

**Nicole da Rosa Feijó**

***Rede de Telefonia Colaborativa com VoIP***

São José – SC

Julho / 2015

**Nicole da Rosa Feijó**

***Rede de Telefonia Colaborativa com VoIP***

Monografia apresentada à Coordenação do Curso Superior de Tecnologia em Sistemas de Telecomunicações do Instituto Federal de Santa Catarina para a obtenção do diploma de Tecnólogo em Sistemas de Telecomunicações.

Orientador:  
Prof. Marcelo Maia Sobral, Dr.

CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS DE TELECOMUNICAÇÕES  
INSTITUTO FEDERAL DE SANTA CATARINA

São José – SC

Julho / 2015

Monografia sob o título “*Rede de Telefonia Colaborativa com VoIP*”, defendida por Nicole da Rosa Feijó e aprovada em 01 de julho de 2015, em São José, Santa Catarina, pela banca examinadora assim constituída:

---

Prof. Marcelo Maia Sobral, Dr. Eng.  
Orientador

---

Prof. Eraldo Silveira e Silva, Dr. Eng.  
IFSC

---

Prof. Ederson Torresini, Me.  
IFSC

# *Resumo*

Hoje com o crescente uso da internet, a tecnologia VoIP vem se desenvolvendo cada vez mais. O custo das chamadas DDD através da telefonia comum (PSTN) ainda é elevado, se comparado ao custo praticamente nulo de chamadas VoIP. A proposta deste trabalho é especificar uma rede colaborativa para realização de chamadas DDD usando VoIP com tecnologia SIP, na qual seus usuários participantes compartilham suas linhas telefônicas. Nessa rede colaborativa, cada usuário possui um PBX IP que integra sua linha telefônica, seus aparelhos telefônicos e o envio e recebimento de chamadas VoIP via Internet. A integração entre todos os usuários participantes se faz por meio de um servidor chamada de contabilizador, o qual incorpora um proxy SIP. Assim, uma chamada DDD iniciada por um usuário participante é encaminhada com uso de VoIP até o contabilizador, que a encaminha para o PBX de outro usuário participante na cidade de destino, e dali via PSTN até o número telefônico chamado. Um protótipo dessa rede colaborativa foi desenvolvido para demonstrar o encaminhamento de chamadas entre os usuários.

# *Abstract*

Today with the increasing use of internet, VoIP technology been developing more and more. The cost of DDD calls over ordinary telephony (PSTN) is still high when compared to virtually no cost VoIP calls. The purpose of this work is to specify a collaborative network for performing DDD calls using VoIP with SIP technology, in which participants users share their phone lines. In this collaborative network, each user has an IP PBX that integrates your phone line, your phones and sending and receiving VoIP calls over the Internet. The integration between all participating users is done through a call server contabilizador, which incorporates a SIP proxy. So a DDD call launched by a participant user is forwarded to VoIP use until the contabilizador, which forwards it to the other participant user PBX in the destination city, and from there via PSTN to the telephone number called. A prototype of this collaborative network is designed to demonstrate the forwarding of calls between users.

# *Sumário*

## **Lista de Figuras**

## **Lista de Tabelas**

<b>1</b>	<b>Introdução</b>	p. 10
1.1	Objetivos . . . . .	p. 11
<b>2</b>	<b>Fundamentação Teórica</b>	p. 12
2.1	Protocolos para VoIP . . . . .	p. 12
2.1.1	SIP (Session Initiation Protocol) . . . . .	p. 12
2.1.2	RTP (Real Time Protocol) . . . . .	p. 18
2.2	Modelos para Infraestrutura VoIP . . . . .	p. 19
2.2.1	Descentralizado . . . . .	p. 20
2.2.2	Centralizado . . . . .	p. 20
2.3	PBX IP . . . . .	p. 21
2.3.1	Asterisk . . . . .	p. 22
2.3.2	FreeSwitch . . . . .	p. 25
<b>3</b>	<b>Rede telefônica VoIP colaborativa</b>	p. 28
3.1	Arquitetura do Sistema . . . . .	p. 28
3.1.1	Encaminhamento de chamadas . . . . .	p. 30
3.1.2	Mecanismo de Crédito . . . . .	p. 31
3.2	Protótipo do Sistema . . . . .	p. 33

3.2.1	Contabilizador . . . . .	p. 33
3.2.2	A Infraestrutura do Participante . . . . .	p. 37
<b>4</b>	<b>Implantação do Projeto e testes realizados</b>	<b>p. 41</b>
4.1	Usuário participante . . . . .	p. 41
4.2	Contabilizador . . . . .	p. 42
<b>5</b>	<b>Conclusões</b>	<b>p. 47</b>
	<b>Referências Bibliográficas</b>	<b>p. 48</b>

# *Lista de Figuras*

2.1	Componentes de uma entidade SIP . . . . .	p. 14
2.2	Exemplo de UAS e UAC ( <a href="http://mobilezoo.biz/jsr/180/overview-summary.html">http://mobilezoo.biz/jsr/180/overview-summary.html</a> )	p. 14
2.3	Exemplo de uma conversação entre dois agentes de usuários utilizando Proxy Server . . . . .	p. 15
2.4	Exemplo de uso do Redirect Server ( <a href="http://www.kamailio.org/pub/ser/ser-releases/0.8.14/doc/html/sip-introduction.html">http://www.kamailio.org/pub/ser/ser-releases/0.8.14/doc/html/sip-introduction.html</a> ) . . . . .	p. 16
2.5	Exemplo de como é feito o registro do usuário ( <a href="http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/voice/sip/configuration/15-mt/sip-config-15-mt-book/voi-sip-aaa.html">http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/voice/sip/configuration/15-mt/sip-config-15-mt-book/voi-sip-aaa.html</a> )	p. 16
2.6	Exemplo de mensagens SIP durante uma chamada ( <a href="http://blog.sippulse.com/entenda-como-funciona-um-dialogo-sip-protocolo-utilizado-em-ligacoes-voip/">http://blog.sippulse.com/entenda-como-funciona-um-dialogo-sip-protocolo-utilizado-em-ligacoes-voip/</a> ) . . . . .	p. 18
2.7	Exemplo do protocolo RTP em uma chamada VoIP . . . . .	p. 19
2.8	Exemplo de uma conversação utilizando modelo descentralizado . . . . .	p. 20
2.9	Exemplo de um modelo centralizado utilizando PBX . . . . .	p. 21
2.10	Exemplo da arquitetura do Asterisk . . . . .	p. 22
2.11	Exemplo de um plano de discagem do Asterisk . . . . .	p. 23
2.12	Exemplo de utilização de entroncamento SIP . . . . .	p. 24
2.13	Exemplo de uma configuração VoIP com FreeSwitch e mensagens SIP . . . . .	p. 26
3.1	Componentes do sistema . . . . .	p. 29
3.2	Diagrama do sistema . . . . .	p. 30
3.3	Encaminhamento de chamadas . . . . .	p. 31
3.4	Exemplo de valores de crédito e minutos disponíveis dos dois usuários participantes no início da chamada . . . . .	p. 32



3.5	Exemplo de valores de crédito e minutos disponíveis atualizados ao fim da chamada, supondo que a ligação durou 5 minutos . . . . .	p. 33
3.6	Arquitetura do Contabilizador . . . . .	p. 34
3.7	Fluxograma do plano de discagem do contabilizador . . . . .	p. 36
3.8	Fluxograma do encerramento da chamada . . . . .	p. 37
3.9	Fluxograma da atualização mensal de crédito e minutos disponíveis . . . . .	p. 37
3.10	A Infraestrutura do participante . . . . .	p. 38
3.11	Registro do usuário participante na rede . . . . .	p. 39
3.12	Fluxograma do encaminhamento das chamadas . . . . .	p. 40
4.1	Informações do usuário para se registrar no contabilizador . . . . .	p. 41
4.2	Contexto <i>troncoSip</i> implementado no plano de discagem de um usuário . . . . .	p. 42
4.3	Informação do nome do programa externo com configuração de autenticação de usuários . . . . .	p. 43
4.4	Como foi criada a tabela no <i>sqlite</i> com as informações de usuários . . . . .	p. 43
4.5	Informação do nome de arquivo do plano de discagem . . . . .	p. 43
4.6	Atualização de créditos ao final da chamada . . . . .	p. 45
4.7	Atualização de crédito mensalmente . . . . .	p. 46

## *Lista de Tabelas*

- 2.1 Exemplos de mensagens de pedido SIP . . . . . p. 17
- 2.2 Exemplos de mensagem de resposta SIP . . . . . p. 17
- 3.1 Exemplo de usuários na base de dados sqlite . . . . . p. 35

# 1 *Introdução*

A utilização da Internet como meio de comunicação de voz vem aumentando e tem se tornado uma opção econômica se comparada à telefonia convencional, principalmente para chamadas de longas distâncias. Comparada com a telefonia comutada (PSTN), que transmite o áudio em sinal analógico, a tecnologia VoIP (*Voice over IP*) converte esse sinal de áudio analógico em sinal digital que pode ser transmitido através da Internet, usando protocolos específicos.

Para uma chamada VoIP ser efetuada é necessário a utilização de alguns protocolos, como SIP e RTP. O protocolo SIP é responsável pela sinalização, permitindo que um usuário convide outro para estabelecer uma chamada. O RTP é a transmissão de mídia, ou seja, é o transporte da voz digitalizada.

O aumento nas taxas de transmissões da internet e a produção de equipamentos destinados a tecnologia proporcionaram uma melhoria considerável na qualidade dos serviços de telefonia VoIP. Como consequência obteve-se um crescimento mundial na utilização da tecnologia (VOCALTEC, 2008).

Já que a transmissão de dados é via IP, não se usa comutação de circuitos e sim comutação de pacotes. A vantagem da comutação de pacotes é que a capacidade da rede não fica reservada ao se fazer uma chamada. Essa capacidade é ocupada somente durante a transmissão de cada pacote de voz codificada, ficando ociosa nos intervalos de silêncio que existem durante uma chamada. Portanto é possível fazer outras chamadas ao mesmo tempo, de forma a aproveitar a capacidade ociosa da rede.

Uma característica da telefonia atual é o custo das ligações interurbanas, considerado ainda elevado se comparado com o custo de ligações locais.

Se as chamadas interurbanas fossem feitas com VoIP tornaria possível a realização de chamadas interurbanas com preço de ligação local. Isso pode ser provido por um sistema em que usuários com acesso a Internet compartilham suas linhas telefônicas.

Neste trabalho propõe-se um modelo de uma rede de telefonia VoIP colaborativa, em que

seus usuários participantes compartilham suas linhas telefônicas comuns. Dessa forma, uma pessoa de Florianópolis pode ligar para alguém de São Paulo, sendo que a chamada é feita via VoIP até a rede de outro participante, e dali o PBX dessa pessoa de São Paulo encaminha essa ligação via PSTN, fazendo com que o custo seja de ligação local.

## 1.1 Objetivos

O objetivo deste trabalho é criar uma rede colaborativa com VoIP onde usuários participantes poderão efetuar ligação DDD com o custo de ligação local.

- Os usuários participantes precisam ceder sua rede de telefonia local (PSTN) e terem PBX configurado;
- A rede irá premiar o usuário que mais ceder tempo de sua linha, sendo que cada usuário terá um limite estabelecido de tempo para efetuar ligação e sempre que ceder sua linha telefônica para outro usuário, o tempo da chamada será convertido para o usuário intermediador poder efetuar ligação em outros momentos;
- A rede terá um contabilizador que fará um papel de *proxy* SIP intermediando todas as chamadas.

## 2 *Fundamentação Teórica*

VoIP, que significa voz sobre IP, ou ainda *voice over IP*, é a tecnologia que possibilita transmitir áudio através da Internet. Isso funciona porque o sinal de áudio analógico transmitido é transformado em sinal digital, e esse pacote de voz codificado digitalmente é transmitido ao destino.

