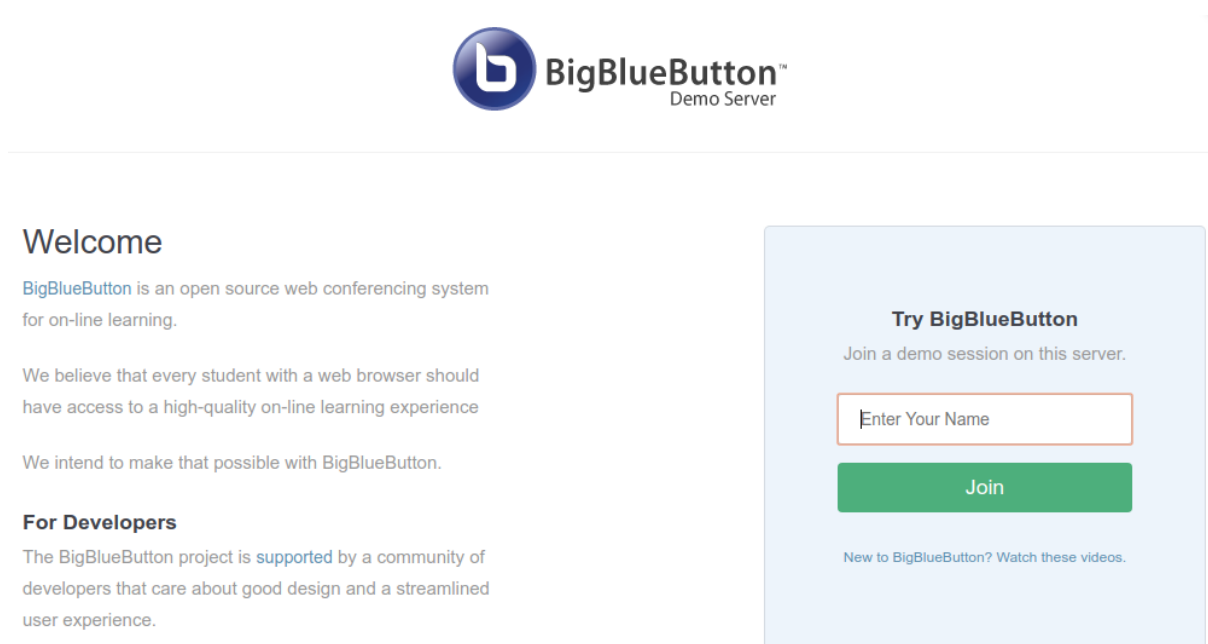


Figura 2 – Tela de Login do BigBlueButton

Outro módulo muito importante do BigBlue Button é a sua API(Application Programming Interface) que é responsável por realizar a integração e conexão de usuários de diferentes front-ends, pois, apesar de ser dotado do GreenLight, o BigBlueButton não é restrito apenas ao seu *front-end*, a Api Web por sua vez fornece abstração para que front-ends de terceiros possam ser utilizados, como alguns exemplos cita-se o Canvas, Wordpress, Sakai, Moodle, entre outros que podem ser encontrados na página do desenvolvedor.

3.2 Módulos de Processamento de Informação

É possível denominar de módulos de processamento de informação, pois, os componentes citados a seguir são os principais responsáveis pelo funcionamento essencial e coordenação das funcionalidades do BigBlueButton, pois são eles que trabalham os dados recebidos e fazem com que as Conferências Web ocorram. Sendo elas ferramentas da Redis que estão presentes no BigBlueButton, com isso tem-se:

- Redis PubSub - tem o papel de prover a comunicação entre algumas aplicações que rodam no BigBlueButton por meio do estabelecimento de canais, com ele possível criar canais de notificação (Pub) e assim selecionar quais os clientes irão receber tais notificações (Subs);
- Redis DB - é um banco de dados não relacional que armazena valores chave, nele são armazenados os eventos de uma chamada quando esta foi gravada;
- Aplicações do Red 5 - estas aplicações são as responsáveis por manterem os canais de mídia como por exemplo o compartilhamento de tela, canais de áudio e vídeo entre outros.

3.3 Servidor Red5

O Red 5 demanda uma atenção especial pois, ele possui uma gama de aplicações que são responsáveis por diversas funções essenciais dentro do BigBlueButton. Dentre essas funções se evidenciam principalmente o *Streaming* de mídia, *chat*, apresentações de slides, compartilhamento de tela, ingresso via áudio e vídeo entre outras funções (Figura 3).

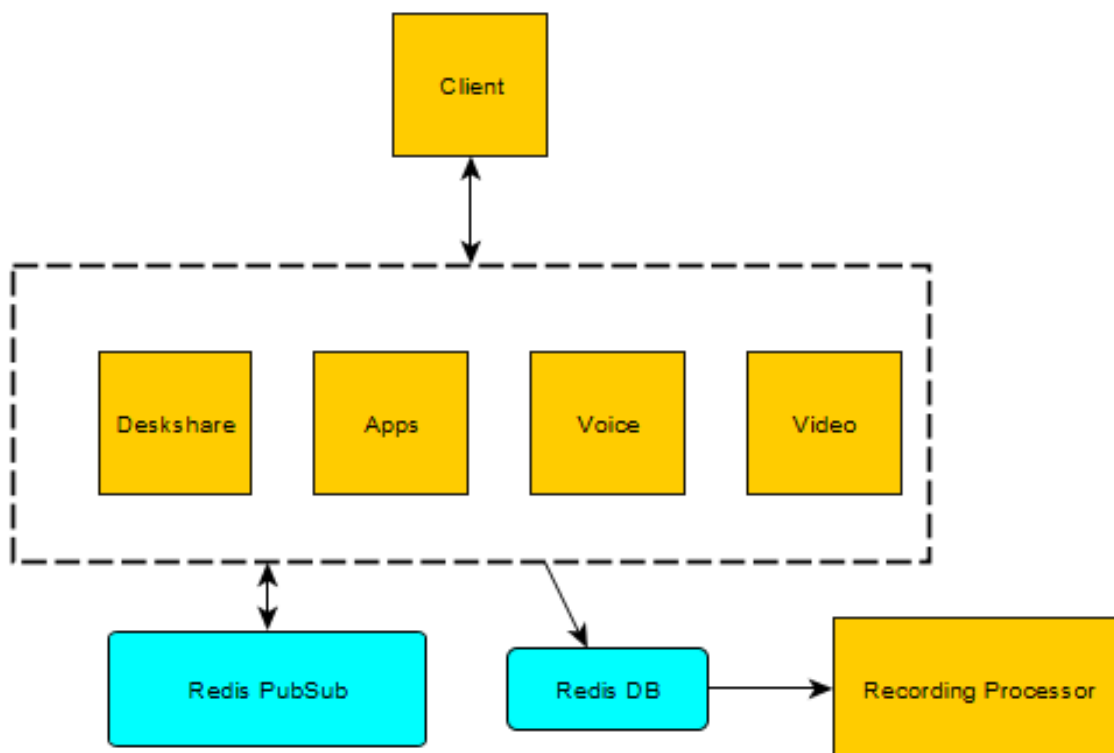


Figura 3 – Explosão da Arquitetura do Red5

Essas são as funções de destaque no BigBlueButton, pois elas compõem uma colaboração *online*, onde é possível os diversos usuários ingressarem e adaptarem os componentes da reunião de acordo com sua necessidade. Pensando nisto, se começa a notar pontos onde é possível estudar o controle de congestionamento e o funcionamento dos fluxos, de acordo com cada um dos canais de mídia existentes na plataforma, e assim propor níveis de prioridade de acordo com o tipo de uma conferência e com a necessidade de o usuário manter estes canais ativos e transmitindo.

