

## 6 Avaliação e Testes

Neste capítulo são apresentados alguns testes que foram realizados com o a solução de Gerenciamento de Mobilidade (API SIP User Agent) e com o sistema publish/subscribe desenvolvidos neste trabalho. Inicialmente, são apresentadas duas aplicações de testes para a API SIP User Agent e para o ECI. Finalmente, é apresentada uma avaliação de desempenho do ECI com diferentes configurações (e.g. utilização de Gerenciamento de Mobilidade com travessia de NAT, utilização de Gerenciamento de Mobilidade sem travessia de NAT, e utilização do ECI original, que não utiliza Gerenciamento de Mobilidade), com o objetivo de medir o overhead causado pelo Gerenciamento de Mobilidade. É apresentado também um teste que mostra a vazão do sistema para a publicação de eventos em um cenário de mobilidade.

### 6.1. Testes de Funcionamento da API SIP User Agent

Para testar o funcionamento da API SIP User Agent, foi desenvolvida uma aplicação de testes com uma interface gráfica (GUI) que expõe diretamente as principais funcionalidades da API. Através desta aplicação, é possível inicializar a API com um arquivo de propriedades especificado, estabelecer, atualizar e encerrar sessões de comunicação e adicionar ou remover descrições de mídia locais, permitindo que o resultado das principais operações possa ser visualizado (estas funcionalidades são respectivamente expostas na GUI mostrada na **Figura 35**).

O comportamento da API SIP User Agent na troca de mensagens do protocolo SIP foi analisado a partir de uma ferramenta que permite a visualização das mensagens que trafegam através do servidor SIP. O Proxy SIP fornecido pelo NIST, utilizado para os testes, oferece uma interface gráfica para o acompanhamento da execução do protocolo. Além disso, ele gera um arquivo de registro com todas as mensagens trocadas.

**SIP User Agent Test**

Properties File Complete Path:

**Proxy and Registrar Properties**

Outbound Proxy:  Max Forwards:

Registrar Address:  Registrar Port:  Registrar Transport:

Registrar Status:  Registration Expiration (ms):

**SIP User Agent Properties**

Local User Name:  Local User URI:

Local IP Address:  Local Port:  Local Transport:

**Local Session Description (SDP)**

Local SDP Media Descriptions:

New Media Port:  New Media Protocol:  New SDP Media TCP Setup Type:

**Add Session**

URI to Invite:

**Update Session**

Session Id:

**End Session**

Session Id:

**Communication Sessions:**

```

=====
SessionId: 920654
Remote Party: sip:user2@139.82.24.238

Local SDP:
v=0
o=user1 652450 653819 IN IP4 139.82.24.238
s=-
t=0 0
m=application 50000 UDP app
c=IN IP4 139.82.24.238
m=application 50001 UDP app
c=IN IP4 139.82.24.238

Remote SDP:
v=0
o=user2 69783 71152 IN IP4 139.82.24.239
s=-
t=0 0
m=application 35004 UDP app
c=IN IP4 139.82.24.239
m=application 35006 UDP app
c=IN IP4 139.82.24.239

SIP State: CONNECTED
Media State: CONNECTED
=====

```

Figura 35 - Interface gráfica com o usuário (GUI) da aplicação de testes da API SIP User Agent

## 6.2. Testes de Funcionamento do Sistema Pub/Sub com Gerenciamento de Mobilidade

Da mesma forma com que foi feito para a API SIP User Agent, foi desenvolvida, para testar a API do ECI, uma aplicação com uma interface gráfica com o usuário que expõe as principais funcionalidades desta API. Dentre as principais funcionalidades estão a inicialização da API com propriedades a partir de um arquivo especificado, a obtenção das subscrições existentes no ECIBroker

para o usuário que está utilizando a aplicação, a subscrição para recepção de eventos em um assunto e com uma expressão de interesse, e a publicação de eventos para um assunto especificado possivelmente com propriedades anexadas aos mesmos.

