Afonso Zimmermann

Asterisk Cluster

São José – SC

Março / 2008

Afonso Zimmermann

Asterisk Cluster

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

Orientador:
Marcelo Araujo

Co-orientador:
Prof. Eraldo Silveira e Silva

Curso Superior de Tecnologia em Sistemas de Telecomunicações
Centro Federal de Educação Tecnológica de Santa Catarina

São José – SC

Março / 2008

Monografia sob o título *"Asterisk Cluster"*, defendida por Afonso Zimmermann e aprovada em 18 de março de 2008, em São José, Santa Catarina, pela banca examinadora assim constituída:

Marcelo Araujo
Orientador

Prof. M. Eraldo Silveira e Silva
Coorientador

Prof. M. Emerson Ribeiro de Mello
CEFET / SC

Resumo

Neste trabalho são apresentadas duas técnicas de Cluster possíveis de se usar com sistemas baseados no Asterisk.

O DNS Round-Robin e o DUNDi. O DNS Round-Robin é configurado diretamente no servidor DNS, podendo ser usado para programas compatíveis com o Linux.

O DUNDi está incorporado ao Asterisk desde a versão 1.0. Este possui um arquivo de configuração própria, necessitando de um canal de comunicação entre os servidores que compõe o Cluster. Sozinho ou aliado a outras técnicas, pode ser usado para prover várias formas de Cluster. Nesse trabalho foi usado para prover balanceamento de carga com a ajuda do DNS Round-Robin.

Para se chegar a essas duas técnicas, foi realizada uma pesquisa para ver quais técnicas poderiam ser usadas com o Asterisk. Então passou-se para um estudo das técnicas para realizar a implementação das mesmas. Por fim foi testada a eficiência do DUNDi, de forma a permitir avaliar a necessidade ou não de seu uso.

Palavras-Chave: Cluster, Asterisk, IAX.

Abstract

In this work are presented two techniques of Cluster possible to use with systems based on Asterisk. The DNS Round-Robin and the DUNDi.

The DNS Round-Robin is configured directly on the DNS server and it can be used for programs compatible with Linux.

The DUNDi is incorporated to Asterisk since version 1.0. He has a configuration file itself and require a channel of communication between the machines that make up the cluster. Alone or together with other techniques, it can be used to provide various forms of Cluster. That work was used to provide load balancing, with the help of the DNS Round-Robin. It tested its effectiveness in some situations.

To achieve these two techniques, a search was carried out to see what techniques could be used with Asterisk. Then it moved to a study of techniques to achieve the implementation of them. Finally was tested the efficiency of DUNDi, to allow assess the need or otherwise of their use.

Keywords: Cluster, Asterisk, IAX.

Sumário

Lista de Figuras

1 Introdução

Atualmente, centrais telefônicas baseadas em VoIP (Voz sobre IP) estão se tornando cada vez mais comuns. Entre as plataformas mais utilizadas está o Asterisk, que se executa em sistemas UNIX compatíveis. Entre as várias vantagens que ele possui está o fato do limite de ramais e troncos ser determinado pelo computador onde está o PABX. Por outro lado, sistemas maiores podem demandar uma carga de processamento muito alta até mesmo para computadores modernos.

Para operar com vários ramais SIP ou IAX2 e com troncos SIP ou IAX2, um bom volume de chamadas podem ser estabelecidas por um só servidor. Isso usando codecs PCM "alaw" ou "ulaw". O uso de outros codecs causa um aumento na carga do processador, pois além de gerar o PCM ainda precisam fazer a codificação para o respectivo codec para aumentar a compressão. Mas mesmo com "alaw" ou "ulaw", ao se aumentar muito o volume de chamadas, começa a ocorrer picos de processamento que fazem com que ocorra picotamento da voz ou até a ausência dela por um longo período no canal.

Esse problema pode ser ainda agravado pelo uso de placas para a interconexão do PABX com a PSTN, sendo tanto via entroncamento E1 como via entroncamento analógico. Isso porque essas placas utilizam muitas interrupções por segundo, usando processamento e principalmente congestionando a IRQ atribuída ao barramento em que essa placa se encontra.