O conceito VoIP tomou forma em meados da década de 1990, quando surgiu o primeiro software comercial, o Internet Phone, a permitir troca de pacotes IP transportando amostras de voz entre computadores pessoais (COLCHER et al., 2005).

Serviços de VoIP se enquadram, na terminologia da ANATEL, como *Serviços de Comunicação Multimídia* (SCM). Esses serviços contêm restrições, e a principal delas é o impedimento de uma instância de uso desse serviço iniciar e terminar simultaneamente chamadas telefônicas na rede de telefonia pública. Isso permite a oferta de serviços VoIP no âmbito de uma rede privada e apenas com uma das pontas na rede telefônica pública, o que dá margem à redução de custos (COLCHER et al., 2005).

### 2.1 Protocolos para VoIP

Os protocolos são responsáveis pela comunicação entre os equipamentos utilizados na tecnologia VoIP. Durante uma chamada, os principais componentes são:

- Sinalização das chamadas telefônicas (SIP);
- Controle e transporte de mídia (RTP).

#### 2.1.1 SIP (Session Initiation Protocol)

O protocolo SIP, um padrão da Internet definido na RFC 2543, tem por finalidade estabelecer, manter e terminar sessões de comunicação. Em sistemas VoIP, é usado para iniciar

e terminar chamadas, fornecendo também suporte a negociação de parâmetros relacionados a codificação e transporte de áudio. Esse protocolo pode seguir o modelo P2P (*peer-to-peer*) em que dois ou mais participantes, chamados de agentes, trocam mensagens com a finalidade de estabelecerem uma sessão. De acordo com (JOHNSTON, 2009), as principais funções do SIP são:

- Localização de um ponto final (usuário);
- Entrar em contato com um ponto final para determinar a vontade de estabelecer uma sessão;
- Troca de informações de mídia para permitir que a sessão seja estabelecida;
- Modificação das sessões de mídia existentes;
- Subdivisão das sessões de mídia existentes.

SIP também foi estendido para solicitar e enviar notificações de presença (informações de status e de localização), bem como sessões de mensagens instantâneas (JOHNSTON, 2009). Embora possa utilizar o protocolo *Transmission Control Protocol* (TCP) e o *Stream Control Transmission Protocol* (SCTP), o SIP é mais utilizado sobre o protocolo *User Datagram Protocol* (UDP) (OLIVER, 2005).

### **Entidade SIP**

Uma rede SIP é constituída por entidades SIP. Cada entidade tem uma função específica e participa na comunicação SIP como um cliente (solicitando pedidos), um servidor (respondendo os pedidos) ou ambos. Exemplo de entidades SIP, como mostra a figura 2.1: User Agents, Proxy Server, Redirect Server e Registrar Servers (COLCHER et al., 2005).

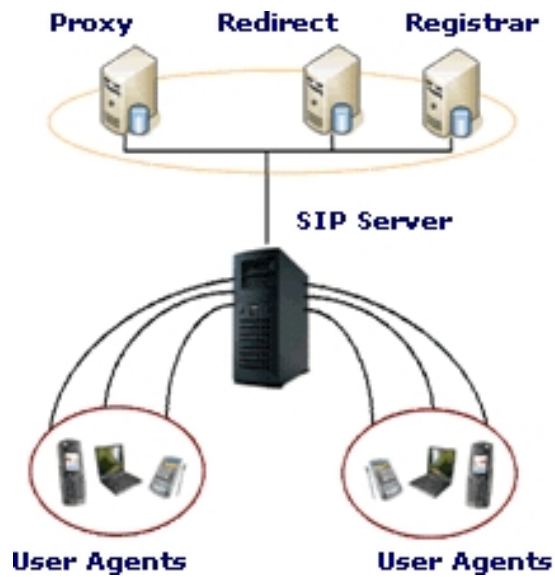


Figura 2.1: Componentes de uma entidade SIP

### User Agents (UA) – Agentes de Usuário

Um agente de usuário SIP se divide em em duas partes: i) a que inicia requisições se chama UAC (*User Agent Client*), e ii) a que responde requisições é denominada UAS (*User Agent Server*). Durante uma sessão um agente de usuário geralmente irá operar como UAC e UAS, ou seja, irá operar como cliente e como servidor. Agentes de usuário SIP normalmente se registram com um servidor SIP Registrar no seu domínio. A figura 2.2 demonstra UAC e UAS se comunicando por meio de mensagens SIP .

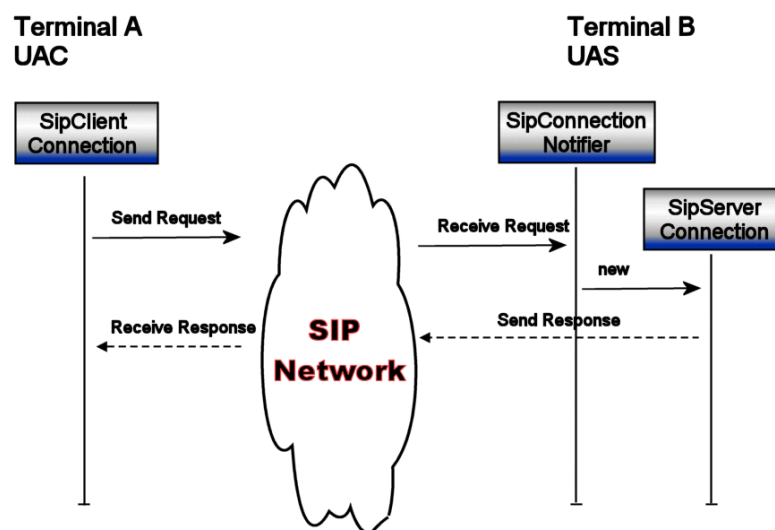


Figura 2.2: Exemplo de UAS e UAC (<http://mobilezoo.biz/jsr/180/overview-summary.html>)

### Proxy Server (Servidor Proxy)

Um servidor proxy SIP recebe uma solicitação SIP de um agente do usuário ou outro proxy e age em nome do UA em encaminhar ou responder à solicitação. Assim como um roteador encaminha pacotes IP na camada IP, um proxy SIP encaminha mensagens SIP na camada de aplicação (JOHNSTON, 2009). A figura 2.3 exemplifica uma comunicação entre dois agentes de usuários sendo intermediada pelo proxy server.

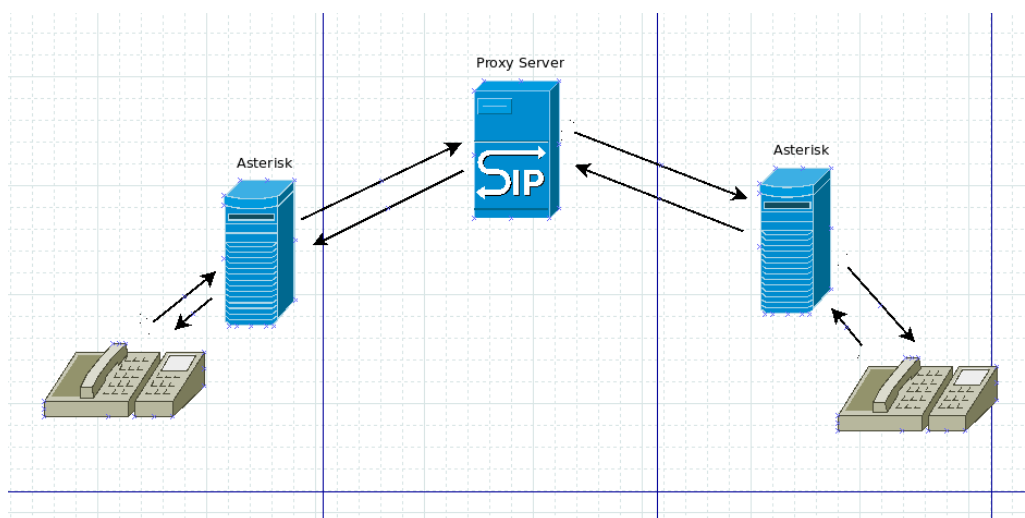


Figura 2.3: Exemplo de uma conversação entre dois agentes de usuários utilizando Proxy Server

### Redirect Server (Servidor Redirecionador)

No modo Redirect Server, como mostra a figura 2.4, os usuários se falam diretamente, sem a intermediação do servidor. Quando um usuário quiser efetuar uma chamada, o servidor irá localizar o usuário destino. Após fazer essa localização, o servidor encaminha uma mensagem para o usuário chamador informando a localidade do destino, dessa forma os dois usuários podem se falar diretamente (JOHNSTON, 2009).



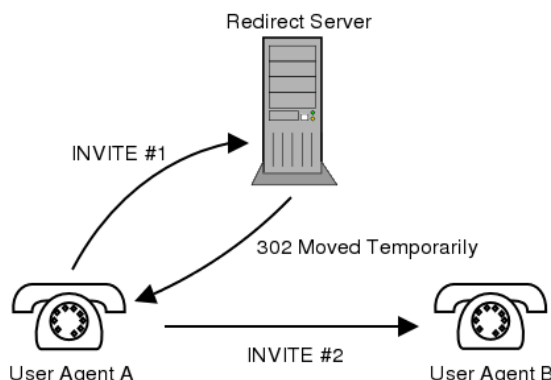


Figura 2.4: Exemplo de uso do Redirect Server  
([http://www.kamailio.org/pub/ser/ser-releases/0.8.14/doc/html/sip\\_introduction.html](http://www.kamailio.org/pub/ser/ser-releases/0.8.14/doc/html/sip_introduction.html))

### Registrar Server (Servidor de Registro)

Ao iniciar uma chamada para um destino, o UA precisará se registrar para o servidor ter informações sobre sua localidade, dessa forma o UA manda uma mensagem (REGISTER) para o servidor SIP, como mostra a figura 2.5, com suas informações de usuário, incluindo localização. Dessa forma o servidor terá as informações necessárias para localizar o usuário e estabelecer a chamada.

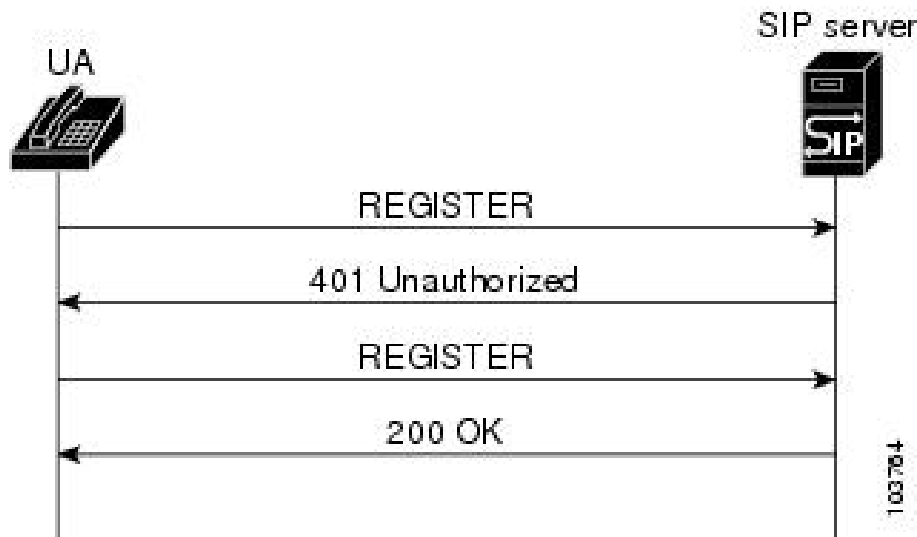


Figura 2.5: Exemplo de como é feito o registro do usuário  
(<http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/voice/sip/configuration/15-mt/sip-config-15-mt-book/voi-sip-aaa.html>)

### Mensagens SIP

O protocolo SIP possui um conjunto de mensagens específicas para o estabelecimento de sessões. Essas mensagens são divididas em dois grupos: mensagens de pedido, que são enviadas

pelo UAC, e mensagens de resposta, enviadas pelo UAS (OLIVER, 2005).

