Uma arquitectura IPtel baseada no protocolo SIP

João Paulo Sousa
Instituto Politécnico de Bragança
R. João Maria Sarmento Pimentel,
5370-326 Mirandela,
Portugal + 351 27 820 13 40
jpaulo@ipb.pt

Eurico Carrapatoso
FEUP/INESC Porto
R. Dr. Roberto Frias,
4200-465 Porto,
Portugal + 351 22 209 42 98
emc@fe.up.pt

RESUMO

Mais e melhores acessos à Internet, bem como um preço atractivo, fazem com que seja cada vez maior o interesse em transportar voz e vídeo sobre redes de dados. Neste contexto nasceu a Telefonia sobre IP, que oferece através desta infra-estrutura a oportunidade de criar sistemas globais de comunicação multimédia.

O *Session Initiation Protocol*, utilizado nesta arquitectura, é um protocolo de sinalização e controlo de chamadas entre dois ou mais participantes.

Neste artigo é feita uma apresentação do serviço de Telefonia sobre IP, são analisados diversos protocolos tipicamente usados na arquitectura protocolar da IPtel, e é explicada a arquitectura e o funcionamento do protocolo SIP. São também referidos vários modos de mobilidade disponibilizados pelo SIP ao nível da aplicação. Finalmente são apresentadas algumas características do sIPtel, um sistema IPtel desenvolvido em Java com suporte para vídeo e que utiliza o protocolo SIP na sinalização de chamadas.

Palavras-chave

Telefonia sobre IP, IPtel, SIP, Protocolos, Sinalização, Codificadores de áudio, Codificadores de vídeo.

1. INTRODUÇÃO

Durante anos, companhias e organizações utilizaram um conjunto limitado de serviços de comunicação, como o telefone e o fax, suportados pela rede telefónica tradicional e uma rede comutada de pacotes para o transporte de dados. Actualmente têm a oportunidade de utilizar a rede IP como única infra-estrutura para as comunicações, permitindo a integração de novos serviços que possibilitam novas formas de comunicação e de condução de negócios, e a redução de custos, tornando-as mais competitivas.

Ao contrário da telefonia tradicional, as comunicações IP permitem, para além do transporte de voz, a integração de vídeo, dados e de novos serviços como *chat*, mensagens instantâneas e *Web*, numa única infra-estrutura e num único serviço, o IPtel. Um dos exemplos mais comuns da integração da Web com as chamadas de voz e/ou vídeo é o serviço "*click to dia*l", em que por exemplo, é iniciada uma chamada telefónica para o serviço de apoio ao cliente durante uma compra de um produto num site electrónico, para o esclarecimento de uma dúvida.

Actualmente existem dois protocolos para as comunicações em tempo real: o *Session Initiation Protocol* (SIP) da *Internet Engineering Task Force* (IETF) e o H.323 da *International Telecommunications Union* (ITU). Estes dois protocolos são utilizados para encaminhamento, sinalização e controlo de chamadas e outros serviços suplementares. O H.323 é um protocolo já estabelecido e largamente utilizado devido principalmente a ter dado provas da sua capacidade e à interoperabilidade com a rede telefónica pública comutada. O SIP é um protocolo recente, que se integra facilmente em aplicações de Internet, devido à semelhança com protocolos da *Web* e correio electrónico, e que promete escalabilidade, flexibilidade e facilidade na criação de serviços.

Desenvolvido pelo grupo MMUSIC do IETF, o SIP, foi inicialmente publicado na RFC 2543 em 1996 e tornada obsoleta com a publicação da RFC 3261 em Junho de 2002. É um protocolo de controlo (sinalização) ao nível da aplicação para a criação, alteração e finalização de sessões entre um ou mais intervenientes. Estas sessões incluem chamadas de Telefonia sobre IP, distribuição e conferência multimédia. O SIP ou extensões do SIP podem ainda ser usadas para mensagens instantâneas, notificação de presença, e jogos distribuídos. Após o grande sucesso do SIP, o IETF decidiu criar o *SIP Working Group*, um grupo independente para continuar desenvolvimento deste protocolo iniciado pelo MMUSIC.

Este artigo inicia-se com uma descrição da arquitectura IPtel. Em seguida é feita uma abordagem ao protocolo SIP e é apresentada a especificação de um serviço IPtel. São também descritos três modos de mobilidade, terminal, sessão e pessoal. Finalmente é apresentado o sIPtel, um sistema IPtel que foi desenvolvido e que permite a comunicação entre dois participantes em tempo real com áudio e vídeo, utilizando o protocolo de sinalização SIP para o estabelecimento das sessões.

2. ARQUITECTURA DA IPtel

A principal diferença entre a IPtel e a rede telefónica tradicional (PSTN) manifesta-se na arquitectura de comutação: enquanto a PSTN é uma rede de comutação de circuitos, a rede IP é uma rede de comutação de pacotes. Esta particularidade permite que numa rede IP dois dispositivos troquem diferentes tipos de informação sem necessitarem de estar directamente conectados e sem reserva de recursos, sendo características de localização e encaminhamento da responsabilidade dos protocolos. O

mesmo não se passa numa rede PSTN, onde é estabelecido um circuito físico entre os dois dispositivos, reservando um canal *full-duplex* para cada sessão de conversação, independentemente da existência ou não de tráfego de voz.

A Figura 2.1 mostra os três tipos de dispositivos que Lennox e Schulzrinne [1] identificaram numa rede IPtel: terminais, *gateways* e servidores de sinalização. Os terminais permitem executar os serviços, como por exemplo fazer e receber chamadas. Estes dispositivos terminais na rede IP são entendidos como dispositivos inteligentes possuindo total controlo sobre o estado da chamada, ao contrário dos telefones tradicionais que apenas reagem a comandos de uma central controladora, reflectindo uma arquitectura mestre-escravo.