Uma das formas para poder aumentar a capacidade do PABX é a formação de arranjos computacionais. Tais sistemas, denominados de "Clusters", se utilizam de técnicas e ferramentas específicas para a distribuição de carga de processamento, redundância ou tanto balanceamento de carga como redundância.

Neste projeto foi implementado um sistema que utiliza vários servidores, balanceando carga entre elas, de forma a permitir uma maior capacidade no PABX com o Asterisk. As ferramentas escolhidas foram o DNS Round Robin e o DUNDi. Como PABX foi escolhido o DISC-OS (DISC-OS, 2007), distribuição baseada no Asterisk.

1.1 Objetivo

O objetivo geral deste trabalho é implementar um sistema capaz de melhorar o desempenho do DISC-OS, fazendo com que ele consiga atender uma carga maior de serviços, com capacidade de espansão se necessário.

1.2 Organização do texto

O texto está organizado da seguinte forma: No capítulo 2 é apresentado a fundamentação teórica que serviu de base para definir o rumo do projeto. No capítulo 3 são apresentadas as ferramentas escolhidas para realização do Projeto. No capítulo 4 são apresentados os experimentos realizados e os seus resultados. No capítulo 5 são apresentadas as conclusões sobre este projeto. Por fim, no anexo A são apresentadas informações extras das instalações, configurações e testes realizados.

2 Fundamentação Teórica

2.1 Histórico sobre Cluster

Em 1993 começa-se a trabalhar em um sistema de Cluster distribuído usando computadores convencionais no lugar dos Mainframes. O projeto é batizado de Beowulf (MERKEY, 2004) . Ele obtém êxito e acaba sendo usado pela Nasa entre outros.

O primeiro protótipo do Beowulf era composto de processadores DX4 e eram conectados por uma rede Ethernet 10Mbit/s. Em 1997 se usava 16 processadores P6 de 200 MHz e rede Ethernet de 100Mbiti/s (SUSIN, 2001).

2.2 Definição de Cluster e apresentação de seus componentes

Cluster é um sistema composto por dois ou mais computadores independentes e interligados entre si por alguma forma de rede (BAKER, 2000).

Um cluster é composto tanto por hardware quanto por software, e juntos viabilizam o Cluster.

2.2.1 Hardware

Os hardwares principais associados ao Cluster são os nós (computadores independentes) executando algum processo e a rede de comunicação dedicada entre eles (BAKER, 2000).

O nó possuí vários componentes que irão influenciar na capacidade do Cluster. São eles o Processador, Memória, Mídia de Armazenamento e Interface de Rede. Todos esses componentes já estão presentes tanto nas estações de trabalho quanto em servidores de alto desempenho, não sendo necessário adquiri-los especificamente para realizar um Cluster

(BAKER, 2000).

Para que os nós possam se comunicar e formar efetivamente um Cluster, é necessária uma rede que os interligue. Clusters do tipo Beowulf usavam a tecnologia Ethernet para a interconexão dos nós. Dependendo da necessidade de tráfego e da latência pode-se empregar técnicas de interconexão específicas para o Cluster (BAKER, 2000).

2.2.2 Software

Apenas possuir computadores independentes e uma interconexão entre eles não forma um Cluster. O software deve ser capaz de prover a interligação lógica entre os nós e permitir o processamento paralelo (BAKER, 2000).

Os softwares usados em Clusters são divididos em duas categorias: Ferramentas de Programação e Sistema de Gestão de Recursos (BAKER, 2000).

As ferramentas de programação fornecem linguagens, bibliotecas e ferramentas de debug para prover a programação paralela (BAKER, 2000).

Os sistemas de gestão de recursos relacionam a instalação, configuração e administração do hardware e do software existente (BAKER, 2000).

2.3 Tipos de Cluster

Os Clusters são classificados de Três formas (SILVA, 2005):

- Alta Disponibilidade
- Balanceamento de Carga
- Alto Desempenho

Clusters de Alta Disponibilidade são arranjos de Failover (Redundância). Normalmente são constituídos de servidores com a mesma função. Um servidor trabalha preferencialmente, e quando ele para de funcionar, o outro assume (SILVA, 2005).