As mensagens SIP seguem a RFC 822 e são compostas por três campos: linha inicial, cabeçalhos e corpo da mensagem. O campo linha inicial identifica o tipo de mensagem, quando a mensagem é um pedido, a linha inicial inclui uma URI de requisição que indica o usuário ao qual este pedido está sendo encaminhado. O campo cabeçalho é usado para transportar os atributos da mensagem e modificar o significado deles. Por último, o corpo da mensagem é usado para descrever a sessão a ser iniciada. Os corpos de mensagens podem aparecer no pedido e em mensagens de resposta (NELSON, 2002). As tabelas 2.1 e 2.2 listam as principais mensagens de pedido e resposta, respectivamente.

Mensagens	Descrição
INVITE	Convida uma entidade a iniciar uma sessão
ACK	Confirma respostas finais a requisição INVITE
REGISTER	Registra as informações de usuário
CANCEL	Cancela pedidos enviados que não obtiveram resposta no tempo determinado
OPTIONS	Obtém informações sobre o agente
BYE	Termina uma sessão estabelecida

Tabela 2.1: Exemplos de mensagens de pedido SIP

Tipo de Resposta	Código de Status
Informativo	1XX
Êxito	2XX
Redirecionamento	3XX
Falha no Cliente	4XX
Falha no Servidor	5XX
Falha Global	6XX

Tabela 2.2: Exemplos de mensagem de resposta SIP

As trocas de mensagens seguem uma hierarquia. Uma mensagem de pedido seguida de suas mensagens de resposta temporárias (opcionais) e final formam uma transação. O conjunto de transações desde o início até o final de uma sessão formam um diálogo. E o conjunto de diálogos de uma sessão formam uma chamada.

A figura 2.6 mostra um exemplo de uma chamada entre dois usuários. As mensagens SIP são intermediadas pelo Proxy SIP e a mídia é encaminhada direto entre os usuários. Sempre que o ramal destino estiver disponível e receber a requisição de chamada (INVITE) ele retorna o

Ring (180 Ringing) informando que está chamando. Caso este ramal esteja ocupado, não esteja registrado ou por algum outro motivo não possa atender ele irá retornar um código de erro.

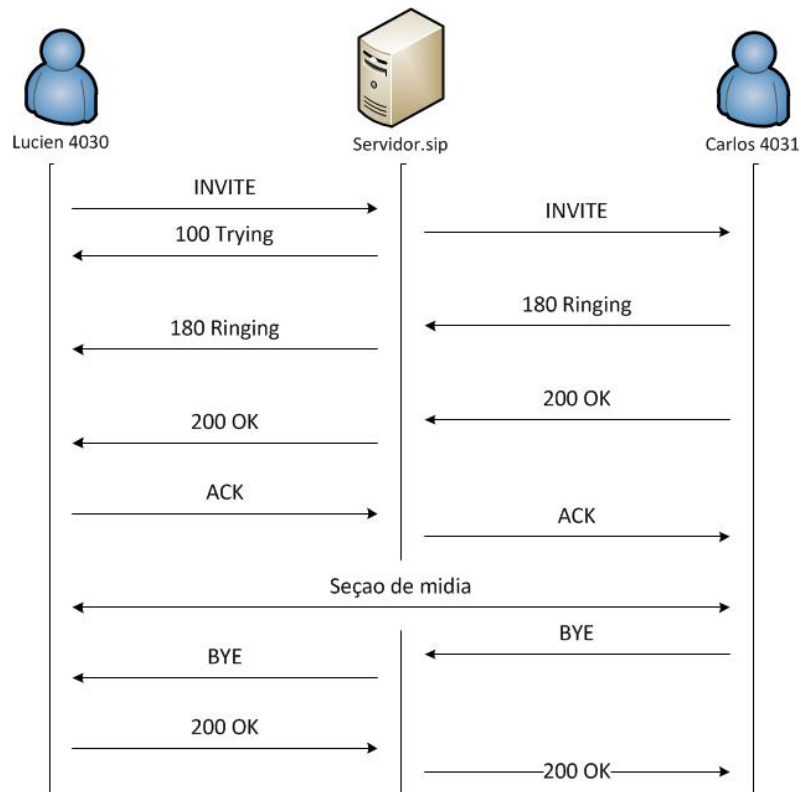


Figura 2.6: Exemplo de mensagens SIP durante uma chamada (<http://blog.sippulse.com/entenda-como-funciona-um-dialogo-sip-protocolo-utilizado-em-ligacoes-voip/>)

### 2.1.2 RTP (Real Time Protocol)

O protocolo RTP, definido pela IETF (Internet Engineering Task Force), segue a RFC 3550 e é utilizado em aplicações de tempo real, como transmissão de áudio e vídeo ponto a ponto e utiliza o UDP como protocolo de transporte. O RTP ajuda a informar os tempos em que cada mensagem foi transmitida e define como deve ser feita a fragmentação do fluxo de dados, adicionando a cada fragmento informação de sequência e de tempo de entrega.

O endereço IP, porta e codecs suportados nas *streams* de áudio são informadas através da mensagem SDP (Session Description Protocol). Essas mensagens são encapsuladas dentro das mensagens SIP INVITE e informam o tipo de chamada solicitada, como: áudio, vídeo.

Quando uma chamada VoIP é iniciada, sempre é encaminhada uma mensagem SIP para informar que um usuário está querendo completar uma chamada, dentro dessa mensagem SIP é informado através do SDP que é uma mensagem de áudio. Quando o usuário destino aceita a

chamada e ela então é completada, o protocolo RTP, responsável por encaminhar o áudio, fará todo o transporte da mídia de um usuário para outro.

Ao transmitir as *streams* de áudio, o protocolo deve lidar com as seguintes condições da rede:

- A rede pode tirar os pacotes da sequência;
- Alguns pacotes podem ser perdidos;
- Ocorrência de *jitter*. *Jitter* é a variação do atraso da entrega dos pacotes;

Se um pacote se perde numa transmissão em tempo real, não vale a pena recuperá-lo, pois só atrapalharia no entendimento do receptor, já que pode chegar fora de ordem no destino. No caso das chamadas VoIP, o áudio codificado é encapsulado em mensagens RTP para que possa ser transportado entre os agentes de usuário.

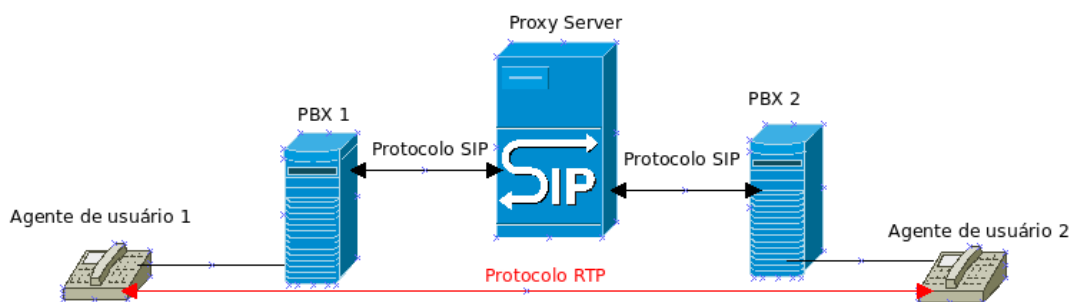


Figura 2.7: Exemplo do protocolo RTP em uma chamada VoIP

## 2.2 Modelos para Infraestrutura VoIP

No contexto deste trabalho, dois modelos para uma infraestrutura VoIP podem ser identificados:

- **Descentralizado:** Onde a comunicação ocorre diretamente entre os agentes de usuário;
- **Centralizado:** Onde a comunicação é intermediada por um ou mais servidores proxy.

### 2.2.1 Descentralizado

No modelo descentralizado os agentes de usuários podem se comunicar diretamente, como mostra figura 2.8, sem o uso de provedores para intermediar esta comunicação. Dessa forma, os agentes agem como cliente e servidor, iniciando e respondendo requisições SIP. Um modelo como esse depende de existir uma forma de localizar contatos SIP a partir de suas URI SIP. A localização do usuário é essencial para que a chamada seja iniciada, visto que um usuário necessitam saber o endereço IP do outro usuário para poder lhe encaminhar uma mensagem INVITE diretamente. Isso aumenta a complexidade desse modelo, visto que mecanismos de localização adicionais são necessários para que os usuários possam se localizar.

Um exemplo de mecanismo para fins de localização de usuário é o ENUM (E.164 Number to URI Mapping), que mapeia números telefônicos para endereços de internet (contatos SIP) usando a infraestrutura DNS (*Domain Name System*) existente. Outra forma de localizar usuários, baseada também em DNS, implica cada usuário possuir um domínio ou nome DNS, e a ele associar um registro SRV (*Service*) para indicar o UAS responsável por receber chamadas. Ambos exemplos evidenciam a necessidade de criar uma infraestrutura de localização adicional, incrementando a complexidade do sistema.

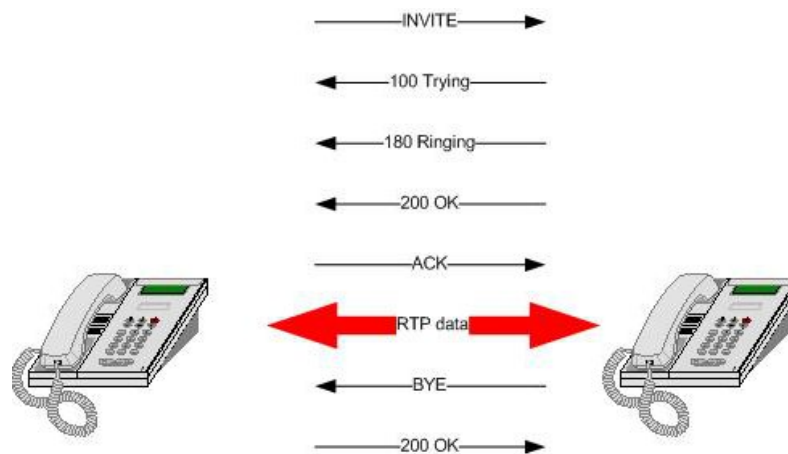


Figura 2.8: Exemplo de uma conversação utilizando modelo descentralizado

### 2.2.2 Centralizado

No modelo centralizado a comunicação entre os agentes de usuários ocorre através da intermediação de um proxy SIP. Na figura 2.9 ilustra-se um modelo centralizado utilizando um proxy SIP. Esse proxy contém as informações de localização dos usuários e é responsável por fazer o encaminhamento de chamadas de um agente de usuário para o outro.

Neste modelo, é necessário que haja um ou mais proxy SIP ou gateway de mídia. Nestes dois casos as chamadas são intermediadas, ou seja, não são realizadas diretamente entre os agentes de usuário. Para fins de localização, os agentes devem se registrar previamente em um servidor SIP. A decisão sobre como localizar e contatar o destino é tomada pelo servidor, que pode funcionar como um proxy SIP ou como um gateway de mídia (JOHNSTON, 2009).