Pensando a nível de voz, o módulo de destaque é o FreeSWITCH, ele fornece as funcionalidades do *chat* de voz, graças ao FreeSWITCH os usuários têm a possibilidade de desfrutarem de uma conferência de áudio tanto dos navegadores padrões (Google Chrome, Mozilla Firefox) quanto de provedores VOIP (Voice over Internet Protocol), onde é configurado e discado a partir de um telefone convencional compatível com tal funcionalidade.

As apresentações, passam por um módulo de conversão para que sejam exibidas durante a conferência, com isso, o resultado final é um arquivo no formato SWF que pode ser exibido em flash.

Para desenvolvimento deste projeto, foi necessário entender o funcionamento de uma Conferência Web a nível de rede. Em vista disso, se optou por analisar os canais de mídia presentes em uma conferência a fim de entender quais são os fluxos mais importantes e quais deles acabam por trazer mais impacto no nível de congestionamento de uma rede. Conforme analisado e demonstrado mais a frente, também descrito na documentação da plataforma, é possível ressaltar quais os canais estão presentes em uma conferência, dentre eles estão os canais de áudio, vídeo, *chat*, controle, apresentação.

3.4 Protocolo RTMP

O protocolo de troca de mensagens em tempo real também conhecido por RTMP é quem rege as trocas de dados de áudio e vídeo entre os participantes de uma Conferência Web no BigBlueButton, com base nisto é necessário entendermos seu funcionamento para que assim seja possível tirar algumas conclusões a respeito do congestionamento e o que é possível ser feito para ameniza-lo. Para analisar os canais deste protocolo foram utilizadas duas ferramentas de capturas de pacotes, o Wireshark e o rtmpdump. Ambas ferramentas têm a função de capturar todos os pacotes que fluem pelas interfaces de redes e após isso permite aplicar filtros para análise do tráfego capturado.

Várias análises serão feitas acerca deste protocolo a fim de compreender toda a alocação de portas, fluxo de informações e como se portam as conexões na medida que vários participantes entrem em uma sala de webconferência.

Na imagem abaixo é possível verificamos como funciona o *handshake* ("aperto de mão", termo usado para o estabelecimento de comunicação) no protocolo RTMP e TCP:

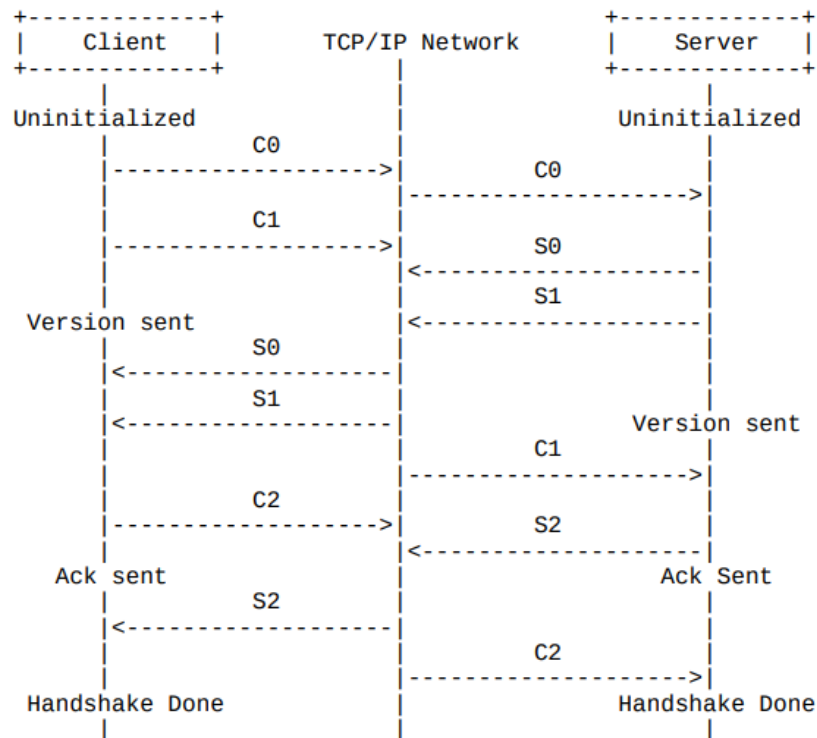


Figura 4 – Handshake do Protocolo RTMP

Segundo PARMAR e THORNBURGH (2012), que redigiram a documentação oficial da Adobe a respeito deste protocolo, a conexão se inicia com o cliente enviando os pacotes C0 e C1 ao servidor e precisam aguardar até que o servidor devolva os pacotes S0 e S1 em resposta para que assim seja permitido que o cliente envie o próximo pacote. Após estas etapas concluídas o cliente envia um novo pacote C2 ao servidor, neste pacote além das informações básicas de cabeçalho, também está presente o tipo de conexão que este canal deverá fechar, por exemplo um canal de vídeo ou de áudio. O fluxo de conexão RTMP gerado por uma comunicação do BigBlueButton será exibido nas sessões seguintes ao decorrer da explicação dos testes e medições realizados.

4 Análise de Tráfego no BigBlueButton

Por ser uma ferramenta que faz uso do código livre, o BigBlueButton pode ser instalado e testado sem muitas barreiras e dificuldades, conforme será observado nas sessões seguintes, foi preparado um ambiente de testes controlado, a fim de amostrar e decifrar o funcionamento da ferramenta estudada, e assim dar andamento nos objetivos propostos.

4.1 Instalação e Configuração do Servidor BigBlueButton

4.1.1 Repositório do BigBlueButton

Em vista dos conceitos apresentados acima, foi preparado o ambiente para testes na plataforma do BigBlueButton. O próprio desenvolvedor fornece o código para fácil instalação do servidor de demonstração para fins de pesquisa e desenvolvimento. O mesmo pode ser encontrado no seguinte endereço:

- **GitHub** - <https://github.com/bigbluebutton/bbb-install>
- **Guia de instalação** - <http://docs.bigbluebutton.org/install/install.html>

A versão da aplicação utilizada foi o BigBlueButton 2.0, este é o último release disponibilizado pelo desenvolvedor e também o mais completo. Como recomendação para utilização do serviço e melhoria de segurança o desenvolvedor sugere a utilização de um certificado SSL (Secure Sockets Layer) que basicamente tem a função de estabelecer uma conexão entre cliente e servidor que seja codificada, assim trazendo um maior nível de segurança no acesso ao servidor. Como o ambiente de testes utilizado foi montado de maneira controlada, não se fez necessário a utilização de um certificado SSL para os serviços testados, isto também devido à disposição do ambiente estar montada em uma rede LAN (Local Area Network).

4.1.2 Configuração do Ambiente para o BigBlueButton

Para montar o ambiente de testes e análises, foi utilizada a ferramenta Virtual Box para criar três máquinas virtuais, sendo uma delas o servidor que hospeda o serviço do BigBlueButton e as outras duas máquinas são clientes para realizar os testes de conexão.

O servidor precisou ter uma configuração específica conforme requisitos mínimos exigidos pelo desenvolvedor da solução, segue a mesma:

- Sistema Operacional Ubuntu 16.04 (Xenial Xerus);

- 4GB de memória RAM;
- 2 núcleos de processamento;
- 10 GB de disco rígido;
- IP LAN: 192.168.56.2.

Já os clientes foram criados com a seguinte configuração:

- Sistema Operacional Ubuntu 16.04 (Xenial Xerus);
- 1 GB de memória RAM;
- 1 núcleo de processamento;
- 10 GB de disco rígido;
- IP LAN: 192.168.56.3.