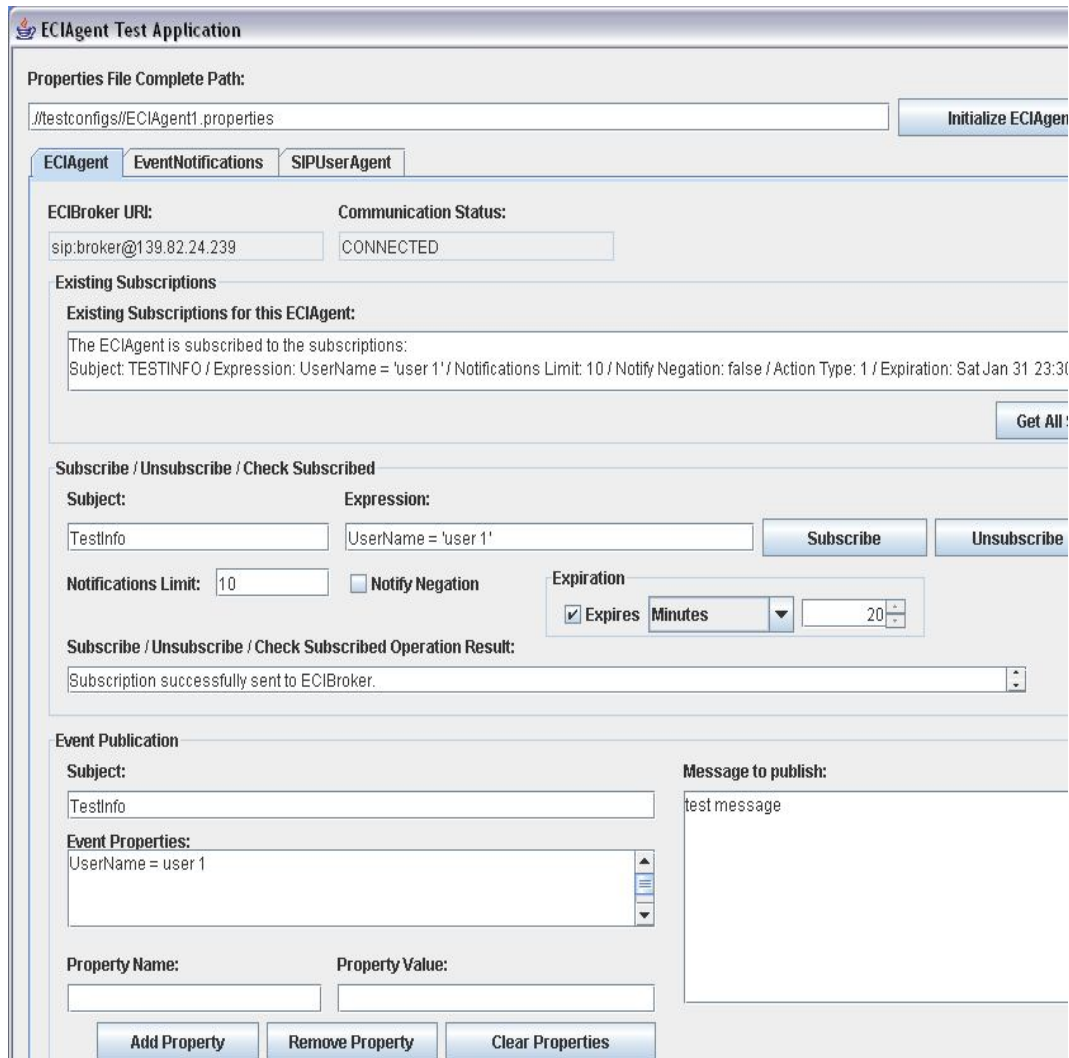


Figura 36 - Aplicação de testes da API ECIAgent com suas principais funcionalidades expostas para o usuário

Além disso, abas com telas auxiliares permitem que o usuário visualize as notificações de eventos recebidas e também visualize dados relativos à sessão de comunicação SIP estabelecida com o ECIBroker, já que o ECIAgent utiliza a API SIP User Agent. A **Figura 37** e a **Figura 38** mostram as telas auxiliares para visualização de notificações de eventos recebidas, e para a visualização de propriedades e sessões de comunicação da API SIP User Agent utilizada pelo ECIAgent para estabelecer uma sessão de comunicação com o ECIBroker.