- Proxy SIP: Um servidor proxy SIP recebe uma solicitação SIP de um agente do usuário ou outro proxy e age em nome do UA em encaminhar ou responder à solicitação. Assim como um roteador encaminha pacotes IP na camada IP, um proxy SIP encaminha mensagens SIP na camada de aplicação (JOHNSTON, 2009).
- Gateway de mídia: sua principal função é fazer a conversão de diferentes tipos de transmissão e codificação. Durante uma chamada, para que as *streams* de áudio sejam entendidas em ambos os lados, é necessário que os dois usuários estejam utilizando o mesmo codec, o gateway de mídia possibilita que os usuários usando diferentes codecs consigam se comunicar, devido a conversão de codificação que ele é capaz de fazer.

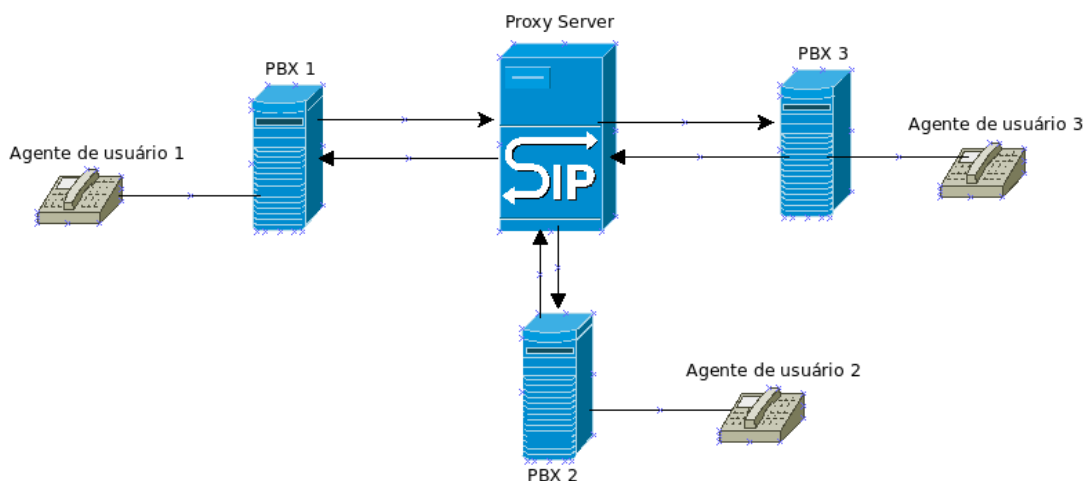


Figura 2.9: Exemplo de um modelo centralizado utilizando PBX

## 2.3 PBX IP

PBX é um termo usado para denominar centrais telefônicas privadas, usadas tipicamente por empresas para implantar suas redes telefônicas internas e integrá-las com a rede telefônica pública. Quando esse tipo de central é capaz de receber e realizar chamadas VoIP, passa a ser chamado de PBX IP.

O Servidor PBX IP também funciona como um Proxy SIP, pois ele é capaz de intermediar chamadas entre outros agentes SIP. Existem vários softwares que implementam PBX IP, citando-se como exemplos Asterisk e Freeswitch.

### 2.3.1 Asterisk

O Asterisk é um software livre e foi desenvolvido pela empresa Digium. Ele implementa os recursos encontrados em PBX convencional, utilizando a tecnologia VoIP. O Asterisk usa canais para realizar as sinalizações das chamadas telefônicas na rede IP. Pode ser utilizado como gateway de mídia, correio de voz, entre outras funcionalidades típicas de PBX (MADSEN; SMITH; MEGGELEN, 2005).

O Asterisk é usado por pequenas e grandes empresas, call centers, operadoras e agências governamentais, em todo o mundo (asterisk.org).

#### Arquitetura

Conforme mostra a figura 2.10, o Asterisk é formado por 4 blocos: Canais, Aplicações, Codificação e Formato de arquivos.

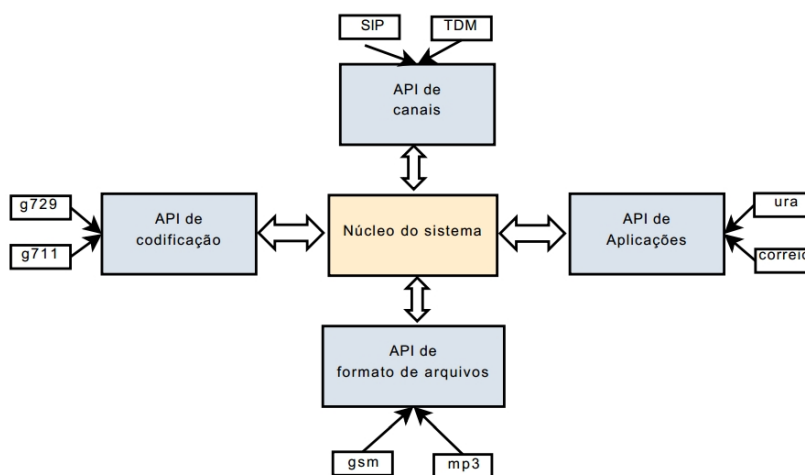


Figura 2.10: Exemplo da arquitetura do Asterisk

#### Plano de Discagem

O plano de discagem é verdadeiramente o coração de qualquer sistema Asterisk, já que ele define como o Asterisk trata as chamadas que chegam e que são enviadas. Ele consiste

em uma lista de instruções ou passos que o Asterisk deve seguir para processar chamadas. Ao contrário de centrais telefônicas tradicionais, o plano de discagem do Asterisk é completamente personalizável (MADSEN; SMITH; MEGGELEN, 2005).

O plano de discagem do Asterisk é composto por extensões, as quais são agrupadas em contextos:

- **Extensões:** são instruções que o Asterisk irá seguir, acionadas por uma chamada de entrada ou por dígitos sendo discados num canal. As extensões especificam o que acontece às chamadas enquanto elas seguem seu caminho pelo plano de discagem. Uma extensão completa é formada pelos seguintes componentes: nome ou número da extensão, prioridade e aplicação (executa ação na chamada);
- **Prioridades:** cada extensão pode ter vários passos, chamados de prioridades. Cada prioridade é numerada sequencialmente, começando por 1. Cada prioridade executa uma aplicação específica;
- **Aplicações:** executam uma ação no canal em questão, tal como emitir som ou desligar uma chamada.
- **Contextos:** são chamados de grupos de extensões. Eles evitam que partes diferentes do plano de discagem interajam umas com as outras. Uma extensão definida em um contexto é completamente isolada das extensões em outro contexto, a não ser que a interação seja especificamente permitida (MADSEN; SMITH; MEGGELEN, 2005).

No exemplo da figura 2.11, *TroncoSIP* é o contexto, 100 e 101 são números que podem ser chamados por meio do Asterisk, *exten* é a extensão, *Dial* e *Hangup* são aplicações e 1 é a prioridade (o n representa o próximo número).

```
[ TroncoSIP ]
exten=>100,1,Dial(SIP/100)
same=>n, Hangup
exten=>101,1,Dial(SIP/101)
same=>n, Hangup
```

Figura 2.11: Exemplo de um plano de discagem do Asterisk

## Canais

Um canal é um recurso com uma tecnologia específica através do qual chamadas podem ser recebidas ou enviadas pelo PBX. De certa forma, trata-se de uma abstração do Asterisk para



representar os mecanismos e recursos envolvidos nas comunicações das chamadas. Alguns tipos de canais são:

- SIP: recebe e realiza chamadas VoIP usando o protocolo SIP, conforme mostra a figura 2.12;
- IAX2: recebe e realiza chamadas VoIP usando o protocolo IAX2 (Inter-Asterisk eXchange);
- H323: recebe e realiza chamadas VoIP usando protocolos H323, um dos protocolos mais antigos de VoIP;
- DAHDI: é o responsável pela comunicação entre aplicação e o hardware de telefonia, o que torna possível receber e encaminhar chamadas para a rede telefônica convencional e celular.



Figura 2.12: Exemplo de utilização de entroncamento SIP

## Aplicações

Aplicações são as ações que podem ser realizadas durante o processamento de encaminhamento de chamadas, como atender, discar, encerrar.

## Codificação

Os codecs são responsáveis por converter a voz em sinal digital, e vice-versa. No caso de VoIP, os codecs influenciam a quantidade de chamadas que podem ser colocadas em um mesmo enlace de dados. Para aumentar a quantidade de chamadas é necessário que a codificação da voz seja feita de uma forma que ocupe menor banda. Exemplos de codecs são:

- G.729a: 8Kbit;

- G.711 ulaw: usado nos EUA (64 Kbps);
- G.711 alaw: usado na Europa e Brasil (64 Kbps);
- GSM: Telefonia Celular (12-13 Kbps);

### 2.3.2 FreeSwitch

FreeSwitch é um software livre de comunicação, para a criação de produtos de voz e de mensagens. Assim como o Asterisk, o FreeSwitch consegue tornar um computador comum em um PBX, podendo funcionar como gateway de mídia e PBX IP. Ele é licenciado segundo a Mozilla Public License (MPL), e surgiu como uma alternativa ao Asterisk para a maioria das necessidades em aplicações de voz, desde roteamento de sinalização SIP e mídia RTP ou SRTP, até URAs e fácil integração com aplicações externas ([freeswitch.org](http://freeswitch.org)).

FreeSwitch foi originalmente concebido e implementado por Anthony Minessale com a ajuda de Brian West e Michael Jerris, todos eles ex-desenvolvedores do Asterisk. Atualmente uma comunidade de desenvolvedores contribui para o projeto. O FreeSwitch foi inicialmente criado tendo em vista limitações existentes no Asterisk, tais como a limitação na quantidade de chamadas simultâneas. Além disso, o FreeSwitch oferece uma interface de programação (API) para que desenvolvedores possam adicionar funcionalidades ou controlar o sistema.

O suporte ao protocolo SIP no FreeSwitch é implementado por meio do projeto Sofia-SIP. Trata-se de um projeto de código aberto que fornece uma interface de programação com o protocolo SIP, a qual é disponibilizada em uma biblioteca. No FreeSwitch essa biblioteca é utilizada em um módulo chamado `mod_sofia`. Todas as ações envolvendo chamadas SIP oferecidas por esse módulo podem ser acessadas por meio da API do FreeSwitch. Assim, desenvolvedores têm liberdade para controlar o comportamento do FreeSwitch quanto ao processamento dessas chamadas.

A arquitetura do FreeSwitch é dividida em aplicações, codecs e protocolos.

#### Aplicações

O FreeSwitch possui várias aplicações, que assim como o Asterisk, são as ações realizadas dentro de uma chamada.

## Codecs

FreeSwitch suporta vários codecs, que são responsáveis por codificar e decodificar arquivos de mídia, dentre eles: CELT (32 kHz e 48 kHz), G.722.1 (banda larga), G.722.1C (banda larga de 32 kHz), G.722 (banda larga), G.711, G.726 (16k, 24k, 32k, 48k).

## Protocolos

Protocolos são responsáveis por realizar as sinalizações entre os agentes de usuário em uma chamada, dentre os que o FreeSwitch suporta estão: SIP, RTP, SRTP, H.323, IAX2, entre outros.

## API do FreeSwitch (FSAPI)

A API é uma interface de programação, para que se possam escrever softwares que acessem as funcionalidades do FreeSwitch. Através da API é possível verificar algumas informações, como status do usuário ou a quantidade de usuários que estão registrados no momento. A API pode ser acessada usando diferentes linguagens de programação, tais como C, Python, Lua e Java.

A API do FreeSwitch permite acessar vários comandos no próprio console do FreeSwitch ou então em algum programa externo. Os comandos são fornecidos dentro de um módulo chamado *mod\_commands*.



Figura 2.13: Exemplo de uma configuração VoIP com FreeSwitch e mensagens SIP

## FreeSwitch X Asterisk

De acordo com freeswitch.org, o FreeSwitch foi inspirado pelo design modular do Apache Web Server, seus objetivos eram usar essa abordagem modular para obter maior escalabilidade e estabilidade através de múltiplas plataformas. A principal vantagem do FreeSwitch sobre o Asterisk é sua performance, já que suporta mais chamadas simultâneas, e isso ocorre pela

diferença no modelo de processamento de chamadas. A escalabilidade é maior pois o FreeSwitch suporta mais chamadas simultâneas do que o Asterisk.