Foi necessário criar uma rede LAN (vboxnet0) dentro do ambiente de virtualização, a fim de estabelecer uma comunicação entre as duas máquinas. Com isso foi atribuído um IP fixo de LAN a cada uma delas conforme descrito nas especificações acima. Também foi possível integrar o host para realizar testes em conjunto as máquinas virtuais citadas. A seguir estão descritas as configurações da rede criada:

- **Gateway** - 192.168.56.1
- **Máscara** - 255.255.255.0

4.1.3 Teste de Conexão e Abertura de Canais

A primeira etapa de testes consistiu em identificar todos os fluxos e canais que são abertos quando um cliente ingressa em uma Conferência Web pela primeira vez. Este teste teve o intuito de identificar quantos canais são abertos com o servidor e, como o mesmo gestiona estes canais, isso é necessário para que seja possível identificar os principais canais que possam causar maior congestionamento no tráfego da informação na medida que a quantidade de clientes cresce.

Como o protocolo RTMP tem por padrão a utilização da porta 1935, foi feita uma análise inicial em cima das conexões nesta porta, assim observa-se na imagem abaixo que foi capturada no terminal do servidor após a configuração e inicialização do serviço de conferências do BigBlueButton, que a mesma está no estado de LISTEN (escutando em uma porta), com isso ela está no aguardo de novas conexões que possam chegar (Figura 5) nesta porta e ser estabelecidas.

```
root@bbb-VirtualBox:~# netstat -putan | grep 1935
tcp6      0      0  :::1935          :::*              OUÇA      1355/java
```

Figura 5 – Servidor Aguardando Conexões na Porta 1935

O comando utilizado para analisar as portas foi o **netstat -putan | grep 1935**, a primeira parte do comando antes do pipe, exibe as portas e conexões do servidor já segunda parte filtra as portas de acordo com o número da porta especificada.

Após constatar que o servidor está aguardando conexões na porta 1935, foi iniciado o cliente e conectado através do *Browser* Firefox no IP do servidor (192.168.56.2). Uma vez que o BigBlueButton é uma aplicação onde o cliente não necessita de uma instalação de *software* específica para a utilização do serviço, torna-se fácil ingressar com um novo participante. Ao solicitar o ingresso em uma nova conferência, foi possível observar os canais se conectando na porta 1935 do servidor (Figura 6).

```
root@bbb-VirtualBox:~# netstat -putan | grep 1935
tcp6      0      0  :::1935          :::*              OUÇA      1355/java
tcp6      0      0  192.168.56.2:1935  192.168.56.3:34152 ESTABELECID 1355/java
tcp6      0      0  192.168.56.2:1935  192.168.56.3:34154 ESTABELECID 1355/java
tcp6      0      0  192.168.56.2:1935  192.168.56.3:34156 ESTABELECID 1355/java
tcp6      0      0  192.168.56.2:1935  192.168.56.3:34174 ESTABELECID 1355/java
```

Figura 6 – Conexões na Porta 1935 do Servidor

Também é possível evidenciar que a porta ainda está escutando por novas conexões para caso novos cliente ingressem na conferência.

Como as conexões ocorrem de maneira muito rápida, apenas analisar a conexão nas portas não permitiu a visualização de como o fluxo foi criado neste exemplo, com isso foi utilizada a ferramenta Wireshark para realizar uma captura de todos os pacotes que fluíam pelas interfaces de rede, assim foi possível identificar como os canais foram estabelecidos e identificar também qual informação cada um dos fluxos transmite.

A captura dos dados no wireshark foi realizada através do comando **tshark -i any -w /tmp/captura-conection.pcap**, este comando captura os pacotes em todas as interfaces de rede (parâmetro -i any) e armazena no arquivo captura-conection.pcap dentro do diretório tmp (parâmetro -w /tmp/captura-conection.pcap).

A coleta de pacotes foi realizada por aproximadamente 20 segundos, este é o tempo médio que um cliente gasta para abrir a página do servidor em seu *Browser* e entrar com suas informações para ingressar em uma conferência. Com isso foi possível levantar alguns pontos para análise posterior na ferramenta. O primeiro ponto analisado na captura do wireshark foi o estabelecimento dos canais de comunicações para confirmar que as conexões ocorreram da mesma maneira que na análise da porta 1935 realizada previamente e explicitada no início deste tópico. A surpresa é que na captura foram criados 5 canais

de comunicação e não apenas 4 conforme mostrado na análise da porta. Isso mostra que apenas listar as conexões estabelecidas nas portas do servidor nem sempre evidência todos os canais estabelecidos para a iniciar uma comunicação de Conferência Web no caso do BigBlueButton. Este fato está evidenciado na imagem abaixo (Figura 7).

No.	Time	Source	Destination	Protocol	Length	Info
167	2018-10-18 19:26:34,1226342...	192.168.56.1	192.168.56.2	RTMP	1605	Handshake C0+C1
169	2018-10-18 19:26:34,1243297...	192.168.56.2	192.168.56.1	RTMP	3141	Handshake S0+S1+S2
171	2018-10-18 19:26:34,1247153...	192.168.56.1	192.168.56.2	RTMP	2127	Handshake C2 connect('video/portTest')
203	2018-10-18 19:26:34,4092940...	192.168.56.1	192.168.56.2	RTMP	1605	Handshake C0+C1
205	2018-10-18 19:26:34,4115484...	192.168.56.2	192.168.56.1	RTMP	3141	Handshake S0+S1+S2
207	2018-10-18 19:26:34,4141447...	192.168.56.1	192.168.56.2	RTMP	2275	Handshake C2 connect('bigbluebutton/183f0b...
426	2018-10-18 19:26:36,7634424...	192.168.56.1	192.168.56.2	RTMP	1605	Handshake C0+C1
428	2018-10-18 19:26:36,7652222...	192.168.56.2	192.168.56.1	RTMP	3141	Handshake S0+S1+S2
430	2018-10-18 19:26:36,7655998...	192.168.56.1	192.168.56.2	RTMP	2115	Handshake C2 connect('screenshare/183f0bf3...
581	2018-10-18 19:26:39,1906827...	192.168.56.1	192.168.56.2	RTMP	1605	Handshake C0+C1
583	2018-10-18 19:26:39,1923440...	192.168.56.2	192.168.56.1	RTMP	3141	Handshake S0+S1+S2
585	2018-10-18 19:26:39,1926804...	192.168.56.1	192.168.56.2	RTMP	2210	Handshake C2 connect('video/183f0bf3a0982a...
3414	2018-10-18 19:27:02,1690363...	192.168.56.1	192.168.56.2	RTMP	1605	Handshake C0+C1
3416	2018-10-18 19:27:02,1745268...	192.168.56.2	192.168.56.1	RTMP	3141	Handshake S0+S1+S2
3418	2018-10-18 19:27:02,1766488...	192.168.56.1	192.168.56.2	RTMP	2119	Handshake C2 connect('sip')

Figura 7 – Canais de Comunicação Capturados pelo Wireshark

4.1.4 Canal de Teste de Conexão

Analisando o primeiro canal criado, foi identificado que o cliente ao ingressar em uma conferência, realiza primeiramente um teste na conexão abrindo um fluxo de vídeo mesmo não transmitindo nenhuma imagem, após este teste o canal é encerrado automaticamente e aberto o próximo canal que é responsável pelo fluxo de controle.

110	String 'portTestMeetingId-1539890794944'	110	String 'Alex'
		111	AMF0 type: String (0x02)
		112	String length: 4
		113	String: Alex
		114	String 'MODERATOR'
		115	AMF0 type: String (0x02)
		116	String length: 9
		117	String: MODERATOR
		118	String '183f0bf3a0982a127bdb8161e0c44eb696b3e75c-1539890704941'
		119	AMF0 type: String (0x02)
		120	String length: 54
		121	String: 183f0bf3a0982a127bdb8161e0c44eb696b3e75c-1539890704941
		122	String '72308'

Figura 8 – Comparação do Payload do Pacote de Testes e do Pacote de Controle

Na Figura 9 ao colocar os *payloads* do pacote de controle e do pacote de teste lado a lado em um verificador de diferenças de texto, é possível analisar que as informações são bastante divergentes e que o pacote de teste realmente não trás nenhuma informação de grande relevância para a efetivação dos demais fluxos de dados. Este canal serve apenas

para que a conexão seja testada sem a necessidade de iniciar os demais canais de uma só vez.