Clusters de Balanceamento de Carga são arranjos que tem o objetivo de dividir o processamento entre os servidores, fazendo com que cada servidor trabalhe uma requisição difirente do processo. Normalmente possuem um elemento balanceador de carga, para dividir as requisições (SILVA, 2005).

Clusters de Alto Desempenho são arranjos que tem como objetivo dividir processamento, como no Balanceamento de Carga. Porém eles devem tratar partes de um mesmo processo, de forma a aumentar a performance do processamento. Normalmente se utilizam de processamento paralelo (SILVA, 2005).

2.4 Softwares Capazes de Implementar um Cluster

Alguns softwares que são capazes de implementar sistemas de Cluster estão listados abaixo:

- DNS Round-Robin
- OpenMosix
- OpenMP
- PVM
- MPI

O DNS Round-Robin é a capacidade que o servidor DNS tem de atribuir dois ou mais endereços IP para um mesmo nome. Com isso pode-se realizar um balanceamento de carga simples.

O OpenMosix é uma extensão ao Kernel Linux que transforma uma rede de computadores em um super-computador. Os servidores com OpenMosix se monitoram afim de saber se um servidor tem recursos disponíveis (processador e memória) para compartilhar. Quando encontra, o servidor com mais carga envia um ou mais processos para que o outro servidor possa auxiliá-la (KNOX, 2002).

O OpenMP é uma API para multiprocessamento para as linguagens C/C++ e Fortran. Pode ser usados em sistemas UNIX e no Windows NT. Ele provê processamento paralelo (FRIEDMAN, 1997).

O PVM é um programa que permite servidores UNIX e/ou Windows trabalharem em uma rede de dados distribuindo os seus processos para que esses sejam usados de forma paralela. A partir da sua terceira versão em 1993 adquiriu suporte à tolerância a falhas (Fault-Tolerant) (PVM, 2007).

O MPI é uma biblioteca para processamento paralelo. Ele provê uma comunicação básica entre os processadores. Como o OpenMP, é implementado em C/C++ e Fortran, além de prover uma API para o processamento paralelo (GROPP; LUSK, 2005).

Além desses softwares, pode ser usado o OpenSER integrado ao Asterisk se o Cluster for uma central Telefônica VoIP. O OpenSER é um servidor SIP. Ele é bem mais leve que o Asterisk como servidor de registro. É usado como servidor de registro e interligado ao Asterisk que cuida dos serviços.

Existem também softwares capazes de implementar Clusters em aplicações específicas. Um deles é o DUNDi.

O DUNDi é um sistema para interligação de PABX Asterisk. Possui uma rede própria para a pesquisa de números de ramais, e usa os protocolos SIP/RTP, IAX2 ou H323 para a transmissão de voz. Os pacotes DUNDi usam o protocolo de transporte UDP. O DUNDi pode ser usado para balanceamento de carga.

3 Cluster Asterisk com DUNDi e DNS Round-Robin

Para a escolha dos softwares a serem utilizados no projeto foi feita uma pesquisa sobre os softwares citados no item anterior.

Como o projeto tem como objetivo implementar um sistema de Cluster no Asterisk, mais precisamente usando o sistema DISC-OS, que usa o Asterisk 1.2, considerou-se o DUNDi a melhor opção, por ser um programa feito pelo próprio criador do Asterisk e imcorporado ao mesmo.

Também optou-se por implementar o DNS Round-Robin por se tratar de uma solução muito simples e que poderia usada junto com outras soluções.

3.1 DNS Round-Robin

O DNS Round-Robin é uma forma de realizar um balanceamento de carga pelo próprio servidor DNS. Seu funcionamento consiste em dividir as conexões entre os servidores. Essa divisão é feita a partir do primeiro endereço cadastrado no arquivo até o último.

Algo importante sobre o DNS Round-Robin é que ele não é uma solução de redundância. Se um dos servidores parar de funcionar, o DNS Round-Robin continuará mandando requisições para esse servidor.

3.2 DUNDi

Segundo J.R.Richardson (RICHARDSON, 2006a),

> "DUNDi é um sistema peer-to-peer para localização de gateways Internet para serviços de telefonia em um Cluster Asterisk. Diferente dos serviços centralizados tradicionais (como o padrão ENUM consideravelmente simples

> e conciso), o DUNDi é completamente distribuído sem nenhuma hierarquia centralizada.".