## ***3 Rede telefônica VoIP colaborativa***

Este trabalho propõe a implantação de uma rede VoIP colaborativa, para possibilitar chamadas de longa distância gratuitas. Essa rede deve oferecer mecanismos para que seus participantes compartilhem suas linhas telefônicas, de forma a completarem chamadas de longa distância solicitadas por outros participantes. O trecho de longa distância dessas chamadas deve ser feito com VoIP. Por fim, essa rede colaborativa deve premiar os participantes que cedem suas linhas telefônicas por mais tempo, concedendo-lhes um crédito maior para realizar chamadas de longa distância.

### **3.1 Arquitetura do Sistema**

Esse sistema implementa uma rede de telefonia onde os usuários participantes poderão efetuar ligação DDD com custo de ligação local. Essa rede é composta por seus usuários e suas linhas telefônicas com acesso a PSTN, e contabilizador. Os usuários participantes efetuam, recebem e encaminham ligações. O contabilizador encaminha chamadas de usuários chamadores para usuários localizados na área que contém o número chamado. Por fim, usuários compartilham suas linhas telefônicas de forma a encaminharem chamadas para números de telefones locais

Como mostra na figura 3.1, o sistema é composto pelo usuário chamador, usuário chamado e contabilizador, além dos usuários compartilharem sua linha telefônica local (PSTN). O usuário chamador tem seu PBX que está descrito como usuário 1 e o usuário chamado tem seu PBX, usuário 2. O contabilizador faz a comunicação entre os usuários, depois que ele encontra um usuário disponível para intermediar a chamada, ela é então encaminhada via PSTN até o usuário chamado.

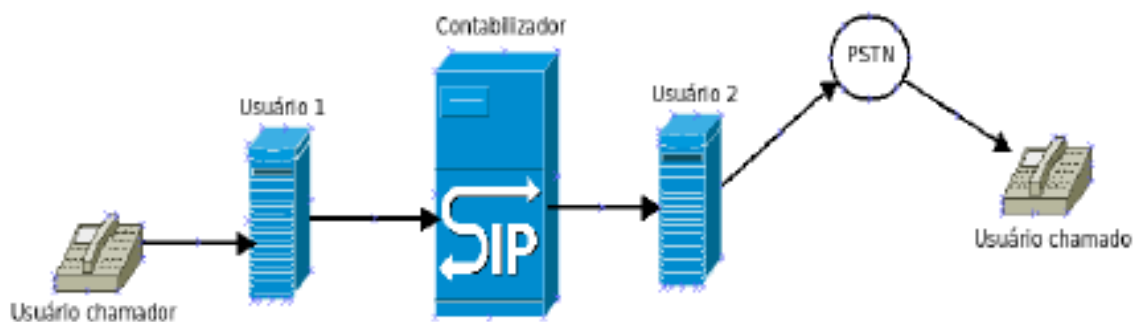


Figura 3.1: Componentes do sistema

Para o usuário participar dessa rede, é necessário que ele se cadastre no contabilizador. Nesse cadastro é informado o número de seu telefone com DDD, os minutos disponíveis mensais que irá ceder de sua rede de telefonia, além de sua credencial de acesso. A quantidade de minutos mensais que um usuário pode usar o sistema, denominada crédito, é limitada pelos minutos disponibilizados mensalmente por esse usuário. Quando um usuário quiser fazer uma chamada interurbana, será feito de seu PBX para o contabilizador que então encaminha para o PBX do usuário que se encontra na mesma localidade. A partir daí encaminha-se a chamada via PSTN, fazendo com que o custo seja o mesmo de uma ligação local. Se a pessoa com quem o usuário deseja conversar tiver um PBX e fizer parte dessa rede, a chamada então é toda via VoIP e o custo é zero.

O modelo da infraestrutura utilizada é o centralizado, sendo que o contabilizador tem as informações de usuário, como localização, crédito e minutos disponíveis.

As configurações do contabilizador foram feitas com o FreeSwitch, pois é necessário que o sistema suporte um número elevado de chamadas simultâneas. Os usuários participantes foram implementados com configurações do software Asterisk.

A comunicação entre o contabilizador e os usuários é feita através do protocolo SIP. Sempre que uma chamada é iniciada, é enviada uma mensagem SIP com informações do usuário chamador e do usuário destino, tais como nome do usuário chamador e número de destino. As streams de áudio são encaminhadas por RTP.

Cada participante pode fazer chamadas a partir de softphone, telefone IP, celulares ou telefones convencionais associados a ATAs (Adaptadores para Telefone Analógico).

Através da figura 3.2 é possível verificar os usuários cadastrados no contabilizador. A comunicação entre o contabilizador e os usuários é sempre via VoIP. Depois que a chamada chegar ao PBX do usuário com mesmo DDD de destino, então segue via PSTN até o número

chamado.

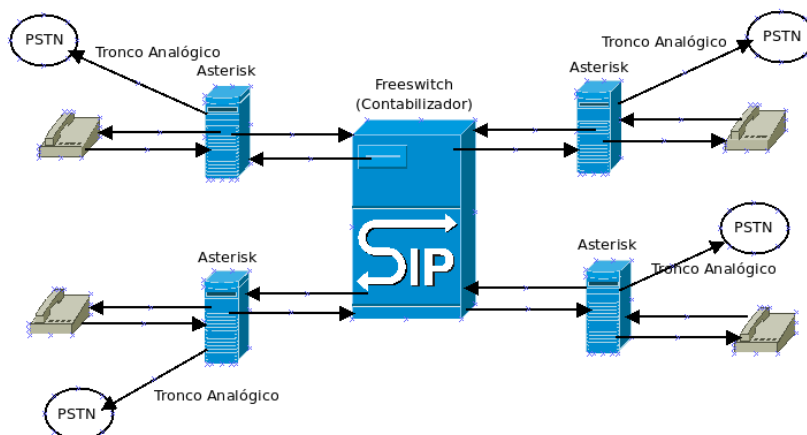


Figura 3.2: Diagrama do sistema

### 3.1.1 Encaminhamento de chamadas

Encaminhamento de chamadas é o procedimento realizado desde o usuário chamador, passando pelo contabilizador, até o usuário chamado, para a chamada então ser completada.

Quando um usuário quiser efetuar uma chamada, ele discar o número desejado em seu aparelho telefônico (softphone, telefone analógico, etc.), que o encaminhará para o PBX do usuário. O PBX, por sua vez, verifica se o número chamado é local ou interurbano. No primeiro caso a chamada é realizada diretamente pela linha telefônica do usuário. No segundo caso, o PBX usa o protocolo SIP para encaminhá-la pela rede de dados para o contabilizador. Desta forma, o contabilizador deve agir como um proxy SIP, tendo o papel de encontrar um usuário na área de destino do número chamado e encaminhar-lhe a chamada.

O contabilizador verifica em seu banco de dados se esse usuário chamador possui crédito. Se o usuário possuir crédito suficiente para iniciar uma chamada, então é realizada uma outra pesquisa no banco de dados para poder verificar o usuário que irá intermediar a chamada. Essa pesquisa é feita através do DDD do usuário destino.

O contabilizador procura um usuário no DDD destino, o qual deve ter minutos disponíveis para receber ligações. Após encontrá-lo, o contabilizador agenda o término da chamada, pois a duração da chamada não pode passar do valor de crédito do usuário chamador, nem do valor de minutos disponíveis do usuário intermediador, portanto o contabilizador verifica o menor valor entre essas opções e agenda o encerramento. Após agendar o encerramento, o contabilizador encaminha uma mensagem SIP (INVITE) ao usuário no DDD de destino para dar continuidade

à chamada. Quando a chamada chega no segundo usuário, é então encaminhada via PSTN. O usuário que irá pagar a conta da Operadora será o que intermediou, porém ele terá esse valor da duração da chamada convertido em crédito para poder efetuar ligações. A intermediação do contabilizador se limita à sinalização SIP. O transporte do áudio (RTP) se faz diretamente entre os PBX dos usuários.

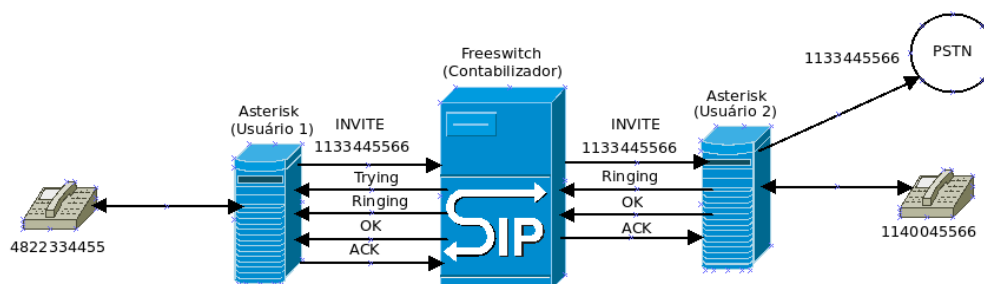


Figura 3.3: Encaminhamento de chamadas

Conforme mostra figura 3.3 o usuário 1 com número 4822334455 gostaria de encaminhar uma chamada para o destino 1133445566. Primeiramente, o PBX do usuário 1 irá verificar se o número é DDD, caso seja é encaminhado um INVITE para o contabilizador. No INVITE vão as informações do usuário chamador junto com o número desejado para destino. O contabilizador então verifica se o usuário chamador possui crédito suficiente para encaminhar a chamada, caso possua, ele busca em seu banco de dados um usuário disponível no DDD 11 e então encaminha para esse usuário um INVITE para iniciar a chamada.

Após o usuário aceitar a chamada, em seu plano de discagem é verificado se a ligação se destina a seu próprio número. Caso seja, então ela deve ser encaminhada aos aparelhos telefônicos do usuário. Caso contrário, a chamada deve ser encaminhada via PSTN para o número de destino.

Quando a chamada for encerrada são atualizados os valores de créditos e minutos disponíveis dos usuários participantes dessa chamada.

### 3.1.2 Mecanismo de Crédito

Quando um usuário se registra na rede, é necessário informar sua identidade, localidade e o valor em minutos mensais que irá ceder de sua linha telefônica local. Esse valor é chamado de minutos disponíveis mensais e é convertido de forma igual em crédito e minutos disponíveis. Por exemplo, se o usuário deseja ceder 40 minutos de seu telefone por mês, então todo mês é atualizado o valor de crédito e minutos disponíveis para 40.



Crédito é o valor em minutos que o usuário poderá utilizar para efetuar chamadas. Minutos disponíveis é o valor em minutos que o usuário cederá de sua rede para receber ou intermediar chamadas.

O usuário que está efetuando a ligação precisa ter crédito disponível, e o usuário que irá intermediar precisará ter minutos disponíveis.

Ao final da chamada, o valor de crédito do usuário que efetuou a chamada é debitado conforme a duração da chamada, da mesma forma que o valor de minutos disponíveis do usuário que cedeu sua linha de telefone é debitado. Quando um usuário cede sua linha de telefone, o valor da chamada em minutos é convertido para ele em crédito. Essa informação é independente dos minutos disponíveis mensais, para dessa forma premiar o usuário que mais ceder tempo de sua linha telefônica.

Com a figura 3.4 é possível verificar o valor de crédito e minutos disponíveis dos dois usuários participantes da chamada. Essa figura ilustra a troca de mensagens realizadas durante um encaminhamento de chamada. Ao final da chamada é possível verificar os valores atualizados conforme mostra figura 3.5, é claro que a chamada foi encerrada por causa da mensagem SIP BYE.