Apesar de não trazer dados contendo informações primordiais como os demais pacotes de mídia contendo imagens, áudio entre outros presentes na Conferência Web, na criação deste canal é possível identificar uma maneira que o desenvolvedor implementou para aplicar uma pequena melhoria no congestionamento da rede, uma vez que caso a conexão de teste não seja bem-sucedida, os demais canais não chegam nem a ser criados, evitando o envio de pacotes desnecessários na rede e assim preservando um pouco mais a integridade dos serviços já ativos no momento.

4.1.5 Canal de Controle

O canal subsequente a ser criado é o responsável por realizar a gestão da Conferência Web em si. Nele são transmitidas informações cruciais a respeito dos participantes, controle de ações realizadas no *Browser* dos clientes, além de todo o controle de gestão de botões. No *payload* do canal de controle, são identificados logo nos primeiros segundos de transmissão o envio de informações para identificação do participante, como nome, identificação de moderador, estado de conexão entre outras informações chave para o ingresso e caracterização do usuário na conferência.

```
▶ Real Time Messaging Protocol (Handshake C2)
▼ Real Time Messaging Protocol (AMF0 Command connect('bigbluebutton/183f0bf3a0982a127bdb8161e0c44eb696b3e75c-1539890704941')
  Response to this call in frame: 214
  ▶ RTMP Header
  ▼ RTMP Body
    ▶ String 'connect'
    ▶ Number 1
    ▶ Object (11 items)
    ▶ String 'Alex'
    ▶ String 'MODERATOR'
    ▶ String '183f0bf3a0982a127bdb8161e0c44eb696b3e75c-1539890704941'
    ▶ String '72308'
    ▶ Boolean false
    ▶ String 'w_ifawiduawyq1'
    ▶ String 'w_ifawiduawyq1'
    ▶ Boolean false
    ▶ Boolean false
    ▶ String 'esnctbbixizl'
    ▶ String 'DXG57XNKOE7ITIHN'
```

Figura 9 – Payload do Pacote de Controle

O canal de controle também tem a função de distribuir as mensagens que são enviadas no *chat* (bate-papo de mensagens de texto), e ainda envia as coordenadas do ponteiro que o apresentador da conferência utiliza assim exibindo em todas as telas a posição exata de onde o mouse do apresentador está, além de transmitir as alterações realizadas na apresentação.

Vale salientar que o por mais que o canal de controle também seja responsável pelo compartilhamento da apresentação de slides, na ferramenta BigBlueButton existe uma diferença entre o compartilhamento de tela e o compartilhamento de slides. Com isso, as informações compartilhadas no canal de controle são referentes apenas aos slides que são carregados em PDF e formatos de arquivos de texto entre outros, caso o apresentador queira exibir um vídeo de conteúdo e mídia de áudio é necessário que o mesmo habilite o compartilhamento de tela e transmita essas informações no canal específico para tais funções.

4.1.6 Canal de Compartilhamento de Vídeo

Ao analisar cada um dos canais em particular, é possível notar que ambos os canais possuem informações que vinculam os pacotes transmitidos com as suas respectivas conferências, observando o *payload* do pacote de vídeo é possível constatar dentro do *body* (corpo do pacote) que existem alguns campos chave que identificam o tipo do pacote e qual a conferência o pacote transmitido pertence, tais campos se evidenciam na figura 10.

```
▶ Real Time Messaging Protocol (Handshake C2)
▼ Real Time Messaging Protocol (AMF0 Command connect('video/183f0bf3a0982a127bdb8161e0c44eb696b3e75c-1555343343270'))
  Response to this call in frame: 3438
  ▶ RTMP Header
  ▼ RTMP Body
    ▼ String 'connect'
      AMF0 type: String (0x02)
      String length: 7
      String: connect
    ▼ Number 1
      AMF0 type: Number (0x00)
      Number: 1
    ▼ Object (11 items)
      AMF0 type: Object (0x03)
      ▶ Property 'app' String 'video/183f0bf3a0982a127bdb8161e0c44eb696b3e75c-1555343343270'
      ▶ Property 'flashVer' String 'LNX 31,0,0,122'
      ▶ Property 'swfUrl' String 'http://192.168.56.2/client/VideoconfModule.swf?v=934'
      ▶ Property 'tcUrl' String 'rtmp://192.168.56.2/video/183f0bf3a0982a127bdb8161e0c44eb696b3e75c-1555343343270'
      ▶ Property 'fpad' Boolean false
      ▶ Property 'capabilities' Number 239
      ▶ Property 'audioCodecs' Number 3575
      ▶ Property 'videoCodecs' Number 252
      ▶ Property 'videoFunction' Number 1
      ▶ Property 'pageUrl' String 'http://192.168.56.2/client/BigBlueButton.html?sessionToken=kpd0c44eb696b3e75c-1555343343270'
      ▶ Property 'objectEncoding' Number 3
      End Of Object Marker
    ▶ String '183f0bf3a0982a127bdb8161e0c44eb696b3e75c-1555343343270'
    ▶ String 'w_kir7uq5ng6ut'
    ▶ String 'ekoyrvwpdp4'
    ▶ String 'LBTTUVUZEGBEJJ7A'
```

Figura 10 – Payload do Pacote de Vídeo

Nota-se que o canal de compartilhamento de vídeo é estabelecido mesmo que o usuário não esteja utilizando a transmissão de sua imagem através de uma câmera, baseado nesse fato, é possível levantar alguns pontos a respeito das características de como se dá uma conferência, que ocorre por exemplo no meio estudantil, onde não é necessário obrigatoriamente que todos os alunos estejam com suas câmeras ligadas. Normalmente o intuito da mensagem a ser transmitida pelo docente a seus discentes em uma conferência é a explicação por meio do áudio e de slides que estão sendo apresentados no *DashBoard* (painel de apresentações), se pode eliminar a necessidade de ter todos os alunos e até mesmo o próprio docente com seu canal de áudio transmitindo imagem. Com isso já se começa a perceber que alguns pontos de melhoria podem ser empregados de acordo com a natureza de uma Conferência Web.

Ainda nesta linha de raciocínio, é possível termos uma quantidade muito maior de discentes escutando a explicação do docente e visualizando uma apresentação sem a necessidade de todos exibirem suas imagens, em comparação a um cenário onde todos estariam com seus canais de vídeo transmitindo a todo tempo. Com isso uma faixa de banda originalmente congestionada por uma quantidade x de canais desnecessários, pode dar espaço a uma quantidade menor de canais desnecessários porém conectando mais participantes legítimos, assim tornando o uso da aplicação mais eficiente e exigindo menos recursos de rede.

4.1.7 Canal de Compartilhamento de Áudio

Como visto na sessão anterior, os *payloads* de cada um dos canais iniciados é bastante semelhante, com a alteração apenas no identificador de cada um dos campos, com base nisso, não é necessário demonstrar o *payload* do canal de áudio dentro do RTMP, porém este canal apresenta uma particularidade muito importante para o BigBlueButton que é o protocolo SIP (Session Initiation Protocol) que pode ser visto na imagem 11.

O fato de o BigBlueButton utilizar este protocolo para a comunicação de áudio é o que possibilita que um participante ingresse em uma Conferência Web através de um telefone convencional, assim é possível que este participante receba pelo menos o áudio da conferência, conforme discutido na sessão anterior de acordo com o cunho de uma Conferência Web, apenas o canal de áudio já é suficiente para que se transmita a informação desejada.