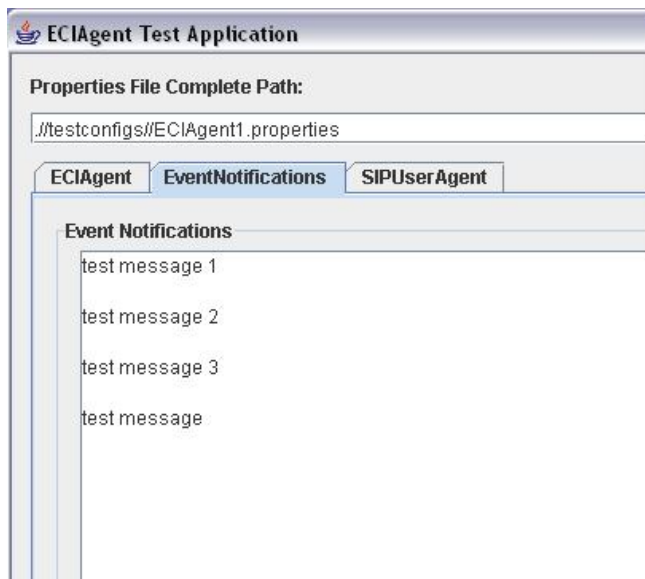


Figura 37 - Tela auxiliar da aplicação de testes da API ECI Agent, onde as notificações recebidas podem ser visualizadas

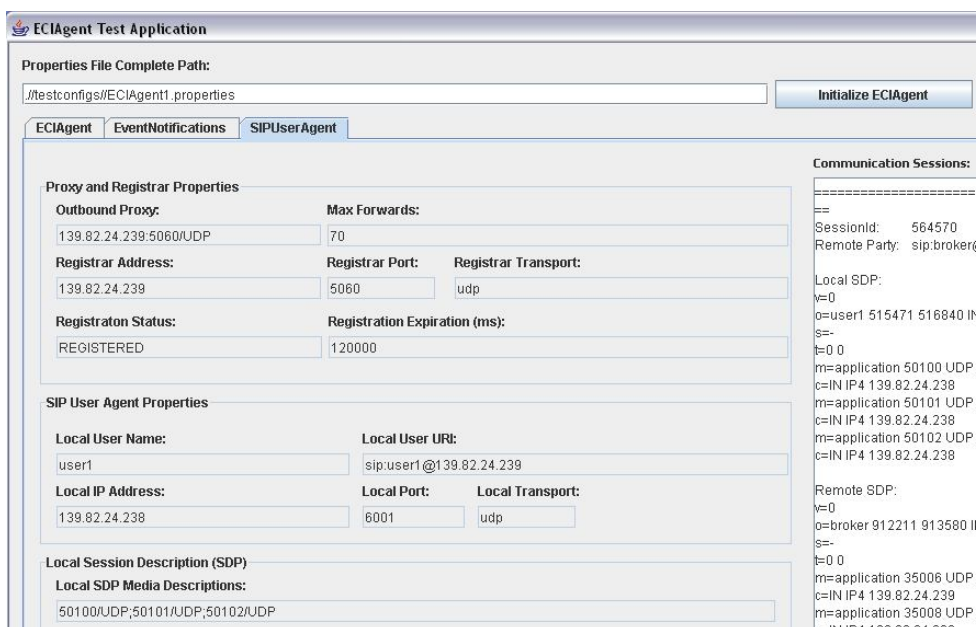


Figura 38 - Tela auxiliar da aplicação de testes da API ECI Agent, onde informações sobre a utilização da API SIP User Agent e a sessão de comunicação com o ECIBroker podem ser visualizadas