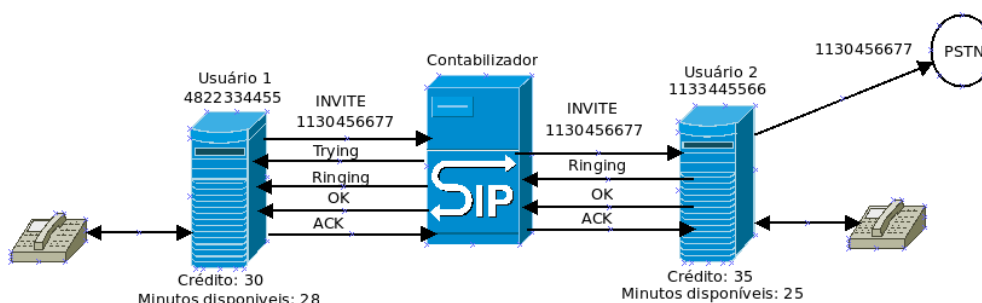


Figura 3.4: Exemplo de valores de crédito e minutos disponíveis dos dois usuários participantes no início da chamada

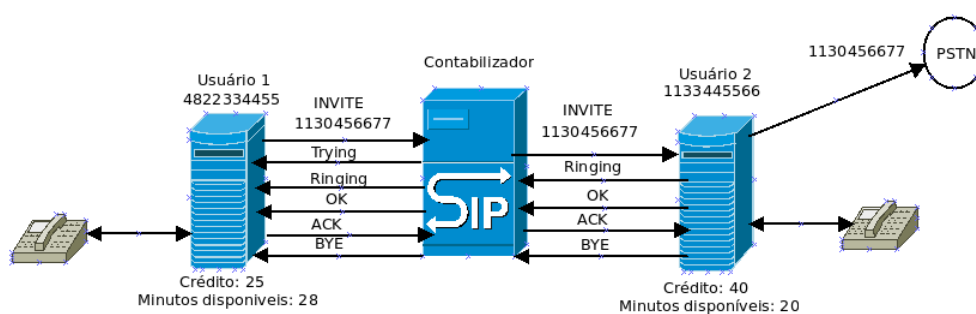


Figura 3.5: Exemplo de valores de crédito e minutos disponíveis atualizados ao fim da chamada, supondo que a ligação durou 5 minutos

## 3.2 Protótipo do Sistema

Um protótipo do sistema foi desenvolvido, contendo o contabilizador e seus usuários participantes. O contabilizador é o coração do sistema, pois ele é responsável pelo encaminhamento das chamadas entre os usuários. Com esse modelo inicial do sistema, foram feitas as devidas implementações para o correto funcionamento.

- **Contabilizador:** Possui as informações dos usuários participantes como localidade, crédito e minutos disponíveis e é o responsável pelo encaminhamento das chamadas, como também pela atualização de créditos.
- **Usuários participantes:** Para participar da rede o usuário precisa se registrar no contabilizador e compartilhar sua rede de telefonia PSTN.

### 3.2.1 Contabilizador

O Contabilizador funciona como um proxy SIP, fazendo a intermediação das chamadas entre os usuários participantes. É nele que ficam armazenadas as informações de todos os usuários participantes da rede, como número, DDD, valores de crédito e minutos disponíveis e ainda o valor em minutos que o usuário irá ceder de seu telefone convencional por mês.

Ele foi implementado com Freeswitch, esse software permitiu que todas as configurações de autenticação de usuários e plano de discagem fossem delegadas para um programa externo, facilitando a implementação do projeto.

No script de autenticação de usuários foi necessário importar informações do Freeswitch e do banco de dados. As informações dos usuários ficam armazenadas em um banco de dados.

A configuração do contabilizador é dividida entre autenticação de usuários e plano de discagem.

- Autenticação de usuários: Quando um usuário deseja se registrar na rede, ele manda uma mensagem SIP REGISTER para o contabilizador com as informações de número de usuário e endereço IP para poder ser contactado.
- Plano de discagem: Onde contém toda a configuração e condições de encaminhamento de chamadas.

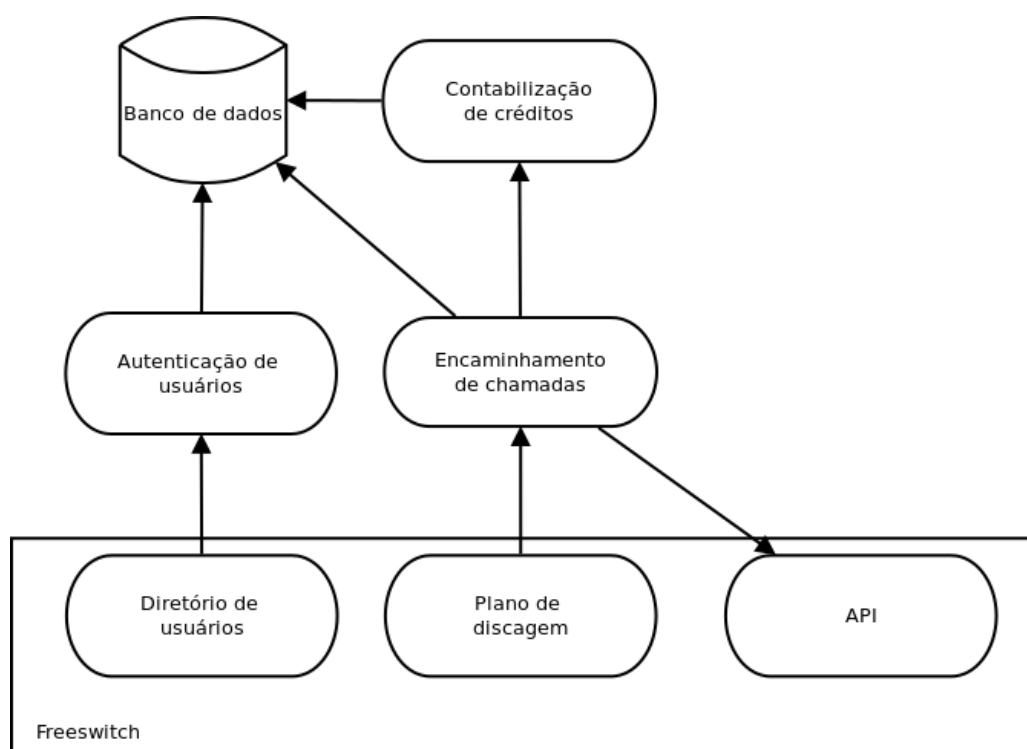


Figura 3.6: Arquitetura do Contabilizador

Conforme ilustra figura 3.6, a contabilização de créditos e encaminhamento de chamadas fazem parte do plano de discagem, já a autenticação de usuários faz parte do diretório de usuários. Para a chamada ser efetuada é necessário que tenha acesso à API do Freeswitch, o acesso ao banco de dados também é requisitado, como para a contabilização de créditos e autenticação de usuários, para acessarem as informações dos usuários cadastrados.

### Autenticação de Usuários

A autenticação de usuários envolve a verificação das credenciais desses usuários quando se registram no contabilizador. O registro se faz com a mensagem SIP REGISTER para fornecer

seus dados como número de usuário e endereço IP. Com essas informações o contabilizador pode identificar que o usuário está ativo e pode receber chamadas.

No banco de dados do contabilizador contém as informações de número e senha de usuários. Quando o usuário quiser se registrar, é verificado se os dados informados estão de acordo com o banco de dados.

Dentro do script de autenticação de usuários é feita a comparação dos dados fornecidos com as informações contidas no banco de dados. Se o número de usuário e senha forem corretos então o usuário se registra na rede, caso contrário, não irá conseguir se autenticar.

Sqlite > SELECT * FROM users:						
user	password	ddd	creditos	minutos_disponiveis	minutos_mensais	data_recarga
4822334455	1234	48	30	37	40	1
4833444408	1234	48	30	10	50	13
4833445566	1234	11	40	15	50	22
1140043344	1234	11	40	15	45	12

Tabela 3.1: Exemplo de usuários na base de dados sqlite

### Plano de discagem

O plano de discagem contém toda a lógica de encaminhamento de chamadas.

Depois que o usuário se autentica, quando ele quiser efetuar uma chamada, será feita uma análise no plano de discagem para verificar se a chamada poderá ser completada. Na primeira parte do plano de discagem é feita a verificação se o usuário chamador possui crédito suficiente para a chamada ser completada.

Após verificar se usuário chamador possui crédito, então o contabilizador analisa os dois primeiros dígitos do usuário destino para poder verificar o DDD e então encontrar um usuário nessa localidade para intermediar a chamada. O critério para escolher o usuário na localidade de destino é a maior quantidade de minutos disponíveis.

Quando o usuário intermediador da chamada for selecionado, então a chamada é encaminhada.

A chamada termina quando um dos usuários encerram ou quando acaba o crédito do usuário chamador ou ainda quando acabam os minutos disponíveis do usuário intermediador.

Após o término da chamada são atualizados os valores de crédito e minutos disponíveis de ambos os participantes.

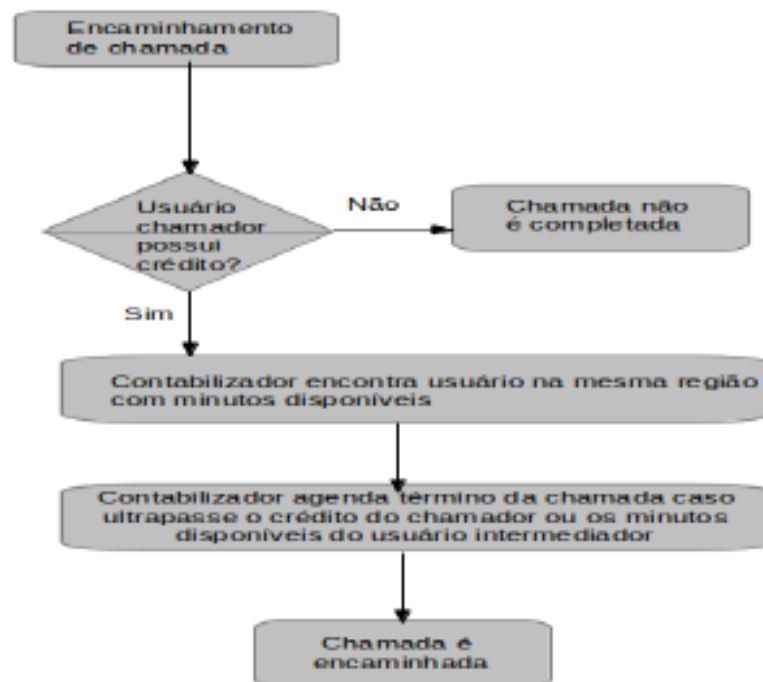


Figura 3.7: Fluxograma do plano de discagem do contabilizador

### Contabilização de Créditos

Após a chamada ser encerrada é feita a atualização de crédito e minutos disponíveis dos dois usuários participantes. Para fazer essa atualização, foi feita uma função dentro do plano de discagem.

O valor é atualizado de acordo com o tempo da chamada, cada unidade de crédito e minuto disponível corresponde a um minuto.

Quando a chamada é encerrada, primeiro é verificado o tempo total de duração dessa chamada. Após verificar o tempo, é então atualizado os valores de crédito e minutos disponíveis de ambos usuários, como demonstra a figura 3.8. Assim que é feita essa atualização, os novos valores são gravados no banco de dados.

Cada usuário quando se registra na rede tem uma data para recarga de crédito e minutos disponíveis, o valor dessa recarga depende de quanto tempo esse usuário irá ceder da sua rede para os demais participantes a utilizarem. O valor que o usuário irá ceder de sua rede é estabelecido assim que é feito o registro na rede.