As *gateways* são dispositivos opcionais numa rede IPtel, sendo apenas usadas quando existe necessidade de interligar duas redes que não usem a mesma tecnologia de comunicação. As *gateways* têm então a função de tradução entre terminais com diferentes formatos de transmissão, localização ou procedimentos de comunicação, e codificação de meios.

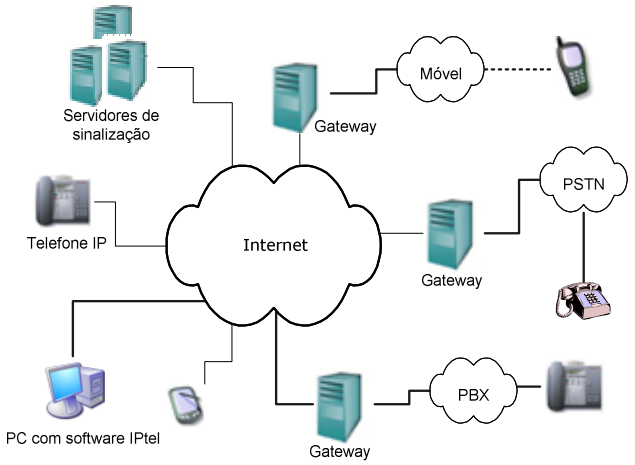

Figura 2.1 - Componentes de uma arquitectura IPtel

Os servidores de sinalização funcionam ao nível da aplicação, controlando o encaminhamento das mensagens de sinalização. Estes servidores disponibilizam serviços de localização do utilizador, mantendo informação sobre onde pode ser encontrado de modo a encaminhar os pedidos de sinalização para a localização actual do utilizador.

3. DESCRIÇÃO DO PROTOCOLO SIP

O SIP é caracterizado por ser um protocolo simples que utiliza um conjunto de protocolos que providenciam diferentes serviços e que permitem implementar uma arquitectura multimédia. O *Real Time Transport Protocol* (RTP) para assegurar o transporte dos meios e o *Real Time Control Protocol* (RTCP) para fornecer informação útil ao nível de QoS. O *Session Description Protocol* (SDP) [2] para descrever a sessão multimédia. O DNS para a determinação do destinatário dos pedidos. O protocolo *Lightweight Directory Access Protocol* (LDAP) [3] para o acesso directo à base de dados de um servidor de localização. O *Telephony Routing over IP* (TRIP) [4] para troca de informação de encaminhamento entre domínios administrativos de telefonia. Finalmente o *Reservation Protocol* (RSVP) para estabelecer a reserva de recursos.

O SIP é baseado no *Simple Mail Transfer Protocol* (SMTP) [5] usado como protocolo base do serviço de *e-mail* e também no *HyperText Transfer Protocol* (HTTP), o protocolo base da *Web*. O SIP é um protocolo de texto e baseia-se no modelo cliente/servidor: o cliente faz pedidos e o servidor retorna respostas aos pedidos do cliente. Utiliza uma semântica e sintaxe semelhantes ao HTTP, e faz uso dos seus métodos de autenticação. Embora possa correr sobre o *Transmission Control Protocol* (TCP) e o *Stream Control Transmission Protocol* (SCTP), o SIP é mais utilizado sobre o *User Datagram Protocol* (UDP) [6], disponibilizando os seus próprios mecanismos de recuperação de erros e permitindo o envio de mensagens *multicast*.

Ao nível dos serviços, o SIP inclui na sua recomendação inicial para o estabelecimento e finalização de sessões, os seguintes serviços:

- Localização do utilizador: responsável pela localização do terminal para estabelecer a comunicação;
- Disponibilidade do utilizador: responsável pela determinação da vontade do utilizador em estabelecer uma sessão de comunicação;
- Recursos do utilizador: responsável pela determinação dos meios a utilizar e dos seus parâmetros;
- Características de negociação: responsável pela negociação e por chegar a acordo relativamente aos recursos disponíveis, porque nem todas as partes apresentam sempre o mesmo nível de recursos;
- Gestão da sessão: possibilidade de transferir, colocar em espera ou terminar sessões, assim como de modificar parâmetros das mesmas e de invocar serviços;
- Alteração das características da sessão: possibilidade de alterar as características da sessão no decurso da mesma.

O *software* SIP localizado no terminal e que interage com o utilizador é designado por *User Agent* (UA). O UA pode funcionar como *software* cliente num PC, num dispositivo móvel, ou como *firmware* num telefone IP. Um UA tem dois componentes: o *User Agent Client* (UAC) e o *User Agent Server* (UAS). O UAC é o responsável pela iniciação da chamada, enviando pedidos, os quais o UAS processa e aos quais responde enviando respostas.

4. ESPECIFICAÇÃO DE UM SERVIÇO IPtel

Na implementação de um serviço IPtel devem ser consideradas diversas propriedades ao nível da sinalização e da troca de meios. Assim a sinalização deve oferecer um conjunto de características como:

- Tradução de nomes e localização de utilizadores: esta propriedade pode ser implementada pelo terminal IPtel, recorrendo por exemplo ao serviço DNS. Esta opção

pode ser dispensada quando por omissão o terminal IPtel envia pedidos para um servidor SIP que implementa esse serviço;

- Capacidade de negociação: permite aos utilizadores que pretendam estabelecer uma sessão de comunicação acordarem o tipo de meios utilizados na sessão e os parâmetros da mesma;
- Controlo de chamadas: após o estabelecimento de uma chamada entre utilizadores, os participantes devem ter a possibilidade de, por exemplo, colocar utilizadores em espera e retomar novamente a sessão quando bem entenderem;
- Mudanças de características da sessão: durante uma chamada deve ser possível alterar características da sessão, por motivos opcionais dos utilizadores ou por necessidade dos recursos envolvidos na sessão.