4.1.8 Canal de Compartilhamento de Tela

Este canal funciona exclusivamente para que um usuário possa compartilhar sua tela com os demais participantes, é possível realizar o compartilhamento de duas maneiras, uma delas é compartilhar a tela inteira e a outra é compartilhar apenas uma área específica.

```
▶ User Datagram Protocol, Src Port: 5070, Dst Port: 5060
▼ Session Initiation Protocol (INVITE)
  ▶ Request-Line: INVITE sip:79181@10.0.2.15 SIP/2.0
  ▶ Message Header
  ▼ Message Body
    ▼ Session Description Protocol
      Session Description Protocol Version (v): 0
      ▶ Owner/Creator, Session Id (o): GLOBAL_AUDIO_79181_0_0 IN IP4 10.0.2.15
      Session Name (s): Session SIP/SDP
      ▶ Connection Information (c): IN IP4 10.0.2.15
      ▶ Time Description, active time (t): 0 0
      ▶ Media Description, name and address (m): audio 15000 RTP/AVP 100 8 18 0 111
      ▶ Media Attribute (a): rtpmap:100 SPEEX/16000/1
      ▶ Media Attribute (a): rtpmap:8 PCMA/8000/1
      ▶ Media Attribute (a): rtpmap:18 G729/8000/1
      ▶ Media Attribute (a): fmp:18 annex=no
      ▶ Media Attribute (a): rtpmap:0 PCMU/8000/1
      ▶ Media Attribute (a): rtpmap:111 ILBC/8000/1
      ▶ Media Attribute (a): fmp:111 mode=30
      ▶ Media Attribute (a):ptime:20
```

Figura 11 – Payload do Pacote de Áudio SIP

Para que o participante consiga compartilhar sua tela, é necessário que o mesmo execute um pequeno arquivo que é baixado assim que ele clica na opção de compartilhamento.

4.2 Cenários para Análise de Tráfego no BigBlueButton

4.2.1 Análise do Tráfego de Vídeo

Para testar o consumo de banda que um participante apresenta durante uma conferência, foi necessário realizar alguns testes de carga em cada um dos canais estabelecidos. Para capturar as métricas e criar os gráficos de exposição, foi utilizada a própria ferramenta Wireshark, ela possui um módulo chamado I/O Graph que permite criar gráficos de barra e também de linhas em tempo real com base nos valores extraídos. Esses valores podem ser obtidos por meio de filtros totalmente customizáveis de acordo com o contexto dos pacotes que estão sendo analisados.

Em cada um dos canais do BigBlueButton, foi necessário utilizar um filtro customizado de maneira diferente, por mais que o protocolo base das trocas de mídia seja o RTMP (Real Time Messaging Protocol), cada um dos canais possuem suas especificidades de parâmetros e seus números de portas exclusivos para uma dada conferência.

O primeiro canal testado foi o canal de vídeo, para plotar o gráfico que pode ser visto na figura 12, foi necessário utilizar o filtro `tcp.srport == 59846`, o comando captura todos os pacotes que estão fluindo pela porta 59846, como esta porta foi utilizada exclusivamente para os pacotes de vídeo entre o cliente e o servidor, é possível evicenciar todo o fluxo e suas variações.

Ainda a cerca da figura 12, é necessário analisar os comparativos entre as diferentes qualidades de transmissão de vídeo que a ferramenta disponibiliza ao usuário. Partimos

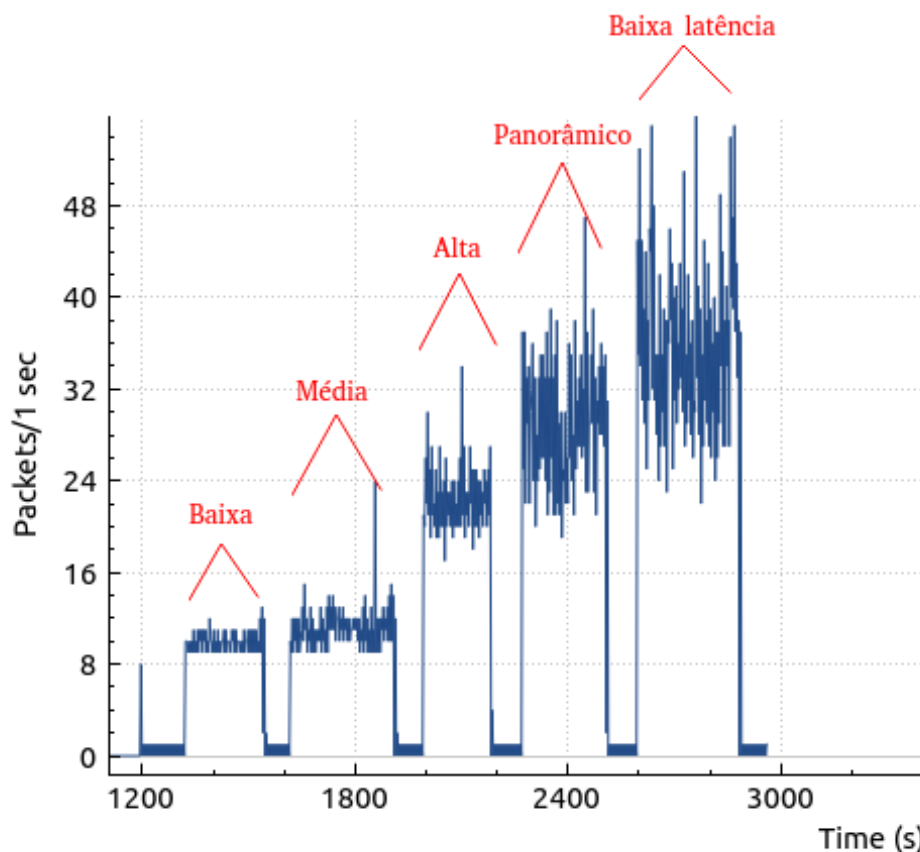


Figura 12 – Tráfego no Canal de Vídeo

do fluxo de menor consumo de banda que é o modo de baixa qualidade, nele a imagem capturada pela câmera é cortada em um *frame* (quadro de imagem) central onde o objetivo é expor apenas o rosto do participante, este modo apresenta problema como atrasos no vídeo e quadriculados na imagem.

O modo de média qualidade também realiza o mesmo corte no *frame* aproximando ao centro, porém, ele transmite uma pequena quantidade a mais de informação a respeito de cada um dos quadros, assim evitando um pouco que a imagem final do usuário fique quadriculada e também melhorando de maneira quase imperceptível a latência na transmissão da imagem.

Já no modo de alta qualidade, a imagem que é transmitida pelo usuário possui uma proporção bem maior que nos demais modos citados, o tamanho da tela também passa a ser de 1:1 sem nenhum corte aproximado ao centro. Também é notável a melhora no atraso dos quadros de vídeo. O modo panorâmico apresenta uma qualidade semelhante ao modo de alta qualidade, porém a diferença entre eles é que o modo panorâmico transmite uma imagem em proporção 16:9, assim abrangendo uma maior área de imagem, normalmente esse modo é utilizado quando existe alguma sala física com a necessidade de exibição de várias pessoas no mesmo local conectadas a uma conferência.

O último modo de vídeo fornecido é o de baixa latência, neste modo o consumo

de banda é muito maior que os demais, como o próprio nome já diz, este modo envia uma quantidade bem maior de quadros de vídeo a fim de minimizar a latência no vídeo transmitido. A qualidade da imagem é semelhante aos dois modos anteriores porém ela também se mantém em uma proporção 1:1 para o tamanho da imagem.

Além da análise por pacotes, também foi feita a análise por taxa de kilobytes por segundo, nos valores foram respectivamente 8 Kbps para a baixa qualidade, 12 Kbps para a qualidade média, 48 Kbps para a qualidade alta, 64 Kbps para a qualidade panorama e 160 kbps para a baixa latência de imagem. Nota-se que no modo de baixa latência a quantidade de banda consumida é muito mais agravada do que nos demais casos, isso pois uma quantidade maior de informação de mídia é transmitida. Este modo normalmente é utilizado é necessário apresentar conteúdos mais interativos através da câmera, como por exemplo um experimento em uma aula de química, onde as reações acontecem em frações de segundos e precisam ser observadas pelos alunos que estão na Conferência Web.