Para fazer essa atualização foi criado um script comparando a data atual com a data da recarga, essa data é estabelecida pelo usuário. Se as datas corresponderem, então o valor de minutos mensais é convertido para crédito e minutos disponíveis. Os valores não são cumulativos,

ou seja, se o usuário ceder 30 minutos de sua rede por mês e no dia da recarga ainda tiver 10 minutos de crédito, o valor será atualizado para 30 e não 40 minutos.

Todo dia é feita a verificação de atualização, pois cada usuário tem uma data de recarga diferente dos demais, a figura 3.9 ilustra a atualização mensal.

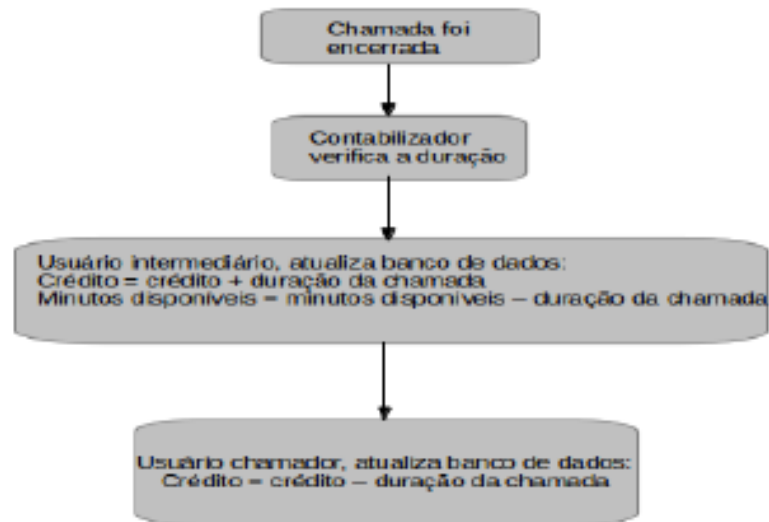


Figura 3.8: Fluxograma do encerramento da chamada

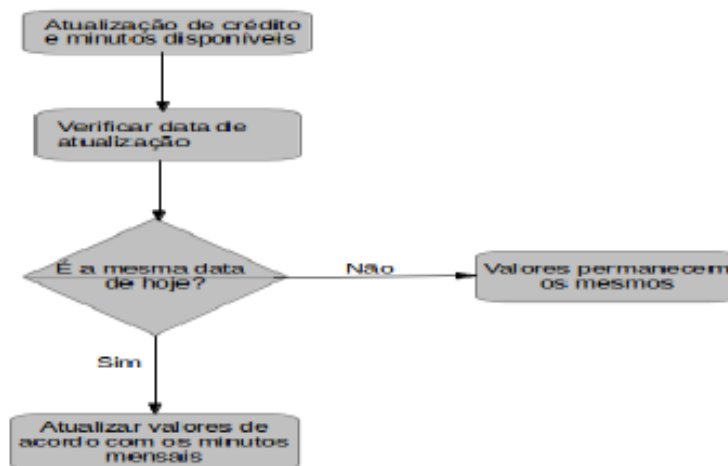


Figura 3.9: Fluxograma da atualização mensal de crédito e minutos disponíveis

### 3.2.2 A Infraestrutura do Participante

O PBX do usuário participante precisa ter configurações de autenticação, onde é informado o número de usuário junto com a senha de acesso. Essas informações são confirmadas pelo contabilizador na hora do registro. No plano de discagem é feita a configuração de encaminhamento de chamadas. O PBX também terá o papel de intermediar uma chamada entre outro usuário, para isso, o usuário irá compartilhar sua rede de telefonia PSTN

Para o usuário compartilhar sua rede de telefonia PSTN, precisa de uma interface FXO para fazer a conversão da linha digital para analógica (no PABX é digital e via PSTN é analógica), ter um PABX configurado e um telefone para efetuar e intermediar chamadas, podendo ser um telefone analógico ou um softphone. Se for um telefone analógico então a chamada é encaminhada por uma interface FXS que converte o áudio digital para analógico.

As funcionalidades do participante são divididas em duas partes:

- Registro: O participante precisa fazer o registro no contabilizador informando seu número de usuário e senha.
- Plano de discagem: Onde estão todas as configurações de encaminhamento de chamadas

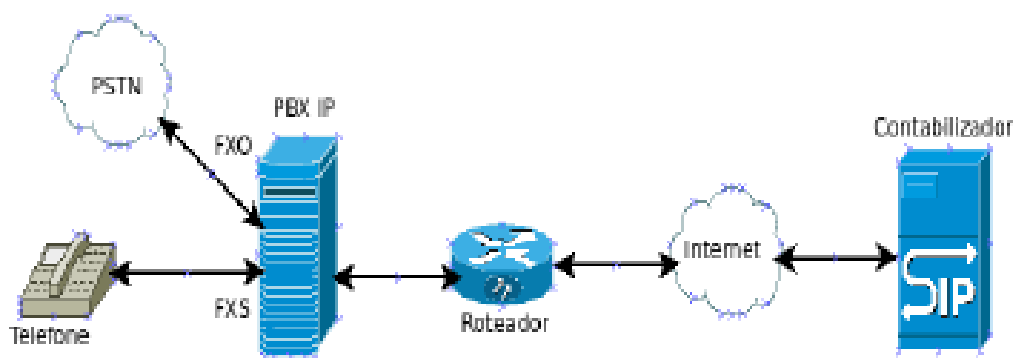


Figura 3.10: A Infraestrutura do participante

## Registro

O participante deve se autenticar na rede para poder ser visível para o contabilizador. Para isso ser possível, ele manda uma mensagem *register*, dessa forma o contabilizador verifica em seu banco de dados se o usuário que está tentando se registrar está cadastrado.

Conforme mostra figura 3.11, o usuário informa seu número e senha (número:senha@IP\_contabilizador). Com esses dados ele se registra no contabilizador e então é verificado se as informações estão corretas. Se o contabilizador verificar que as informações de número e senha de usuário estão corretas, então o registro é completado, caso contrário a autenticação irá falhar.

```
[ general ]
insecure=port , invite
allow=all
qualify=yes
register=>1133445566:1234@192.168.56.1
```

Figura 3.11: Registro do usuário participante na rede

### Plano de discagem

O plano de discagem é onde que estão todas as condições para um usuário efetuar e receber chamada.

Quando o participante quiser efetuar uma chamada, será verificado se é DDD ou local. Se for DDD chamada é encaminhada para o contabilizador, se for ligação local vai pela rede de telefonia PSTN através da porta FXO.

Para receber ligação foram criados três contextos no plano de discagem, pois existem três possíveis formas de receber chamadas:

- Se a chamada vier via PSTN: vai ser encaminhada para o próprio telefone do usuário.
- Se a chamada vier via VoIP: primeiro será verificado se é para o próprio participante (ramal SIP, por exemplo), se for, encaminha para ele mesmo, caso contrário verificar se é local, se for, encaminhar via PSTN.
- Se a chamada vier do próprio telefone: é verificado se é DDD, se for, encaminha via VoIP, caso contrário, encaminha via PSTN. Se o envio via VoIP falhar, encaminha via PSTN.

Quando o usuário recebe ligação para seu próprio número, tem duas maneiras de serem feitos o encaminhamento. Se ele tiver um ramal SIP cadastrado então as ligação irão direto para ele. Caso tenha um telefone analógico, ao receber chamadas, será encaminhado através da porta FXS.



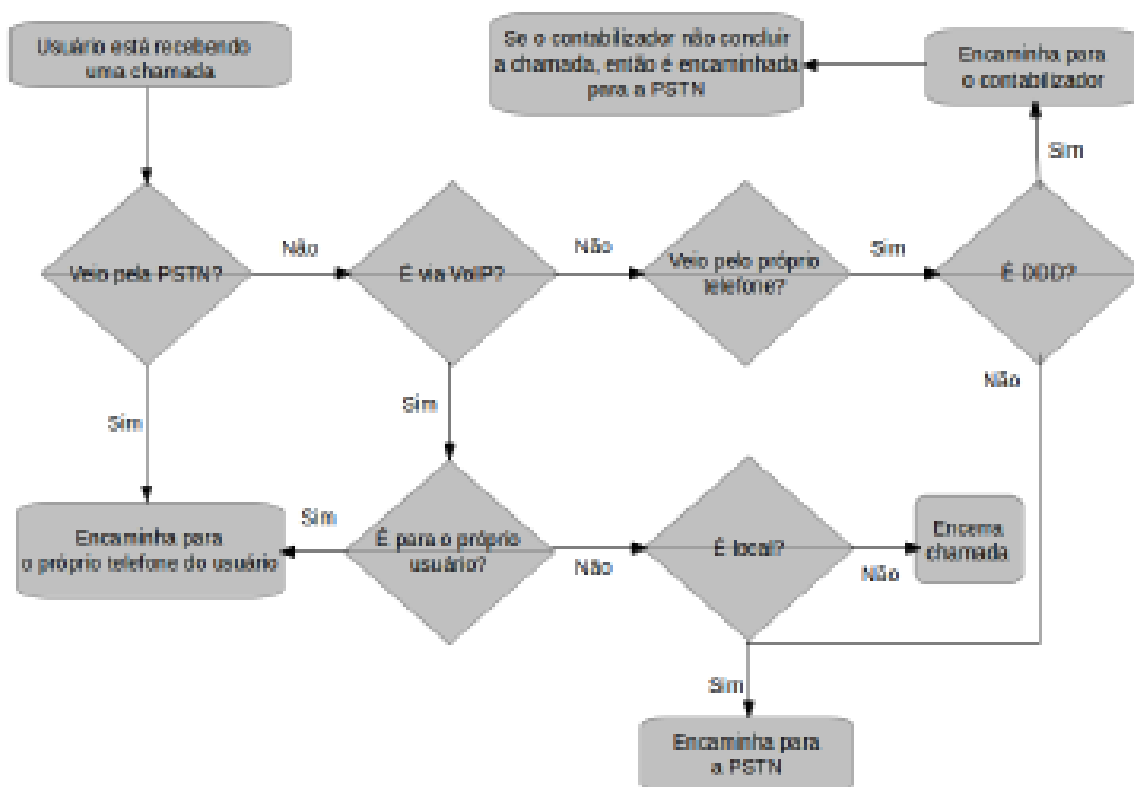


Figura 3.12: Fluxograma do encaminhamento das chamadas

## 4 *Implantação do Projeto e testes realizados*

Nesse capítulo descreve-se a implementação do projeto, incluindo a forma como o protótipo foi configurado e os testes realizados. As configurações estão divididas entre usuário participante e contabilizador.

### 4.1 Usuário participante

Os usuários participantes foram configurados no Asterisk. Ele se divide em registro de usuário e plano de discagem.