Ao nível do fluxo de meios, parâmetros de digitalização de áudio e vídeo e algoritmos de compressão são fundamentais para a viabilização da transmissão multimédia através da rede. Para isso devem ser disponibilizadas várias opções de configuração de parâmetros relacionados com a digitalização de áudio e vídeo e deve ser permitida a escolha entre vários codificadores de áudio e vídeo possibilitando a variação de características como qualidade dos meios, atrasos e largura de banda.

4.1 Sinalização utilizando o SIP

É durante a sinalização que o chamador e o chamado definem parâmetros para o estabelecimento da chamada e troca de dados, como os endereços de transporte para o envio de meios, o tipo de meios transmitidos, mecanismos de codificação de meios, requisitos de largura de banda, e mecanismos de segurança, entre outros.

4.1.1 Componentes SIP

Para implementar as funcionalidades que o SIP disponibiliza, existem vários componentes, alguns dos quais são ilustrados na Figura 4.1. São eles o *User Agent*, o *Proxy Server*, o *Registrar Server* e o *Redirect Server*.

- *User Agent*: consiste em duas partes distintas, o *User Agent Client* e o *User Agent Server*. O UAC é uma entidade lógica que gera pedidos SIP e recebe respostas a esses pedidos. O UAS é uma entidade lógica que gera respostas aos pedidos SIP. O UA permite normalmente a interface com o utilizador, mas pode também ser um sistema automático que não envolva interacção como um sistema de *voice mail* ou um sistema de redireccionamento de chamadas. A Figura 4.2 (Sessão SIP A) ilustra um exemplo de uma sessão entre dois UAs;
- *Proxy Server*: é uma entidade intermediária que actua como cliente e servidor para o propósito de estabelecer chamadas entre os utilizadores. Com uma funcionalidade semelhante à de um *Proxy HTTP*, o *Proxy Server* tem a tarefa de encaminhar os pedidos que recebe para outras entidades mais "próximas" do destinatário. Na Figura 4.2, a Sessão SIP B ilustra um exemplo de um utilizador (Utilizador A) que recorre a um *Proxy Server* para contactar outro utilizador (Utilizador B). Existem dois tipos de *Proxy Servers*: o *Stateful Proxy* e o *Stateless Proxy*:

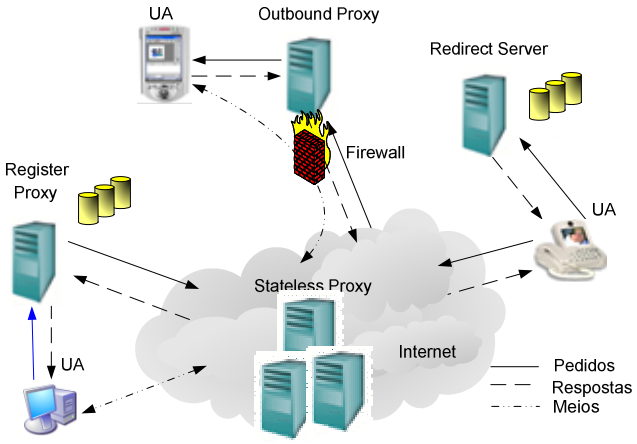

Figura 4.1 – Arquitectura SIP

  - *Stateful Proxy*: mantém o estado das transacções durante o processamento dos pedidos. Permite dividir um pedido em vários (*fork*), na tentativa de encontrar em paralelo múltiplas localizações do chamado e apenas enviar as melhores respostas ao utilizador que fez a chamada;
  - *Stateless Proxy*: não mantém o estado das transacções durante o processamento dos pedidos. São mais adequados quando existem requisitos de velocidade como numa *backbone* de uma infra-estrutura SIP;
  - *Outbound Proxy*: é um *proxy* que recebe pedidos de um utilizador, mesmo que não seja ele o destinatário do pedido. Esta configuração é muito utilizada e adequada quando existem *firewalls*, em que o UA é configurado para enviar pedidos e receber pedidos através deste tipo de servidor.
- *Registrar*: é um servidor que aceita pedidos de registo de utilizadores e guarda a informação desses pedidos para fornecer um serviço de localização e tradução de endereços no domínio que controla;
- *Redirect Server:* é um UAS que gera respostas de redireccionamento aos pedidos que recebe, que devem ser consideradas para completar o pedido inicial. Este servidor não reencaminha os pedidos para o próximo servidor, mas responde, com uma mensagem de redireccionamento (3xx) que contém o endereço do próximo servidor a ser contactado, ao cliente que fez o pedido.

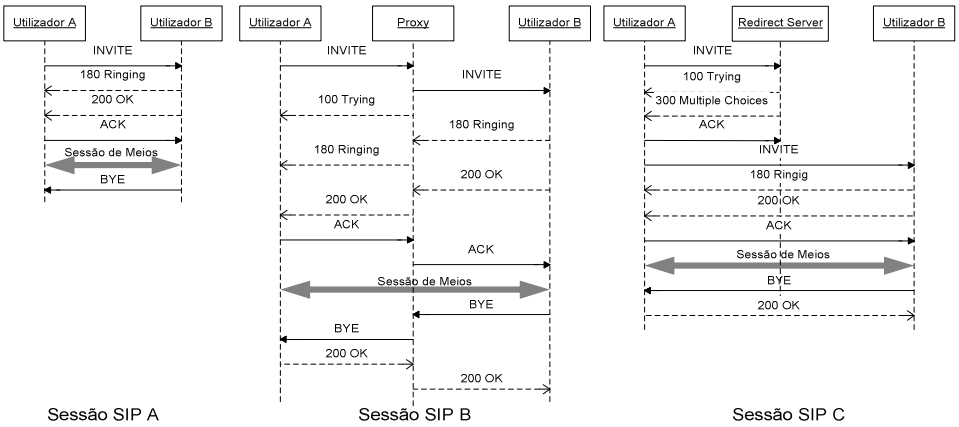

Figura 4.2 - Exemplos de sessões SIP

Na Figura 4.2, a Sessão SIP C mostra um exemplo de um UA (Utilizador A) que utiliza um *Redirect Server* para obter uma lista de localizações alternativas e formular um novo pedido SIP para a localização ou localizações obtidas (neste caso foi directamente para o Utilizador B).