4.2.2 Análise do Tráfego de Controle

Como citado nas sessões anteriores, o canal de controle faz uma agregação de diversas funcionalidades como *chat*, apresentações de slide entre outros. Com isso o fluxo que este canal apresenta não possui uma constância como no caso do canal de vídeo.

Na imagem 13 é possível evidenciar que o canal de controle possui variações de curva diferenciadas a cada momento, este fato ocorre devido tipo ao serviço que está sendo utilizado no momento da transmissão grafada, em casos de transmissão de mensagens de texto através do *chat* na interface, a transmissão de pacotes é muito baixa, quase imperceptível em relação aos *pings* do canal, vale salientar que todos os canais possuem constantes transmissões de pacotes ICMP (Internet Control Message Protocol) contendo *pings* que servem como *heartbeats* (nomeclatura destinada a ação de realizar *pings* em um servidor para saber se o mesmo ainda está ativo e respondendo às requisições) entre os membros da conexão.

Ainda na imagem 13, pode-se verificar os pontos de maior envio de pacotes, estes pontos foram traçados no momento em que ocorreu movimentação no ponteiro laser da apresentação, os pacotes contêm as coordenadas em eixo X e Y em cada momento do ponteiro, isso permite que a movimentação realizada pela apresentação seja propagada a todos os membros que estiverem presentes na conferência de forma quase instantânea. Dentre todos os outros canais analisados, o canal de controle é um dos que apresenta o menor consumo de banca, porém este fato não torna este canal menos importante, uma vez que, todas as funcionalidades disponíveis na interface do usuário são propagadas ao servidor e aos demais participantes da conferência através deste canal.

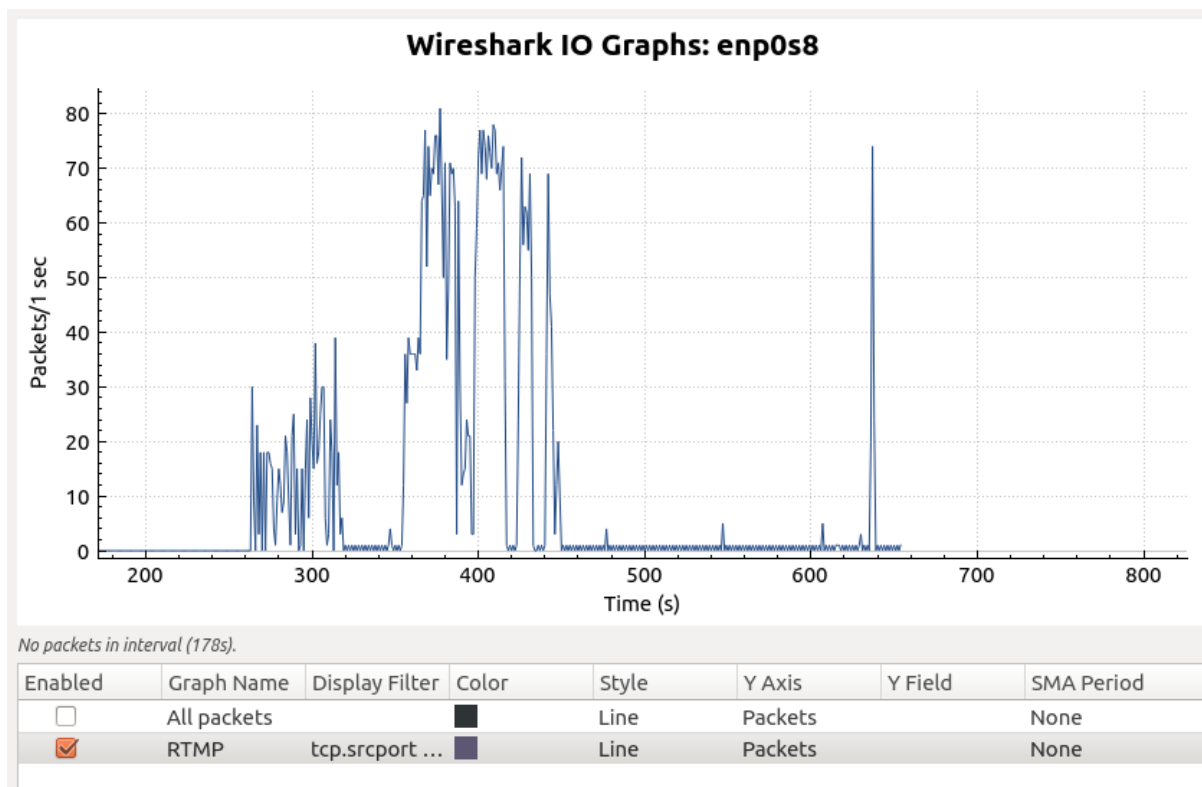


Figura 13 – Tráfego no Canal de Controle

4.2.3 Análise do Tráfego de Áudio

Para a análise do canal de áudio foi utilizado o mesmo conceito que nos demais canais, uma fonte emissora de áudio foi colocada em um dos clientes e o fluxo de dados gerado pela fonte emissora foi capturado com a utilização do Wireshark no servidor, foi possível analisar que, quando um som é transmitido, independente do volume do áudio, a quantidade em Kilobytes de dados no canal fica limitada a uma banda que oscila entre 48 Kb/s e 64 Kb/s sendo este o limite máximo transmitido neste canal. A taxa de transferência fica reduzida a um percentual abaixo destes valores apenas quando o som emitido é interrompido. É possível notarmos estas variações na figura 14 conforme abaixo.

Um outro ponto importante que foi avaliado durante os testes neste canal é o fato de que, como a comunicação se dá através do protocolo SIP, o canal de áudio não funciona como os demais canais que ao estabelecerem uma conexão, imediatamente é iniciado o envio de *pings* a fim de manter a mesma conexão aberta durante todo o período em que o usuário estiver conectado a uma Conferência Web. O canal de áudio estabelece um circuito fim a fim conforme uma chamada VOIP convencional, ao desligar o microfone imediatamente o canal é encerrado, no caso por exemplo do canal de vídeo, ao desligar a câmera, o canal continua ativo na mesma porta do servidor, assim caso o usuário ative novamente a câmera a conexão estará na mesma porta, por isso foi possível traçar o gráfico de diferentes modos de câmera no mesmo gráfico conforme figura 14.