#### Registro de usuário

O registro de usuário foi configurado dentro de */etc/asterisk/sip.conf*. Essa configuração informa as credenciais do usuário para poder se registrar no Contabilizador. É dentro desse arquivo, também, onde se encontram as configurações de contextos, é onde eles são chamados para serem executados no plano de discagem. Como mostra a figura 4.1, o usuário está se registrando no Contabilizador e informa também o contexto a qual ele está vinculado, que no caso é *troncoSip*. Ou seja, sempre que vier uma chamada daquele IP do contabilizador, a chamada será encaminhada para o contexto *troncoSip* do plano de discagem.

```
[ general ]
insecure=port , invite
allow=all
qualify=yes
register=>1133445566:1234@192.168.56.1
context=troncoSip
```

Figura 4.1: Informações do usuário para se registrar no contabilizador

## Plano de Discagem

O plano de discagem foi configurado em `/etc/asterisk/extensions.conf`. É onde mostra os possíveis caminhos de origem da ligação, o plano de discagem então analisa por onde veio para saber exatamente para onde encaminhar. A figura 4.2 mostra o contexto `troncoSip`, que contém as regras para processamento de chamadas vindas pelo contabilizador. A primeira regra encaminha chamadas pela PSTN, caso o número de destino seja local. As regras seguintes encaminham a chamada para o ramal SIP ou ramal analógico via interface FXS, caso a chamada se destine ao próprio usuário.

```
[tronco SIP]
;chamadas vindas do contabilizador
;ligacao para um numero da minha cidade sai pela pstn
;exemplo de um dispositivo que prove portas FXO, b0c64 no modulo Khomp
exten=>_11X.,1,Dial(Khomp/b0c64/${EXTEN})
same=>n, Hangup

;ligacao do contabilizador para meu numero vai direto para meu ramal
SIP
exten=>1133445566,1,Dial(SIP/2002)
same=>n, Hangup

;se o ramal for um telefone analogico (porta FXS)
exten=>1133445566,1,Dial(Khomp/b0c60/${EXTEN})
same=>n, Hangup
```

Figura 4.2: Contexto `troncoSip` implementado no plano de discagem de um usuário

## 4.2 Contabilizador

O contabilizador foi implementado no Freeswitch. As configurações de autenticação de usuário e plano de discagem foram delegadas a um programa externo. No contabilizador se encontra o banco de dados com as informações dos usuários. Dentro do plano de discagem do contabilizador tem as informações de encaminhamento de chamadas e de atualização de crédito.

### Autenticação de usuários

A autenticação de usuários foi implementada em um script em Python. Esse script verifica as credenciais do usuário a fim de aceitar ou recusar seu registro. O caminho para o ar-

quivo que contém esse script foi configurado em */freeswitch/conf/autoload\_configs* no arquivo *python.conf.xml* como mostra figura 4.3.

```
<configuration name="python.conf" description="PYTHON Configuration">
  <settings>
    <param name="xml-handler-script" value="teste"/>
    <param name="xml-handler-bindings" value="directory"/>
  </settings>
</configuration>
```

Figura 4.3: Informação do nome do programa externo com configuração de autenticação de usuários

### Banco de dados

O banco de dados utilizado para implementação foi o *sqlite*. Esse banco de dados possibilita criar bases de dados em arquivos, e acessá-las diretamente com consultas SQL. Dentro do *sqlite* foi criada uma tabela *users* com as informações dos usuários, como mostra a figura 4.4. Essa tabela foi criada dentro do arquivo “meususuarios.db”.

```
Sqlite > CREATE TABLE users(id integer primary key, user varchar(16),
  password varchar(16), credito integer, cod_area integer, cidade
  varchar(64), minutos_disponiveis integer, minutos_mensais integer,
  ultima_recarga integer);
```

Figura 4.4: Como foi criada a tabela no *sqlite* com as informações de usuários

### Plano de discagem

O plano de discagem foi implementado em um script em Python. Como foi configurado em um programa externo, é necessário informar o nome desse arquivo. Essa informação consta dentro de */freeswitch/conf/dialplan* no arquivo *default.xml* como mostra figura 4.5. Nas informações de número de destino ele informa que é qualquer valor entre 0 e 9, que seleciona qualquer sequência de dígitos numéricos.

```
<extension name="nicole">
  <condition field="destination_number" expression="^[0-9]+">
    <action application="python" data="discagem"/>
  </condition>
</extension>
```

Figura 4.5: Informação do nome de arquivo do plano de discagem

### **Encaminhamento de chamadas**

Para encaminhar uma chamada, o contabilizador procura um outro usuário disponível para intermediar essa chamada. Essa verificação é feita através de uma função criada dentro do plano de discagem, onde ele verifica através do DDD um usuário na mesma localidade e que tenha minutos disponíveis maior que zero. Se tiver mais de um usuário na mesma localidade com minutos disponíveis, o contabilizador escolhe o que tem o maior valor de minutos disponíveis.

### **Contabilização de Crédito**

A atualização de créditos é feita de duas formas: ao final da chamada e mensalmente. Para o crédito e minutos disponíveis serem atualizados ao final da chamada, foi criada uma função dentro do plano de discagem. Essa função foi programada para ser executada somente ao final da chamada, é verificado a duração da chamada e com essa informação, os valores são atualizados e salvos no banco de dados, como mostra figura 4.6.

```

def hangup_hook(session, what, args=''):
    f = open('/tmp/hangup.txt', 'a')
    try:
        # pegando conteudo da variavel de inicio – tempo em micro segundos
        inicio = int(session.getVariable('answered_time'))

        # pegando conteudo da variavel de fim de chamada (tempo atual no
            computador em segundos)
        fim = int(time.time*1000000)

        # descobrindo duracao da chamada
        duracao = (fim - inicio)

        # duracao da chamada em minutos – arredonda valor para cima
        duracaochamada = math.ceil(duracao/60000000.0)
        cid_num = session.getVariable('caller_id_number')
        participante = session.getVariable('sip_req_user')
        db = sqlite3.connect("/home/nicole/scripts/meususuarios.db");
        c = db.cursor()

        # atualizacao dos valores de credito e minutos_disponiveis para os
            usuarios participantes da chamada
        c.execute('update users set credito=credito-%d where user="%s"' % (
            duracaochamada, cid_num))
        c.execute('update users set minutos_disponiveis=minutos_disponiveis-%
            d where user="%s"'%(duracaochamada, participante))
        c.execute('update users set credito=credito+%d where user = "%s"' % (
            duracaochamada, participante))

        # salva no banco de dados
        c = db.commit()

```

Figura 4.6: Atualização de créditos ao final da chamada

Para efetuar a atualização mensalmente, foi criado um script onde diariamente são verificadas as datas de atualização dos usuários, como demonstra figura 4.7. A data de recarga de cada usuário consta no banco de dados.

```
import sqlite3
import time
from datetime import date

#pega o valor da data de hoje
today = date.today()
dia = today.day
print dia
db = sqlite3.connect("/home/nicole/scripts/meususuarios.db");
c = db.cursor()

#compara a data de hoje com a data_recarga e atualiza os valores
# de credito e minutos_disponiveis conforme o valor de minutos_mensais
c.execute('update users set credito=minutos_mensais where data_recarga
        =%d' %dia)
c.execute('update users set minutos_disponiveis=minutos_mensais where
        data_recarga=%d' %dia)
db.commit()
db.close()
```

Figura 4.7: Atualização de crédito mensalmente

## 5 *Conclusões*

Este trabalho propôs o desenvolvimento de uma rede colaborativa com VoIP em que usuários participantes puderam efetuar ligação DDD com custo de ligação local, sendo ainda que a rede premia os usuários que mais cedem tempo de sua rede de telefonia local.

O contabilizador conseguiu verificar informações de crédito e minutos disponíveis dos usuários para poder concluir a chamada através das configurações dos planos de discagem.

Foi possível fazer a atualização de crédito e minutos disponíveis ao final da chamada. Uma função foi programada no plano de discagem do contabilizador para ser executada após o término da chamada, através dessa função foi permitido verificar a duração e então, atualizar os valores.

A chamada foi encaminhada de forma correta para o destino correspondente. Depois que a chamada chegou ao PBX do usuário intermediador, seu plano de discagem efetuou o encaminhamento para o destino final.

A atualização de créditos é realizada mensalmente. Foi criado um agendamento para que diariamente se verifiquem as atualizações de créditos de usuários. No banco de dados, cada usuário tem uma data de atualização de valores.

Como trabalho futuro, é possível expandir essa rede criando vários contabilizadores. Com a criação de vários contabilizadores mais usuários podem participar da rede. Para isso ser possível, os contabilizadores teriam que se comunicar para poderem encaminhar chamadas.

É possível melhorar a forma de mecanismo de crédito em trabalhos futuros, para que não haja fraudes, obrigando os usuários a respeitarem esse mecanismo de créditos. Os usuários podem não compartilhar seu PBX já que o crédito será atualizado todo mês, fazendo com que eles rejeitem uma chamada entrante, ou então podem aceitar e logo encerrar para assim ganharem um crédito para efetuarem novas chamadas.



## *Referências Bibliográficas*

- ADMIN. *A Comparison of VOIP Platforms: Asterisk vs. FreeSWITCH*. 2013. <http://gonorthforge.com/a-comparison-of-voip-platforms-asterisk-vs-freeswitch/>.
- ALECRIM, E. *Usando cron e crontab para agendar tarefas*. 2005. <http://www.vivaolinux.com.br/artigo/Usando-cron-e-crontab-para-agendar-tarefas?pagina=1>.
- ARRUDA, F. *Cron: facilite o agendamento de tarefas no Linux*. 2014. <http://canaltech.com.br/tutorial/linux/cron-facilite-o-agendamento-de-tarefas-no-linux/>.
- BROWNORTH, A. *Asterisk vs. FreeSWITCH*. 2008. <http://anders.com/cms/266>.
- COLCHER, S. et al. *VoIP – Voz sobre IP*. [S.l.]: Elsevier Editora Ltda, 2005.
- GUERRA, L. *Data e hora do sistema*. 2012. <http://www.vivaolinux.com.br/dica/Data-e-hora-do-sistema>.
- HANDLEY, M. et al. *SIP: Session Initiation Protocol*. 1999. <http://www.ietf.org/rfc/rfc2543.txt>.
- HARDY, W. *VoIP Service Quality: Measuring and Evaluating Packet Switched Voice*. [S.l.]: McGraw Hill Professional, 2007.
- JERRIS, M. *The World's First Cross-Platform Scalable Free Multi-Protocol Soft Switch*. 2014. <https://freeswitch.org/>.
- JOHNSTON, A. B. *SIP - Understanding the Session Initiation Protocol*. [S.l.]: Artech House, 2009. ISBN 9871607839958.
- MADSEN, L.; SMITH, J.; MEGGELEN, J. V. *Asterisk, O futuro da telefonia*. [S.l.]: Alta Books, 2005.
- MELLO, E. *Voz sobre IP - VoIP*. 2011. <http://tele.sj.ifsc.edu.br/~msobral/rmu/slides/aula-22.pdf>.
- NELSON, J. *SIP For Next-Generation Mobile Services: Mobile IP and SIP*. 2002.
- OLIVER, H. *IP Telephony Packet-based Multimedia Communications Systems*. [S.l.]: Addison Wesley, 2005.
- OSLO. *Applying different types of mobility on one network*. 2002.
- PYMOTW. *datetime – Date/time value manipulation*. 2013. <https://pymotw.com/2/datetime/>.
- SCHULZRINNE, H. et al. *RTP*. 2003. <http://www.ietf.org/rfc/rfc3550.txt>.

SILVA, R. *Desvendando o comando DATE*. 2008. <http://www.vivaolinux.com.br/dica/Desvendando-o-comando-DATE>.

TAROUCO, L. et al. *Introdução ao Protocolo SIP*. [eng.registro.br/inoc/SIP\\_iNOC.pdf](http://eng.registro.br/inoc/SIP_iNOC.pdf).

TELECO. *Voz sobre IP I: Tecnologia e Protocolos*. 2015. [http://www.teleco.com.br/tutoriais/tutorialvoipconv/pagina\\_4.asp](http://www.teleco.com.br/tutoriais/tutorialvoipconv/pagina_4.asp). Último acesso em 03 de abril de 2015.

VOCALTEC. *Expanding the Borders of VoIP*. [S.l.]: Israel, 2008.

WIKIPÉDIA. *Voz sobre IP — Wikipédia, A Enciclopédia Livre*. 2015. [https://pt.wikipedia.org/wiki/Voz\\_sobre\\_IP](https://pt.wikipedia.org/wiki/Voz_sobre_IP). Último acesso em 20 de junho de 2015.