Com o DUNDi é possível localizar uma extensão em qualquer um dos servidores do Cluster. Para isso pode-se usar um servidor de consultas, que será responsável por fazer todas as consultas, ou ligar os servidores do Cluster diretamente. O servidor de consultas será responsável por procurar a extensão nos servidores de registro. Se um dos servidores de registro falhar, ele continuará procurando em outros servidores pela extensão (MEGGELEN; SMITH; MADSEN, 2005) (RICHARDSON, 2006b).

O DUNDi também pode ser integrado ao Asterisk Realtime. Este é um sistema que permite ter a configuração do Asterisk num Banco de dados SQL. Com isso é possível realizar alterações no plano de discagem sem ter de reiniciar o Asterisk (e perder as chamadas correntes). Outra utilidade é que se um dos servidores do Cluster DUNDi parar de funcionar, o Asterisk Realtime pode localizar e ligar essa extensão mesmo que ela não esteja registrada (se ela for um ramal ou serviço que apenas se registrava no servidor).

Existe também a "Rede DUNDi", ao qual participam algumas empresas do Brasil e do mundo. O seu objetivo facilitar a interligação dos sistemas Asterisk, além de facilitar a interconexão com a PSTN.

3.3 Uso do DUNDi e do DNS Round-Robin para a formação de um Cluster Asterisk

Após realizar a configuração do DNS Round-Robin e do DUNDi, foi verificado o seu comportamento:

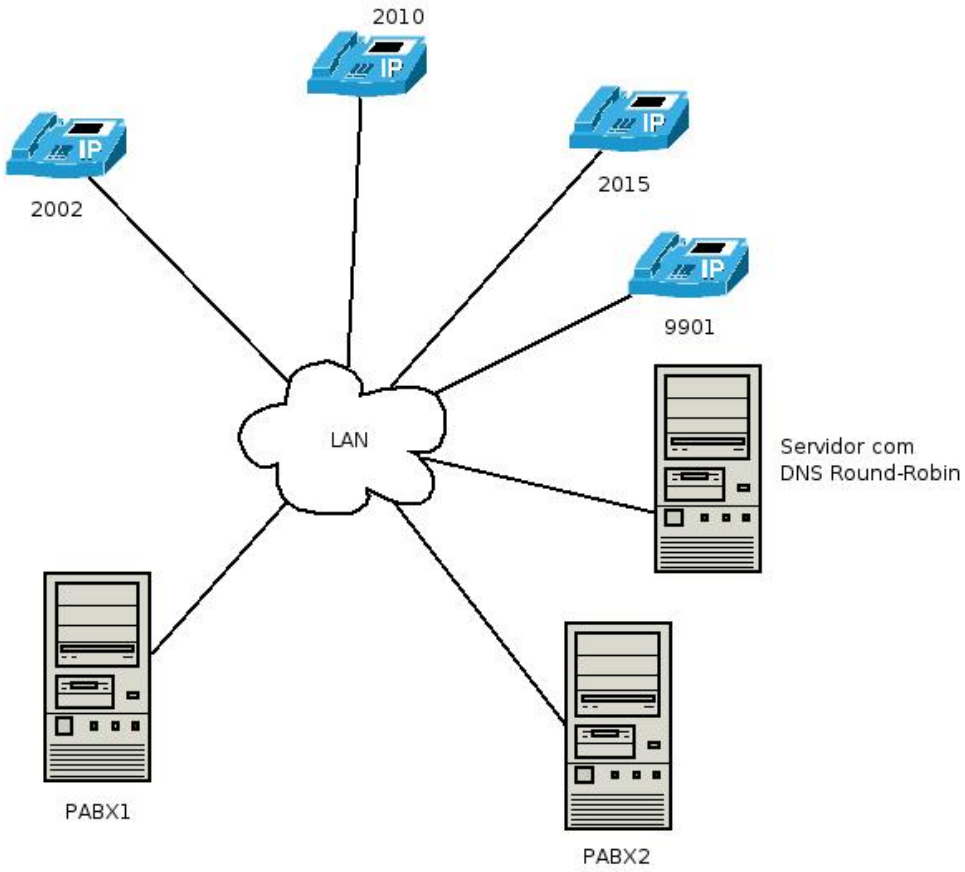

Figura 1: Cenário inicial de testes

Nesse caso, o DNS Round-Robin atua como se fosse um servidor proxy. Ele receberá a requisição dos ramais para registro e enviará para um dos PABX , balanceando as requisições dos mesmos. O DUNDi permitirá a esses ramais se localizarem como se estivessem todos na mesma central.