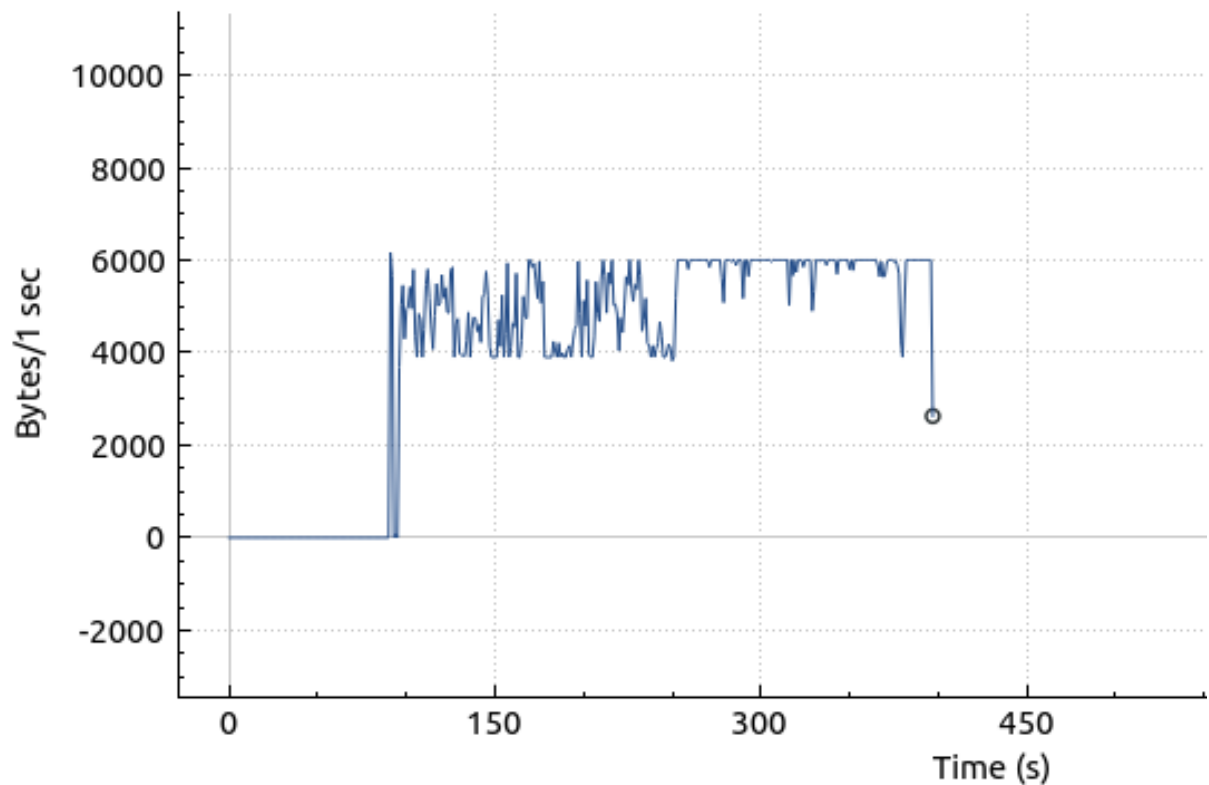


Figura 14 – Tráfego no Canal de Áudio

A figura 15 mostra os pacotes SIP que foram transmitidos nos testes realizado, é possível notar que as chamadas SIP receberam *status* de Not Accpetabe Here devido ao fato de os *codecs* SIP não terem sido configurados, como os testes envolveram apenas usuários dotados de *Browsers* a comunicação foi toda encima do protocolo RTMP.

Protocol	Length	Info
SIP/SDP	2404	Request: INVITE sip:919673425@192.168.56.2
SIP	465	Status: 100 Trying
SIP	859	Status: 488 Not Acceptable Here
SIP	389	Request: ACK sip:919673425@192.168.56.2
SIP/SDP	2402	Request: INVITE sip:919673425@192.168.56.2
SIP	465	Status: 100 Trying
SIP	859	Status: 488 Not Acceptable Here
SIP	389	Request: ACK sip:919673425@192.168.56.2
SIP/SDP	2403	Request: INVITE sip:919673425@192.168.56.2
SIP	464	Status: 100 Trying
SIP	858	Status: 488 Not Acceptable Here
SIP	388	Request: ACK sip:919673425@192.168.56.2

Figura 15 – Pacotes SIP Capturados Durante Testes Realizados

4.2.4 Análise do Tráfego de Compartilhamento de Tela

Conforme já citado nas sessões anteriores no BigBlueButton o compartilhamento de tela e o compartilhamento de slides são distintos, ambos ocupam canais específicos e estabelecem conexões em portas diferentes, assim não causando influência um sob o outro. Com isso é necessário analisar cada um destes canais separadamente.

O canal de compartilhamento de tela estabelece uma conexão apenas quando o usuário clica no botão de compartilhamento na interface, para que a tela seja exibida aos demais participantes o apresentador precisa realizar o *download* de um pequeno agente em JAVA e executar em sua máquina. Após o agente ser iniciado corretamente a tela começa a ser compartilhada imediatamente.

O tráfego de dados neste canal não possui nenhuma característica de constância específica, pois, a transmissão de dados varia de acordo com o que está sendo exibido na tela do usuário que está realizando o compartilhamento, como exemplo é possível analisar um momento onde o apresentador está exibindo apenas slides ou imagens estáticas e outro momento onde o mesmo está reproduzindo um vídeo, nestes dois casos o tráfego é totalmente diferente dentro do canal de compartilhamento de tela, no primeiro exemplo teremos uma taxa de banda consumida bem menos do que no segundo caso. Com isso pode-se concluir que este canal quando ativo, pode ao mesmo tempo apresentar um consumo de banda leve e também alto, isso de acordo com o conteúdo exibido.

Na figura 16, observa-se as variações na transmissão de dados dentro do canal de compartilhamento de tela. Claramente os exemplos citados podem ser demonstrados e mensurados, onde no momento onde a transmissão está mais baixa é onde a tela do usuário estava exibindo apenas imagens estáticas, e no momento de transmissão mais alta estava sendo exibido uma mídia de vídeo.

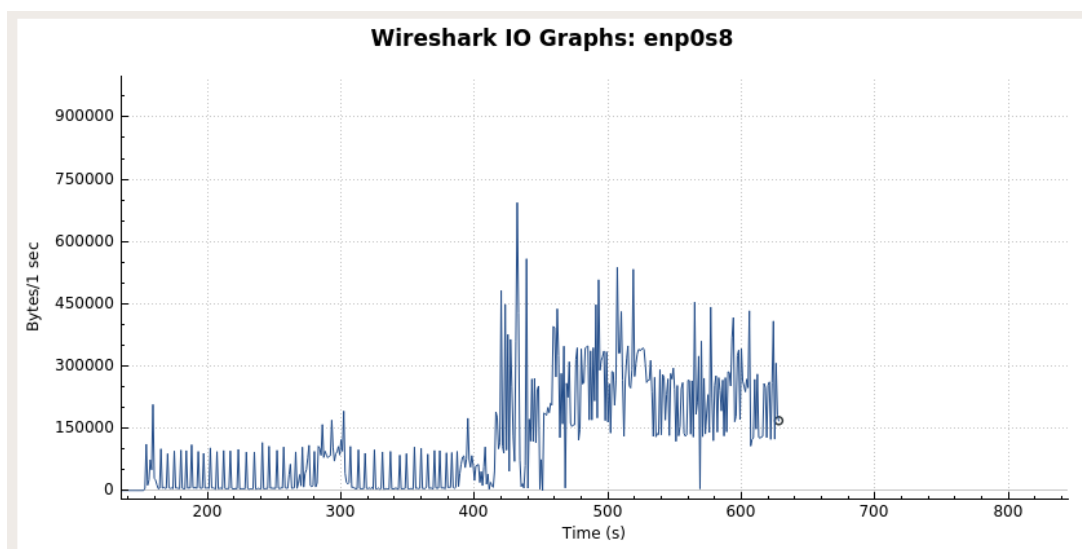


Figura 16 – Tráfego de Compartilhamento de Tela

Pensando no conceito de congestionamento de rede é possível fazer algumas considerações a respeito do canal de compartilhamento de tela, este é um canal que pode apresentar um alto crescimento no consumo de banda em um curto espaço de tempo, com isso é possível que a qualidade dos serviços seja degradada de maneira muito mais rápida do que em um ambiente onde não existe nenhum compartilhamento de tela.

5 Conclusão

O intuito deste trabalho, consistiu em realizar a configuração e testar o BigBlueButton, a fim de analisar os fluxos de mídia que compõem uma Conferência Web. Também foram observados os pontos onde a ferramenta pode ser aplicada no mundo estudantil e como ela se torna uma ótima opção quando falamos de controle de congestionamento de rede.

O BigBlueButton, se mostrou uma ferramenta muito acessível desde a sua instalação e configuração, onde não foram encontradas dificuldades muito amplas para deixar a ferramenta operante. A partir do momento em que as máquinas virtuais foram criadas obedecendo os requisitos mínimos de funcionamento, configuração do BigBlueButton ocorreu de forma muito rápida e prática, com isso sua implantação em ambientes onde não se possui uma mão-de-obra muito especializada se torna mais fácil.