4.1.2 Mensagens SIP

O SIP é um protocolo de texto com uma semântica semelhante à do protocolo HTTP. Esta propriedade permitiu a reutilização de código e uma integração mais simples dos servidores SIP com servidores de *Web* e de *e-mail*. Tal como o HTTP, o SIP define a comunicação através de dois tipos de mensagens: os pedidos e as respostas. Os UACs fazem os pedidos e os UASs retornam respostas aos pedidos dos clientes. Uma mensagem SIP consiste numa linha inicial seguida de um ou mais cabeçalhos (*headers*), uma linha vazia que indica o fim dos cabeçalhos e, por fim, o corpo da mensagem que é opcional. Os cabeçalhos são usados para transportar informação necessária às entidades SIP para processarem os pedidos ou respostas. Caso exista o corpo da mensagem, este é usado para descrever a sessão, contendo normalmente o protocolo *Session Description Protocol* (SDP); no entanto pode ter outro tipo de conteúdo, como ASCII ou HTML.

4.1.3 Cabeçalhos SIP

Os cabeçalhos SIP são similares aos cabeçalhos HTTP tanto na semântica como na sintaxe. Alguns desses cabeçalhos são usados em todas as mensagens enquanto outros só fazem sentido em determinados pedidos ou em respostas. Quando um cabeçalho aparecer numa mensagem e não fizer parte da categoria dessa mensagem é simplesmente ignorado.

4.1.4 Endereços SIP

O SIP identifica o utilizador através de um tipo de *Universal Resource Identifier* (URI) chamado SIP URI [7]. O SIP URI utiliza a forma mais comum de endereçamento de utilizadores na Internet, o formato do endereço de *e-mail*, como por exemplo: *sip:utilizador@dominio*, *sip:utilizador@host*, *sip:utilizador@IP-address* ou *sip:numero-telefone@gateway*. O SIP permite ainda recorrer a identificadores para utilizadores associados a comunicações seguras, denominados SIPS URIs. Este identificador especifica que o recurso a ser contactado é seguro (ex. *sips:utilizador@dominio*). A primeira parte do SIP ou SIPS URI está associada ao utilizador, serviço ou número de telefone. Quando se pretende especificar um utilizador num terminal específico, a segunda parte é normalmente um endereço IP ou o nome do computador no domínio (ex. *sip:jpaulo@sip-multimedia-host.ipb.pt*). Quando o endereço é independente da localização é normalmente especificado o nome de um domínio (ex. *sip:jpaulo@ipb.pt*).

A solução de identificação do SIP é também baseada em entidades existentes na rede IP, como o DNS. Recentemente foi publicada a RFC 3263 [8] que descreve os procedimentos DNS utilizados pelos clientes para traduzir o SIP URI num endereço IP, porta e protocolo de transporte, ou pelos servidores para retornar uma resposta ao cliente caso o pedido falhe. Esta característica permite, como acontece com todos os URIs utilizados na Internet, que os SIP e SIPS URIs possam ser colocados em páginas *Web*, mensagens de *e-mail* ou noutras aplicações. O SIP segue desta forma um paradigma muito simples que lhe permite a utilização de diversos serviços com uma metodologia de "um só endereço".

4.1.5 Criação e finalização de chamadas

A Figura 4.2 ilustra um exemplo do estabelecimento de uma chamada entre dois utilizadores utilizando dois *SIP Proxys*. O *utilizador1* pretende, através de um *software* IPtel instalado no seu PC e identificado como *utilizador1@ipb.pt*, fazer uma chamada para o *utilizador2*,dentro do domínio *utad.pt*, que está registado no *SIP Proxy* como *utilizador2@utad.pt*. Os dois *SIP Proxys* ilustrados na Figura 4.2 têm a função de controlar cada um dos domínios e o registo dos utilizadores que pertencem a esse domínio.

Para iniciar a chamada o *utilizador1* envia ao *utilizador2* um INVITE, que transporta no corpo da mensagem os parâmetros da sessão que pretende estabelecer, utilizando o protocolo SDP. Porque o *software* IPtel não conhece a localização do *utilizador2*, ou tem na sua configuração o *proxy ipb.pt* como *Outbound Proxy*, o INVITE é enviado para o *proxy ipb.pt* que controla o domínio *ipb.pt* e que

terá como função reencaminhar o pedido para um servidor mais próximo do *utilizador2*. No INVITE é incluído o cabeçalho *Route* com o endereço do *proxy ipb.pt* de modo a garantir que todas as mensagens trocadas entre o *utilizador1* e o *utilizador2* passem por este *proxy*.

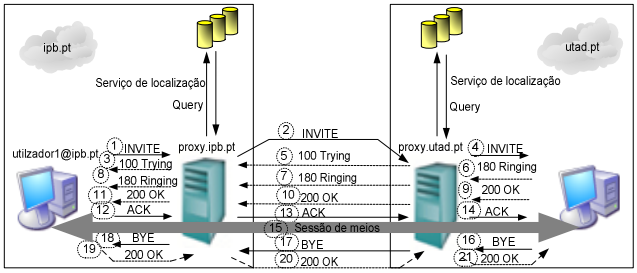

Figura 4.3 - Exemplo de uma chamada entre dois utilizadores

O *Proxy Server,* ao receber a mensagem, terá que tomar uma decisão de encaminhamento do pedido recebido. No decorrer desta decisão o *proxy* envia uma resposta *100 Trying* ao *utilizador1* a indicar que recebeu o INVITE e está a processar o pedido. O *proxy ipb.pt* através de um serviço de localização como uma consulta a uma base de dados conclui que não conhece o *utilizador2*. Utilizando então uma procura DNS descobre o *proxy utad.pt* que controla o domínio *utad.pt* e encaminha o INVITE para esse servidor.