Supondo que o usuário do ramal 9901, que está registrado no PABX1 deseja falar com o usuário do ramal 2010, que está no PABX2, o usuário do ramal 9901 discará: 2010. O PABX1 verificará se o ramal 2010 está registrado nele para realizar a chamada. Ao verificar que ele não se encontra registrado, fará uma consulta DUNDi ao PABX2 para saber se o ramal está registrado nele. Nesse caso receberá uma resposta afirmativa do PABX2. A partir desse ponto os dois PABX estabelecem um canal IAX para a chamada. Então ela transcorre no PABX2 normalmente.

Abaixo segue uma troca de sinalização DUNDi, para o exemplo anterior. Foi retirado do PABX onde estava o ramal 2010, logo a sinalização começa com um pacote recebido. Foi gerado pelo console do Asterisk:

```
Erx-Frame Retry[No] -- OSeqno: 000 ISeqno: 000 Type: DPDISCOVER  (Command)
  Flags: 00 STrans: 08554  DTrans: 00000 [192.168.130.10:4520]
  VERSION                 : 1
  DIRECT EID              : 00:13:72:36:46:45
  CALLED NUMBER           : 2010
  CALLED CONTEXT          : dundi
  TTL                     : 1

Tx-Frame Retry[No] -- OSeqno: 000 ISeqno: 001 Type: ACK      (Response)
  Flags: 00 STrans: 01286  DTrans: 08554 [192.168.130.10:4520]
  ETx-Frame Retry[No] -- OSeqno: 000 ISeqno: 001 Type: DPRESPONSE  (Response)
    Flags: 00 STrans: 01286  DTrans: 08554 [192.168.130.10:4520] (Final)
    ANSWER          : [EXISTS] 0 <IAX/dundi:e0jtNMPrOhTGjb0uTjQKfg@192.168.65.100/2010>
from [00:13:72:02:f5:f6]
    HINT            : [UNAFFECTED]
    EXPIRATION      : 5
```

Figura 2: Mensagens DPDISCOVER e DPRESPONSE

Alguns parâmetros dessas mensagens são (SPENCER, 2004):

DIRECT EID = Identificação do outro nó do Cluster. Normalmente é usado o endereço MAC da interface de rede.

CALLED NUMBER = Número procurado.

CALLED CONTEXT = Contexto onde a extensão deve ser procurada. Na verdade o seu valor refere-se ao tronco que o DUNDi usará para fazer a pesquisa. Na configuração do DUNDi se associa esse tronco ao contexto local onde deve existir o número procurado. Para mais detalhes sobre a configuração do DUNDi, olhar a seção 7.2.

TTL = Número de possíveis encaminhamentos desse pacote. Não é igual ao TTL do IP. Para mais detalhes sobre o que é o TTL, olhar a seção 7.2.

ANSWER = A resposta enviada pela mensagem DPRESPONSE. No caso de uma resposta afirmativa, usa-se o formato definido na configuração do DUNDi. Este pode ser consultado na seção 7.2.

O objetivo da mensagem DPDISCOVER é dar início a procura de um número DUNDi dentro do Cluster. O DPDISCOVER inicia com uma operação e é respondida inicialmente com um ACK (mensagem de confirmação de recebimento) e então é seguida da mensagem DPRESPONSE. O objetivo da mensagem DPRESPONSE é responder uma DPDISCOVER.

Existem mais mensagens, porém essas são as mensagens principais. O console do Asterisk também exibe algumas mensagens relacionadas à criptografia, mas para uma análise

completa com todas as mensagens, é necessário usar um programa para captura de pacotes. O comando "dundi debug", é usado para ajudar a resolver problemas relacionados ao DUNDi.