### 6.3. Testes de Desempenho

Para avaliar a sobrecarga (overhead) introduzida pelo Gerenciamento de Mobilidade, o ECI com suporte a mobilidade e desconexão desenvolvido foi comparado com o ECI original e também foi avaliada sua utilização com diferentes configurações. Para estas comparações, foi utilizada como métrica a diferença entre o tempo de publicação e o tempo de recepção de eventos. Para

calcular esta diferença, foram incluídas dentro dos eventos duas variáveis contendo o tempo exato da publicação do evento (atribuída pelo publicador no momento da transmissão do evento) e o tempo exato da recepção do evento (atribuída pelo receptor no momento do recebimento do evento). Para que o cálculo desta métrica fosse realizado com precisão (sem a necessidade de sincronização de relógios de diferentes dispositivos), foram estabelecidos no mesmo dispositivo um publicador de eventos e diversos receptores de eventos (o número dos receptores de eventos variou ao longo do teste), todos utilizando a API ECIAgent, enquanto que o ECIBroker foi estabelecido em um dispositivo remoto na rede. O publicador de eventos foi programado para enviar eventos em série com uma periodicidade de vinte eventos por segundo, e os receptores de eventos foram programados para contabilizar a soma e a média dos tempos de publicação dos eventos recebidos. A soma das diferenças de tempos de publicação foi comparada entre as seguintes configurações:

- O ECI com o Gerenciamento de Mobilidade SIP e a utilização de um Media Proxy para permitir a travessia de NATs, subdividindo esta configuração em dois casos:
  - Configuração com o Media Proxy em um servidor remoto ao ECIBroker
  - Configuração com o Media Proxy co-localizado com o ECIBroker
- O ECI com o Gerenciamento de Mobilidade SIP, mas sem mecanismo de travessia de NATs.
- O ECI original, que não possui Gerenciamento de Mobilidade

As curvas mostram a média dos tempos de publicação de eventos no sistema. A taxa de publicação de eventos neste teste foi de 20 eventos por segundo e foram publicados 2000 eventos para cada amostra. A curva que representa o ECI original (chamada no gráfico de ECI) apresenta um desempenho tão melhor que as demais, pois, no ECI original não existe overhead da utilização do protocolo de UDP confiável, onde o recebimento de cada pacote UDP deve ser confirmado pelo receptor.

Já a curva que representa o ECI com Gerenciamento de Mobilidade SIP (chamada no gráfico de ECI SIP) apresenta um pequeno crescimento no tempo de publicação, provavelmente devido à necessidade de confirmação de cada pacote enviado através do protocolo RUDP. A curva que representa o ECI com

Gerenciamento de Mobilidade e com mecanismo de travessia de NAT, possui tempos de publicação de eventos ainda maiores, visto que todos os eventos devem passar através do Media Proxy, que neste caso está co-localizado com o ECIBroker a fim de minimizar o desvio no tráfego de dados através de um terceiro nó na rede. Finalmente, a curva que representa o ECI com Gerenciamento de Mobilidade, travessia de NAT, e o Media Proxy remoto é a que apresenta o pior desempenho visto que todos os eventos devem passar por um terceiro nó na rede, causando um aumento no tempo de transmissão de dados.