O *proxy utad.pt* recebe o pedido e responde ao *proxy ipb.pt* com uma resposta *100 Trying*. Como o *utilizador2* está registado no seu domínio, o *proxy* encaminha o INVITE para o *software* IPtel deste utilizador.

A aplicação, ao receber o pedido, alerta o *utilizador2* que tem uma nova chamada e envia uma resposta *180 Ringing* para o *utilizador1* através dos servidores, mas agora em sentido contrário. A aplicação do *utilizador1*, ao receber a mensagem *180 Ringing*, alerta o utilizador indicando que está a "chamar". Os *proxys* utilizam a informação contida nos cabeçalhos *Via* para devolver a resposta, removendo o cabeçalho *Via* do topo da mensagem que continha o seu endereço. Enquanto no encaminhamento do pedido INVITE foi necessário recorrer ao DNS e a serviços de localização, as respostas (ex. *180 Ringing*) podem ser retornadas sem invocar os serviços de localização ou mesmo sem manter o estado da transacção nos *proxys*.

Como no exemplo o *utilizador2* aceita a chamada, a sua aplicação envia ao *utilizador1* uma resposta final 200 OK, que transporta no corpo da mensagem os parâmetros da sessão dos meios que pretende usar, recorrendo ao protocolo SDP, completando a negociação básica de capacidades disponibilizada pelo SIP. A aplicação do *utilizador1* ao receber a resposta 200 OK envia uma mensagem ACK para confirmar que recebeu a resposta. Quando recebe a resposta 200 OK é criada uma relação SIP ponto a ponto referida como *diálogo* (*Dialog*). Em seguida é iniciada a troca de meios utilizando os formatos acordados durante o estabelecimento da chamada recorrendo ao protocolo SDP. O SIP não exerce qualquer controlo sobre o encaminhamento dos pacotes dos meios entre os utilizadores, podendo estes percorrer caminhos diferentes das mensagens de sinalização. Para terminar a sessão o *utilizador2* envia um BYE ao *utilizador1*. A aplicação ao receber este pedido, termina o envio dos meios e retorna uma resposta 200 OK, concluindo a chamada e o *diálogo* entre os dois utilizadores.

4.1.6 Segurança

O SIP utiliza diversos mecanismos de segurança adequados a diferentes aspectos e aplicações, como por exemplo a preservação da confidencialidade e integridade das mensagens, prevenção de ataques que permitam os desvios de mensagens ou provoquem a indisponibilidade do serviço ou que proporcionem a autenticação de utilizadores e a privacidade dos participantes numa sessão. A cifra de todas as mensagens é o melhor mecanismo de segurança para a sinalização, garantindo a confidencialidade e integridade das mensagens. O SIP não permite a cifra das mensagens ponto a ponto, devido à possibilidade de estas poderem percorrer várias entidades intermediárias da rede (ex. *Proxy Server*), que têm de analisar os pedidos e repostas para os poderem encaminhar correctamente e que podem também adicionar ou remover informação. Para obter este grau de segurança são preferencialmente recomendados mecanismos de segurança a um nível mais baixo, em que as mensagens são cifradas entre entidades SIP e permitem aos terminais verificar a identificação dos servidores para quem são enviadas as mensagens de forma segura, utilizando sistemas de autenticação criptográfica. A solução para este mecanismo de segurança passa pela utilização dos protocolos *Transport Layer Security* (TLS) [9] e IPSEC [10], que fornecem segurança ao nível da camada de transporte e ao nível da camada de rede respectivamente, permitindo a confidencialidade e integridade das mensagens. O SIP define URI seguros chamados SIPS URI, que permitem o estabelecimento de sessões seguras, garantindo que é utilizado transporte criptográfico (TLS) para entregar as mensagens.

Para a autenticação da identidade dos utilizadores, o SIP define [7] o método de autenticação *Digest* que se baseia no esquema de autenticação HTTP *Digest*, utilizado pelo protocolo HTTP. Este método permite aos utilizadores identificarem-se perante uma entidade através do nome do utilizador e de uma palavra-chave cifrada, utilizando informação que lhe é fornecida por algum tipos de respostas. No entanto, não garante a confidencialidade e integridade das mensagens.

4.2 Codificadores

Os codificadores e descodificadores, comummente denominados *codecs*, são dispositivos que permitem reduzir a largura de banda para a transmissão de dados utilizando técnicas de compressão. Estas técnicas de compressão devem para isso operar em tempo real, devido a características do próprio serviço, como a comunicação interactiva. A compressão de sinais é baseada em técnicas de processamento que eliminam informação redundante ou não perceptível.

Existem várias entidades responsáveis por normalizar codificadores de áudio e vídeo, tais como a *International Telecommunication Union* (ITU), *Telecommunication Industries Association* (TIA) e *United States Federal Standards* (USFS). Para codificar o sinal áudio em tempo real alguns dos codificadores mais conhecidos são ITU-T G.711, ITU-T G.723, ITU-T G.726, ITU-T G.729, CELP, GSM e MPEG-Audio e para a codificação de vídeo em tempo real os codificadores H.261, H.262, H.263 e JPEG.

4.3 Transporte

Para a entrega de meios em tempo real é usado o *Real Time Transport Protocol* definido na RFC 1889. Este protocolo é normalmente utilizado sobre o UDP e permite serviços de entrega ponto a ponto para a transmissão de dados em tempo real.