Em vista das análises realizadas, é possível concluir que o BigBlueButton desde o momento de sua concepção, já buscava um melhor desempenho e uma menor alocação de recursos de rede, uma vez que os desenvolvedores permitiram que o próprio usuário selecionasse qual a qualidade de transmissão de vídeo que mais se adapta à velocidade de sua conexão de internet. Ainda com base nos testes e análises realizadas, o BigBlueButton se mostrou uma ferramenta bastante dinâmica, permitindo a integração de telefonia convencional em uma conferência, assim o usuário consegue estar conectado independente da presença de um computador.

No meio de Conferências Web o BigBlueButton se mostrou uma ótima opção para escolas, universidades e até mesmo ambientes corporativos, pois ele é dotado de muitos recursos que facilitam a usabilidade e se adaptam às necessidades específicas. Ele também possui uma grande vantagem que é a utilização de código livre, isso torna o BigBlueButton ainda mais customizável aos propósitos que se desejam atingir, como é o exemplo da ferramenta MConf.

Para finalizar, como conclusão final com base nas análises feitas, que, o BigBlueButton é uma ferramenta muito poderosa e versátil, visando simplificar e prover facilidades empregando funcionalidades que aproveitam ao máximo os recursos disponíveis, tanto computacionais quanto de rede, assim permitindo se ter muito mais usuários conectados simultaneamente e desfrutando de serviços de qualidade.

5.1 Trabalhos Futuros

Por fim, apresentam-se as propostas para trabalhos futuros, como exemplo, com base nos fluxos analisados neste trabalho, é possível realizar melhorias ainda mais efetivas no controle de congestionamento da ferramenta BigBlueButton, como os fluxos que demandam uma maior largura de banda já foram identificados, é possível aplicar técnicas onde a utilização do destes fluxos seja melhor aproveitada, como exemplo desenvolver um controle de apresentação realizado pelo moderador onde apenas uma pessoa poderá apresentar sua tela. Também é possível desenvolver uma funcionalidade que o próprio servidor delimita a qualidade do vídeo de todos os participantes uma vez que seja perceptível a degradação dos serviços. Ao fim das melhorias realizadas é possível realizar o commit das melhorias e disponibilizá-las aos demais colaboradores do GitHub.

Outro possível trabalho futuro pode ser desenvolvido dando ênfase na experiência do usuário, onde a interface do BigBlueButton poderá ser incrementada de um painel que mostra qual o nível de consumo de banda de cada um dos canais conectados. Isso permite ao usuário julgar qual dos canais ele pode abrir mão, para que assim os demais canais ganhem mais qualidade. Uma vez que o próprio usuário sabe qual o cunho da Conferência Web que está participando, assim caso o áudio seja primordial, é possível abrir mão do vídeo para que os demais sejam preservados.

Referências

- ABED, A. B. D. E. A. D. *Relatório Analítico de Aprendizagem a Distância no Brasil*. 2016. Disponível em: <http://abed.org.br/censoead2016/Censo_EAD_2016_portugues.pdf>. Acesso em: 20 set. 2018. Citado na página 12.
- BARCELAR, R. R. Congestionamento em redes tcp. p. 1–5, 2012. Acesso em: 18 out. 2018. Citado 2 vezes nas páginas 12 e 18.
- BIGBLUEBUTTON INC. *BigBlueButton*. 2015. Disponível em: <<https://bigbluebutton.org>>. Acesso em: 13 set. 2018. Citado 2 vezes nas páginas 12 e 17.
- CAVALCANTI, J. L. Análise comparativa dos algoritmos de controle de congestionamento do tcp. p. 11–25, 2005. Disponível em: <<http://tcc.ecomp.poli.br/JulianaCavalcanti.pdf>>. Acesso em: 24 out. 2018. Citado na página 15.
- COSTA, F. *Você sabe a Diferença entre Software Livre e Código Aberto?* 2014. Disponível em: <<https://novaescola.org.br/conteudo/4593/voce-sabe-a-diferenca-entre-software-livre-e-codigo-aberto>>. Acesso em: 10 set. 2018. Citado na página 12.
- DANTAS, M. B. T. Melhoria de desempenho do protocolo tcp em canais de hf via escolha de parâmetros e emprego de técnicas de controle de erros. 2006. Disponível em: <<http://tcc.ecomp.poli.br/JulianaCavalcanti.pdf>>. Acesso em: 08 nov. 2018. Citado na página 16.
- DOTTA, S. Uso de webconferência em educação a distância. p. 01–06, 2009. Acesso em: 12 set. 2018. Citado na página 12.
- FERREIRA, C. Controle de congestionamento. 2008. Disponível em: <https://www.projetoderedes.com.br/artigos/artigo_controle_de_congestionamento.php>. Acesso em: 21 nov. 2018. Citado na página 17.
- FONSECA, J. L. A. D.; STANTON, M. A. *Estudo Experimental de Videoconferência Pessoal em Inter-redes IP com QoS*. 2001. Disponível em: <https://memoria.rnp.br/newsgen/0111/jl_wtr.html>. Acesso em: 15 out. 2018. Citado na página 13.
- KICH, S. Integração de ferramenta de videoconferência ao moodle. p. 13–16, 2015. Acesso em: 12 set. 2018. Citado 2 vezes nas páginas 12 e 18.
- MORAES, M. C. *Educação a Distância: Fundamentos e Práticas*. Campinas, São Paulo: MARIA CANDIDA MORAES, 2002. (4. edição). ISBN 85-88833-01-8. Acesso em: 25 nov. 2018. Citado na página 18.
- PARMAR, H.; THORNBURGH, M. *Estudo experimental de videoconferência pessoal em inter-redes IP com QoS*. 2012. Disponível em: <http://www.images.adobe.com/www.adobe.com/content/dam/acom/en/devnet/rtmp/pdf/rtmp_specification_1.0.pdf>. Acesso em: 12 jan. 2019. Citado na página 24.

PINHEIRO, J. M. S. Afinal, o que é qualidade de serviço? 2004. Disponível em: <https://www.projetoderedes.com.br/artigos/artigo_qualidade_servico.php>. Acesso em: 20 nov. 2018. Citado na página 16.

REIS, M. Ead como instrumento de inclusão social. 2015. Acesso em: 09 out. 2018. Citado na página 12.

SILVA, J. A.; PEREIRA, S. B. S. Análise da webconferência como objeto de aprendizagem no curso de licenciatura em artes visuais - ead - ufes. 2002. Disponível em: <<http://sistemas3.sead.ufscar.br/ojs/Trabalhos/261-792-1-ED.pdf>>. Acesso em: 14 fev. 2019. Citado na página 15.

TANENBAUM, A. S. *Redes de computadores*. Amsterdam, Holanda: Prentice Hall, 2002. (4. edição). ISBN 10: 0130661023 AND 13: 9780130661029. Acesso em: 15 nov. 2018. Citado na página 16.

Apêndices

APÊNDICE A – Wireshark

O principal recurso utilizado neste trabalho como instrumento de captura e análise dos dados foi o Wireshark. Com esta ferramenta, é possível capturar todo o tráfego que flui pelas interfaces da máquina em que se está analisando, além desta funcionalidade o Wireshark ainda permite que sejam realizadas análises aprofundadas nos dados capturados, sendo algumas delas a segmentação por protocolos, criação de filtros específicos para qualquer tipo de dado capturado, análise de sequências de fluxo de protocolos de telefonia, criação de gráficos em tempo real com base em filtros personalizados entre outras inúmeras funções.

O Wireshark é uma ferramenta disponibilizada gratuitamente e possui uma documentação muito completa e bem especificada, é possível encontrar a documentação e realizar o download desse software através do link: <https://www.wireshark.org/> .