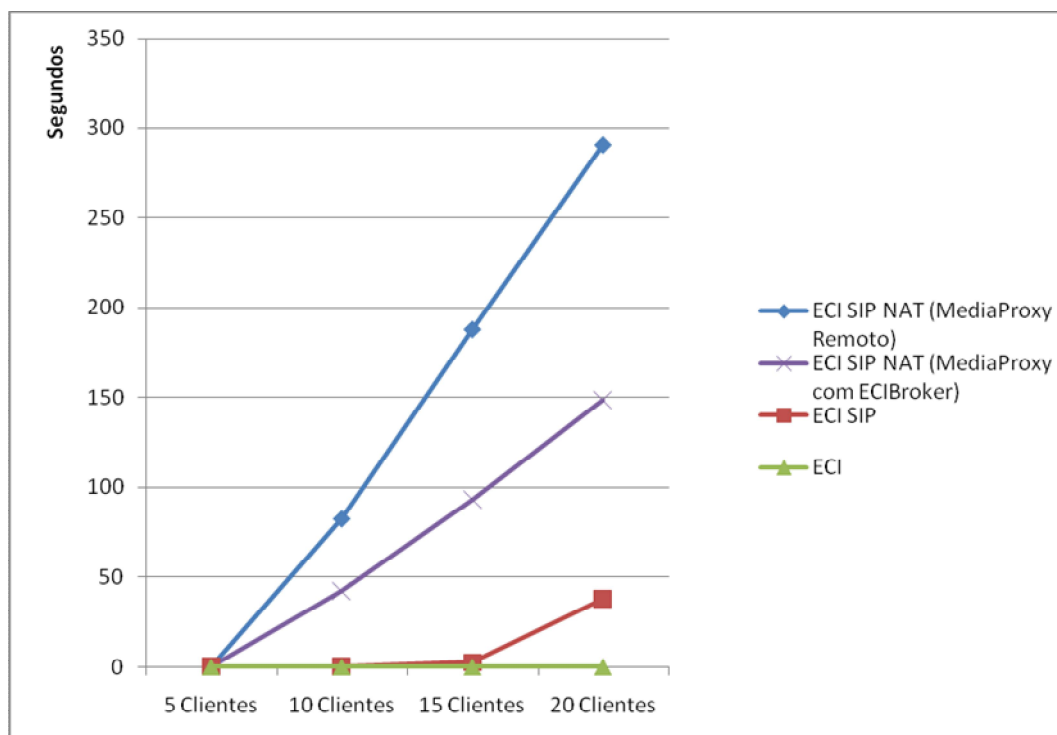


Figura 39 - Avaliação do overhead causado pelo Gerenciamento de Mobilidade.

Para avaliar a vazão do sistema em um cenário de mobilidade, isto é, a capacidade do sistema de entregar os eventos publicados para receptores localizados em dispositivos que mudam de endereço IP, foi utilizada como métrica a porcentagem de perda de eventos para diferentes periodicidades de mudança de IP de um dispositivo e para diferentes configurações no tamanho do buffer utilizado no ECIBroker para armazenar os eventos que não podem ser entregues aos receptores enquanto estes permanecem desconectados durante uma mudança de endereço IP. Para isto, um publicador foi estabelecido em um dispositivo na rede, enquanto que em outro dispositivo (executando em um dispositivo móvel na rede sem fio) foi estabelecido o receptor de eventos. O dispositivo no qual foi estabelecido o receptor de eventos foi programado para

mudar de endereço IP com diferentes periodicidades (i.e. foi programado para mudar de endereço IP a cada minuto, a cada meio minuto e a cada terço de minuto). O publicador de eventos foi programado para enviar publicações de eventos em série a uma taxa de vinte eventos por segundo e foram publicados dois mil eventos para cada amostra. Para contabilizar a perda de eventos, cada receptor contém um contador de eventos recebidos. A tabela abaixo mostra os resultados do teste de vazão, com as porcentagens de perda de eventos para as diferentes periodicidades de mudança de endereço IP do dispositivo e para diferentes tamanhos no buffer de eventos no ECIBroker.