4.3.1 Real Time Transport Protocol

O RTP [11] é utilizado para o suporte de serviços de transporte em aplicações de tempo real, como por exemplo *streaming* a pedido e serviços interactivos, como sejam a videoconferência e o IPtel. O RTP permite funções de transporte ponto a ponto na rede e é apropriado para aplicações que transmitem dados em tempo real como áudio, vídeo, sobre serviços de redes *unicast* ou *multicast* [11].

4.3.2 Real Time Control Protocol

A principal função do protocolo RTCP [11] é fornecer *feedback* da qualidade dos dados distribuídos. É baseado na transmissão periódica de pacotes de controlo a todos os participantes na sessão, usando o mesmo mecanismo de distribuição que o de pacotes de dados.

Os pacotes RTCP podem conter informação sobre a qualidade do serviço para os participantes da sessão, informação sobre a fonte do *stream* que está a ser transmitido, ou estatísticas sobre os dados que já foram transmitidos até ao momento.

5. IMPLEMENTAÇÃO DO sIPtel

No desenvolvimento do sIPtel procurou-se que este tivesse as funcionalidades mais utilizadas por num serviço típico IPtel. Em seguida são especificadas as características do serviço sIPtel ao nível da sinalização:

- Suporte para uma chamada activa em qualquer momento. Entenda-se por chamada activa a troca de meios entre dois utilizadores;
- Suporte para múltiplas chamadas em espera;
- Possibilidade de especificar se pretende receber e/ou enviar cada um dos dois tipos de meios (áudio e vìdeo);
- Possibilidade de registar utilizadores com e sem autenticação;
- Permissão para a autenticação de utilizadores.

O sIPtel suporta ainda cinco codificadores de áudio e dois codificadores de vídeo. Estes codificadores e todo o mecanismo de tratamento dos meios é foram implementados feito recorrendo às classes da API *Java Media Framework* (JMF). Para o desenvolvimento do mecanismo de sinalização foi utilizada a NIST-SIP *Application Programming Interface* (API) desenvolvida pelo *National Institute of Standards and Technology* (NIST) [12].

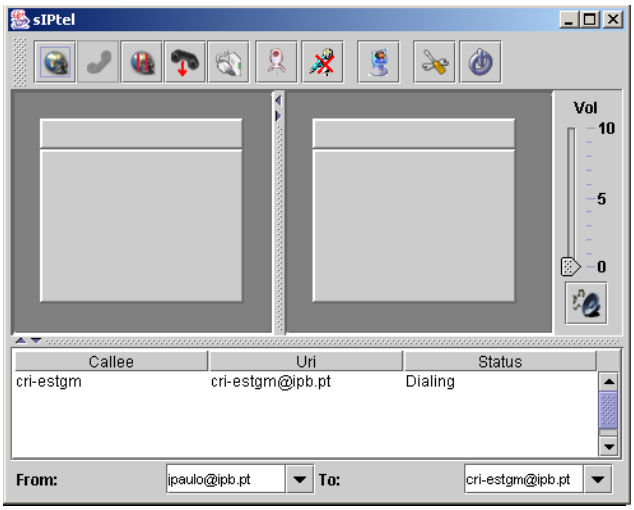

Figura 5.1 - Interface gráfica principal do sIPtel

Ao nível da implementação, o sIPtel está dividido em cinco partes, cada uma correspondendo a um pacote, enunciados em seguida:

- *ipb.iptel.gui*: contém as classes que proporcionam a interface com o utilizador e a apresentação dos meios;
- *ipb.iptel.sip*: contém as classes que permitem implementar a parte de sinalização de chamadas;
- *ipb.iptel.rtp*: fazem parte deste pacote as duas classes responsáveis pelo envio e recepção de meios;
- *ipb.iptel.util*: contém classes utilitárias utilizadas por várias classes contidas nos outros pacotes;
- *ipb.iptel.test*: contém dois utilitários que permitem testar a reprodução de áudio e vídeo no sistema.

O SIP está estruturado como um protocolo em camadas, em que o nível mais baixo é a análise da sintaxe das mensagens e a sua codificação. A segunda camada é o nível de transporte, que define como um cliente envia pedidos e recebe respostas e como um servidor recebe pedidos e envia respostas. A terceira camada é o nível transacção, que trata das retransmissões ao nível aplicacional, associa as respostas aos pedidos e lida com os *timeouts*. A um nível superior existe ainda o conceito de diálogo que pretende facilitar a sequência de mensagens e o encaminhamento apropriado de pedidos entre os UAs.

Durante o desenvolvimento do sIPtel a *stack* NIST-SIP apenas fornecia os dois níveis mais baixos, sendo necessário implementar o nível de transacção e o conceito de diálogo. Estes dois níveis foram implementados obedecendo criteriosamente às especificações definidas na RFC 3261 e revelaram-se uma das maiores dificuldades na criação do sIPtel.

6. MOBILIDADE DO SIP

O SIP permite vários mecanismos de mobilidade ao nível da aplicação. Alguns destes mecanismos são

disponibilizados de forma básica pelo protocolo, enquanto outros podem ser implementados utilizando extensões do SIP. Os mecanismos descritos em seguida não oferecem qualquer suporte de mobilidade na troca dos meios de uma sessão.

6.1 Mobilidade do Terminal

Para suportar a mobilidade do terminal ao nível da aplicação no SIP, é necessário que exista o suporte de mobilidade na rede IP (Mobilidade IP), que se [13], traduzi na capacidade de um terminal se mover enquanto a sessão está activa. A mobilidade do terminal pode ser dividida em três partes, mobilidade antes da chamada, durante a chamada, e recuperação de erros [14].

6.1.1 Mobilidade antes da chamada

O terminal móvel recebe um novo endereço antes de receber ou fazer uma chamada [14]. Cada vez que o terminal detecta que houve alteração do endereço IP, registra-se novamente no seu servidor *Registrar*, actualizando os parâmetros relativos ao mecanismo de descoberta do SIP.