Mas supondo agora que o usuário do ramal 9901, que está registrado no PABX1 deseja falar com o usuário do ramal 2015, que não está registrado em nenhum PABX, o usuário do ramal 9901 discará: 2015. O PABX1 verificará se o ramal 2015 está registrado nele para realizar a chamada. Ao verificar que ele não se encontra registrado, fará uma consulta DUNDi ao PABX2 para saber se o ramal está registrado nele. Nesse caso receberá uma resposta negativa do PABX2. Como não há outro PABX para ser consultado, o PABX1 enviará a mensagem "not found" que será convertida pelo softphone, ata ou telefone IP em tom de ocupado.

4 Experimentos Realizados

Para a realização dos testes de desempenho do DUNDi foram configurados dois cenários. O primeiro cenário configurado foi:

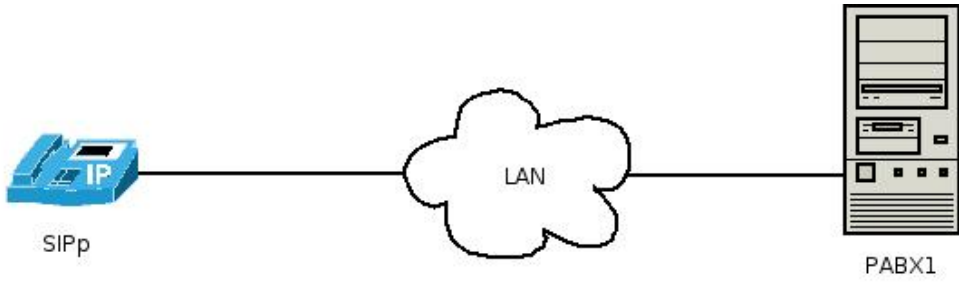

Figura 3: Cenário de Teste sem Cluster

Nesse caso, não há cluster envolvido. Esse cenário serve apenas para comparar o desempenho de um servidor rodando o DUNDi e um servidor sozinho.

Para gerar o tráfego telefônico, foi usado o programa SIPp (SIPP, 2004). Foram gerados diferentes volumes de chamadas para a URA configurada no DISC-OS do servidor PABX1.

O segundo cenário configurado foi:

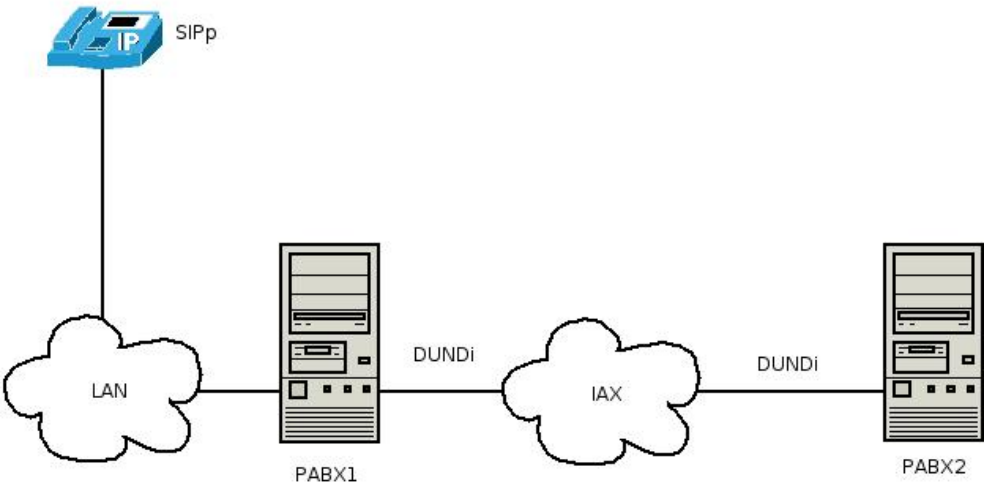

Figura 4: Cenário de Teste com Cluster

Cada um dos dois servidores do Cluster possui os mesmos ramais configurados (2002, 2010, 9001), o mesmo plano de discagem e os mesmos serviços. Num primeiro momento esses ramais foram usados para testar o funcionamento correto do DUNDi. O DNS Round-Robin foi usado para distribuir os ramais nos servidores de forma transparente.