Tamanho do buffer de eventos no ECIBroker	Perda de eventos (%)		
	1 min	1 / 2 min	1 / 3 min
100	8,38%	13,94%	27,67%
150	4,21%	7,77%	27,55%
200	0,00%	0,00%	1,22%
Periodicidade de mudança de IP:	1 min	1 / 2 min	1 / 3 min

Tabela 2 - Avaliação da vazão do sistema em um cenário de mobilidade. A taxa de publicação de eventos neste teste foi de 20 eventos por segundo e foram publicados 2000 eventos para cada amostra

Os resultados mostram que a configuração do ECIBroker para manter um buffer de eventos de no máximo duzentos eventos foi suficiente para possibilitar que nenhum evento seja perdido enquanto o dispositivo muda de endereço IP no máximo duas vezes por minuto, para uma periodicidade de publicação de eventos de vinte eventos por segundo. Os principais fatores que influenciam os resultados deste teste são a periodicidade de envio de eventos por parte do publicador, o tempo em que os receptores permanecem desconectados enquanto mudam de endereço IP e o tamanho do buffer de eventos que podem ser armazenados no ECIBroker. O tempo que o dispositivo leva para mudar de endereço IP (i.e. o tempo em que um receptor permanece desconectado) influencia diretamente a quantidade de eventos que necessita ser armazenada no buffer. Neste teste, a mudança de endereço IP foi forçada para ser realizada na periodicidade determinada e com um pool conhecido de endereços IP, os quais o dispositivo pode utilizar (i.e. mudar de um endereço IP conhecido para outro) com um tempo constante. Entretanto, em um cenário de rede sem fio onde a mudança de endereço IP depende de um serviço de configuração automática (e.g. DHCP) que não apresenta um tempo constante para a mudança de endereço, os resultados acima apresentados podem não se aplicar, sendo necessária uma configuração do

ECIBroker com um tamanho maior das filas de eventos, ou a limitação na periodicidade da publicação de eventos por parte do publicador.

Para validar a solução em dispositivos móveis portáteis, o MD-EDI e o SIP User Agent foram portados para o Sistema Operacional Android (Google), e testados em um dispositivo G1. A **Figura 40** mostra o tempo médio de publicação de eventos para um ECIAgent executando em um dispositivo G1 comparado ao tempo médio de publicação de eventos para um ECIAgent executando em um PC tradicional (notebook).

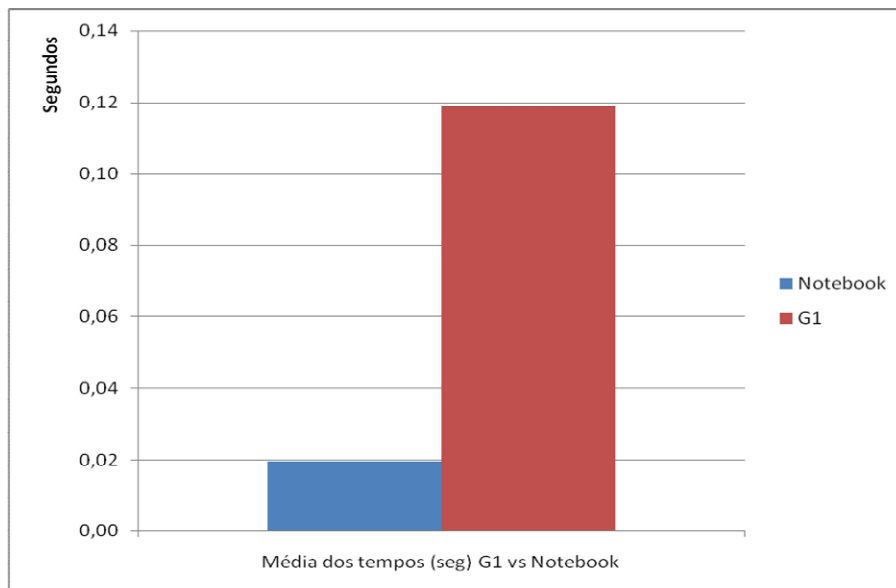


Figura 40 - Avaliação do desempenho do MD-EDI no Android/G1