6.1.2 Mobilidade durante a chamada

Se um terminal se mover durante a sessão e detectar a alteração do endereço de rede, deve enviar um novo INVITE ao utilizador remoto, anunciando as alterações dos parâmetros da sessão [14]. Em paralelo deverá ser iniciado o processo descrito no ponto anterior.

6.1.3 Recuperação de erros

Quando por qualquer motivo existe uma situação de quebra endereços, é aconselhável haver um mecanismo automático para a recuperação desta situação de erro. É o caso de dois terminais móveis que durante uma sessão perdem o contacto por momentos e, quando obtêm novamente o contacto, ambos têm endereços IP diferentes. Uma solução de recuperação consiste em ambos se registarem com os novos endereços num servidor *Registrar*. Em seguida qualquer um deles pode restabelecer a sessão enviando novamente um INVITE [15].

6.2 Mobilidade da Sessão

A mobilidade da sessão permite fazer a transferência de sessões entre utilizadores e/ou terminais. O processo de transferência de sessões, envolve a negociação das características da nova sessão. Neste processo é também possível transferir apenas partes dos meios envolvidos na sessão.

O SIP suporta a mobilidade de sessão através de dois mecanismos. Uma das soluções é chamada *third-party call control* [16], que permite, por exemplo, a transferência de chamadas e sinalização de conferência. Esta aproximação envolve pelo menos três entidades, uma dessas entidades mantém um relacionamento de sinalização com as outras duas partes, que trocam os meios.

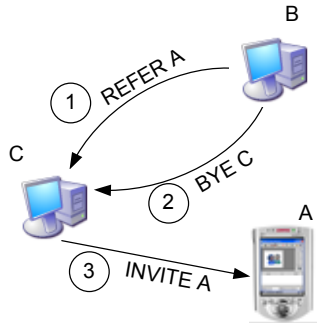

Figura 6.1 – Mobilidade da sessão utilizando o método REFER

A segunda solução é implementada através do método REFER [17], que permite iniciar um pedido entre o UA remoto e uma terceira entidade, como ilustra a Figura 6.1. A sua principal utilização é a transferência de chamadas, podendo também ser utilizada para outros propósitos.

6.3 Mobilidade Pessoal

A sinalização SIP está habilitada a encaminhar pedidos de estabelecimento de chamadas para vários locais, onde os utilizadores indicaram poderem ser contactados. O encaminhamento dos pedidos poderá ser feito de uma forma paralela e ao mesmo tempo (*forking*) ou de uma forma sequencial.

Numa procura paralela (Figura 6.2) o proxy distribui pedidos pelos vários endereços onde é possível o utilizador responder a esse pedido.

Cada um desses pedidos pode gerar respostas que são retornadas através dos vários *proxys* por onde esses pedidos passaram. O SIP possibilita neste processo a definição de regras para o tratamento e retorno dessas respostas ao cliente. Este mecanismo dinâmico disponibilizado de forma básica pelo SIP permite ao utilizador decidir quando, onde e quais os recursos pretende utilizar para responder ao pedido.

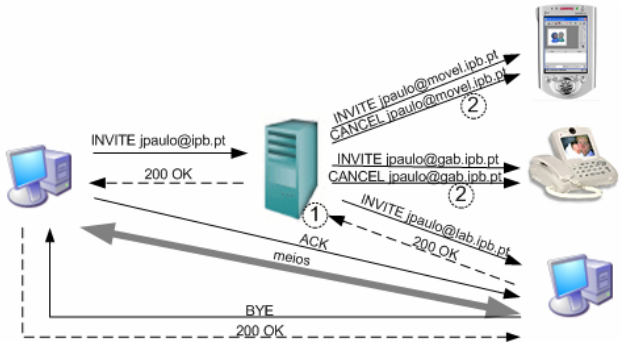

Figura 6.2 – Exemplo de uma procura paralela

Numa procura sequencial (Figura 6.3), o servidor *proxy* tenta contactar cada endereço de contacto sequencialmente, fazendo a próxima tentativa apenas depois da tentativa anterior resultar numa resposta final. A procura acaba quando for gerada uma resposta pertencente à classe sucesso (2xx) ou falha global (6xx) [7].

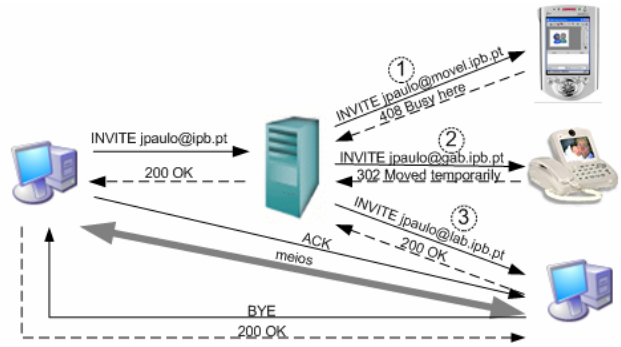

Figura 6.3 – Exemplo de uma procura sequencial

7. CONCLUSÕES

Este artigo analisa a arquitectura de um serviço IPtel, com suporte para vídeo. Esta arquitectura baseia-se num conjunto de protocolos independentes e modulares, os quais quando combinados, permitem implementar um conjunto de características básicas existentes nos serviços de telefonia tradicional e a integração de novos serviços, utilizando como suporte a rede IP.

O sIPtel foi desenvolvido segundo a arquitectura anteriormente analisada, e disponibiliza, através de uma interface gráfica, um conjunto de funcionalidades típicas de um serviço IPtel, tais como, o estabelecimento, o controlo e a finalização de chamadas. Este serviço foi implementado recorrendo ao Java, de modo a beneficiar das características nativas que esta linguagem oferece. Foram ainda utilizadas duas APIs de código aberto, que simplificaram a construção das partes de sinalização e da troca de meios.

O sIPtel foi também sujeito a diversos testes, sobretudo ao nível da interoperabilidade, e para isso foram considerados dois tipos de testes. O primeiro consistiu em avaliar o nível de interoperabilidade através de critérios de avaliação definidos pelo *Technical Program Committee*, tendo -se verificado que o sIPtel satisfaz em 100% os critérios básicos de interoperabilidade. O segundo consistiu em analisar a interoperabilidade do sIPtel com diversos *softwares* IPtel que utilizavam o protocolo SIP para a sinalização de chamadas, verificando-se o sucesso da comunicação em alguns cenários e tendo sido detectadas algumas incompatibilidades no que diz respeito ao anúncio da sessão. Estas incompatibilidades acontecem devido aos proprietários dos softwares, adoptarem diferentes documentos que propõem diferentes cenários para a mesma funcionalidade, como por exemplo a colocação de uma chamada em espera.

A escolha do protocolo SIP, revelou-se na nossa opinião, uma escolha acertada, já que na iniciação deste trabalho as aplicações que utilizavam este protocolo, eram relativamente escassas, verificando-se a sua proliferação nos últimos tempos. Para isso, contribuiu a recente actualização do SIP (RFC 3261), que veio esclarecer alguns detalhes menos claros em relação à versão inicial (RFC 2543) e melhorar o desempenho ao nível da segurança. O SIP foi também escolhido pelo *3rd Generation Partnership Project* (3GPP), para estabelecer sessões multimédia na rede UMTS (R5).

Como trabalho futuro pretende-se a integração no sIPtel de novos serviços, como mensagens instantâneas, presença e de soluções que garantam a qualidade de serviço, uma das características de maior relevância para o aumento da utilização desta tecnologia.

8. AGRADECIMENTOS

Gostaríamos de agradecer ao Eng.º José Manuel Oliveira as conversas esclarecedoras tidas ao longo da execução deste projecto

9. REFERÊNCIAS

[1] Jonathan Lennox e Henning Schulzrinne, “Feature interaction in Internet Telephony”, *Proceedings Feature Interaction in Telecommunications and Software Systems VI*, Glasgow, United Kingdom, Maio 2000. URL: http://www.cs.columbia.edu/~hgs/papers/Lenn0005_Feature.pdf, Julho 2002.

[2] M. Handley e V. Jacobson, "SDP: Session Description Protocol", Request For Comments (RFC 2327), Internet Engineering Task Force, Abril 1998. URL: http://www.ietf.org, Julho 2002.

[3] W. Yeong, T. Howes, S. Kille, “Lightweight Directory Access Protocol”, Request for Comments 1777, Internet Engineering Task Force, Março 1995. URL: http://www.ietf.org, Agosto 2002.

[4] J. Rosenberg, H. Salama, M. Squire, “Telephony Routing over IP (TRIP)”, Request For Comments 3219, Internet Engineering Task Force, Janeiro 2002. URL: http://www.ietf.org, Julho 2002.

[5] J. Postel, “Simple Mail Transfer Protocol”, Request for Comments 821, Internet Engineering Task Force, Agosto 1982. URL: http://www.ietf.org, Agosto 2002.

[6] J. Postel, “User Datagram Protocol”, Request for Comments 768, Internet Engineering Task Force, Agosto de 1980. URL: http://www.ietf.org, Junho 2002.

[7] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler, " SIP: Session Initiation Protocol ", Request For Comments 3261, Internet Engineering Task Force, Junho 2002 (RFC 2543 obsoleta). URL: http://www.ietf.org, Julho 2002.

[8] J. Rosenberg, H. Schulzrinne, “Session Initiation Protocol (SIP): locating SIP servers”, Request for Comments 3263, Internet Engineering Task Force, Junho 2002 (RFC 2543 obsoleta). URL: http://www.ietf.org, Julho 2002.

[9] Dierks, T. e C. Allen, "The TLS protocol version 1.0", Request for Comments 2246, Internet Engineering Task Force, Janeiro 1999. URL: http://www.ietf.org, Setembro 2002.

[10] Kent, S. e R. Atkinson, "Security architecture for the Internet protocol", Request for Comments 2401, Internet Engineering Task Force, Novembro 1998. URL: http://www.ietf.org, Setembro 2002.

[11] H. Schulzrinne., S. Casner, R. Frederick e V. Jacobson, "RTP: a transport protocol for real-time applications", Request For Comments 1889, Internet Engineering Task Force, Janeiro 1996. URL: http://www.ietf.org, Junho 2002.

[12] M. Ranganathan, "Internet Telephony/VOIP", Janeiro 2001, URL: http://dns.antd.nist.gov/proj/iptel/, Novembro 2002.

[13] C. Perkins, "IP mobility support", Request for Comments 2002, Internet Engineering Task Force, Outubro 1996.

[14] H. Schulzrinne, E. Wedlund, "Application-layer mobility using SIP", URL: http://citeseer.nj.nec.com, Setembro 2002.

[15] E. Wedlund, H. Schulzrinne, , "Mobility Support using SIP", *in Second ACM/IEEE International Conference on Wireless and Mobile Multimedia (WoWMoM'99)*, Seatle, Agosto de 1999. URL: http://citeseer.nj.nec.com, Setembro 2002.

[16] J. Rosenberg, J. Peterson, H. Schulzrinne e G. Camarillo, "Best Current Practices for Third Party Call Control in the Session Initiation Protocol," Internet Draft, Internet Engineering Task Force, Março 2003. Work in progress.

[17] R. Sparks, A. Johnston, "Session Initiation Protocol Call Control - Transfer," Internet Draft, Internet Engineering Task Force, Fevereiro, 2003. Work in progress.