Foi configurada uma URA (Atendimento Automático) com uma música de aproximadamente 7 minutos, para que todas as ligações ficassem estacionadas nessa URA (como se fossem ligações telefônicas).

Para gerar o tráfego telefônico, foi usado o programa SIPp. Foram gerados diferentes volumes de chamadas para a URA configurada no DISC-OS do PABX1. Para que o DUNDi operasse nessa situação, parte das chamadas foram geradas para o ramal 2010 (que estava cadastrado no PABX2). No PABX2, foi alterado o dialplan do ramal 2010 para que tudo o que viesse para esse ramal fosse encaminhado para a URA do mesmo servidor.

Dessa forma, o PABX1 teve de fazer a consulta DUNDi para saber se o ramal 2010 estava presente no PABX2, e depois encaminhar a chamada via tronco IAX para o mesmo.

A linha original:

```
exten => 2010,1,ChanIsAvail(IAX2/2010|sj)
```

Foi mudado para:

```
exten => 2010,1, Goto(disc-ivr-7000,s,1)
```

Sendo 7000 o número correspondente a URA configurada no PABX2.

4.1 Resultados

O PABX1 é composto de um processador Intel Pentium D 2.8GHz, com 4MB de memória cache L2, 512MB de memória RAM, Kernel 2.6.9 (CentOS 4).

O PABX2 é composto de um processador Pentuim 4 3.0GHz, com 1MB de memória cache L2, 512MB de memória RAM, Kernel 2.6.9 (CentOS 4).

Para gerar os dados, foi usado o comando "vmstat". Ele mede níveis de processamento, memórias e outros parâmetros do sistema. As saídas do comando "vmstat" e "sipp" usadas na geração desses gráficos estão na seção 7.3. Nesse caso, foi aproveitado somente o nível de processamento. O comando "vmstat 5 10" indica que que estão sendo feitas 10 amostras de 5 em 5 segundos. O único problema do comando "vmstat" é que a sua primeira amostra é imprecisa. Por isso aparecerão valores na primeira amostra as vezes muito diferentes do resto das amostras. A primeira amostra deve ser sempre desconsiderada.

Para gerar os gráficos, foi usado o software GNUPlot (GNUPLOT, 1999). Ele lê valores inscritos em arquivos de texto, linha a linha, separados por espaço. Como a saída do comando "vmstat" na sua primeira possui os nomes dos campos do comando, o GNUPlot começou a fazer os gráficos a partir da segunda linha. Logo, todos eles começam a partir do valor "2" no eixo X. No eixo Y, os valores estão referenciados em porcentagem, como na saída do comando "vmstat".

Após os testes realizados, foram gerados os seguintes dados:

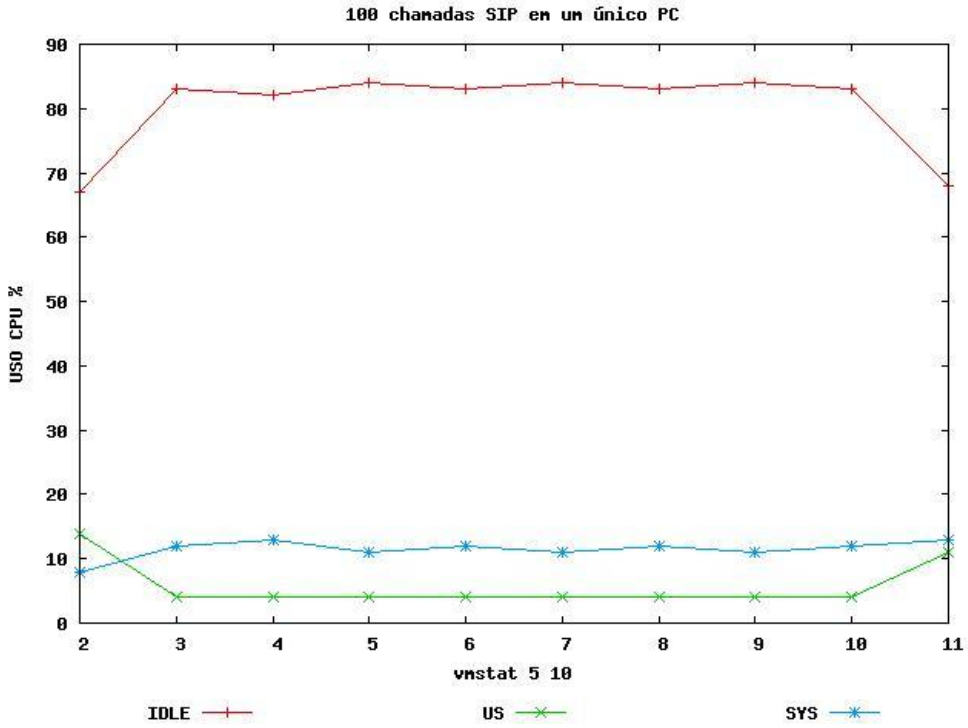

Figura 5: 100 chamadas SIP em um único Servidor

O parâmetro "IDLE"é o percentual livre do processador. O parâmetro "US"é o percentual de uso do processador com processos de usuários. O parâmetro "SYS"é o percentual de uso do processador com processos do sistema.

Ao analisar esse gráfico, percebe-se que o PABX1 foi capaz de suportar a carga imposta sem dificuldades, com um nível de processamento livre acima de 80%. Nessa condição, é possível ouvir nitidamente o som da URA, e ainda é possível ter outros serviços rodando sem comprometer a qualidade do sistema.

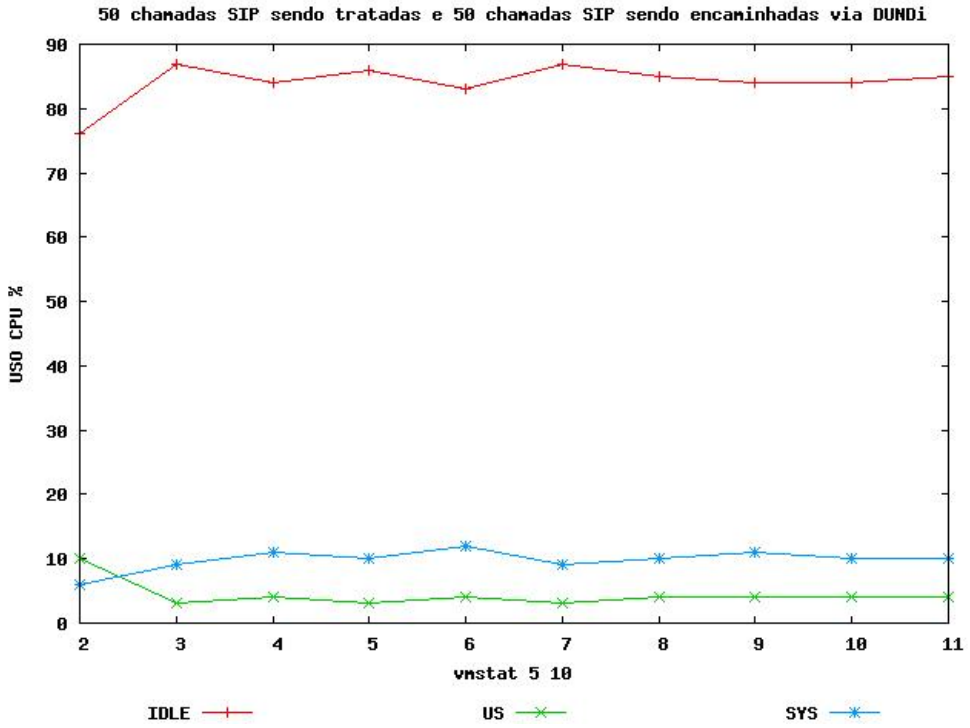

Figura 6: 50 chamadas SIP sendo tratadas e 50 sendo encaminhadas via DUNDi

Ao analisar esse gráfico, percebe-se que o PABX1 foi capaz de suportar a carga imposta sem dificuldades, com um nível de processamento livre acima de 80%. Nessa condição, é possível ouvir nitidamente o som da URA, e ainda é possível ter outros serviços rodando sem comprometer a qualidade do sistema.

Houve um aumento no nível de processamento livre de aproximadamente 2% nesse servidor, o que de forma alguma justificaria o uso de uma solução assim.

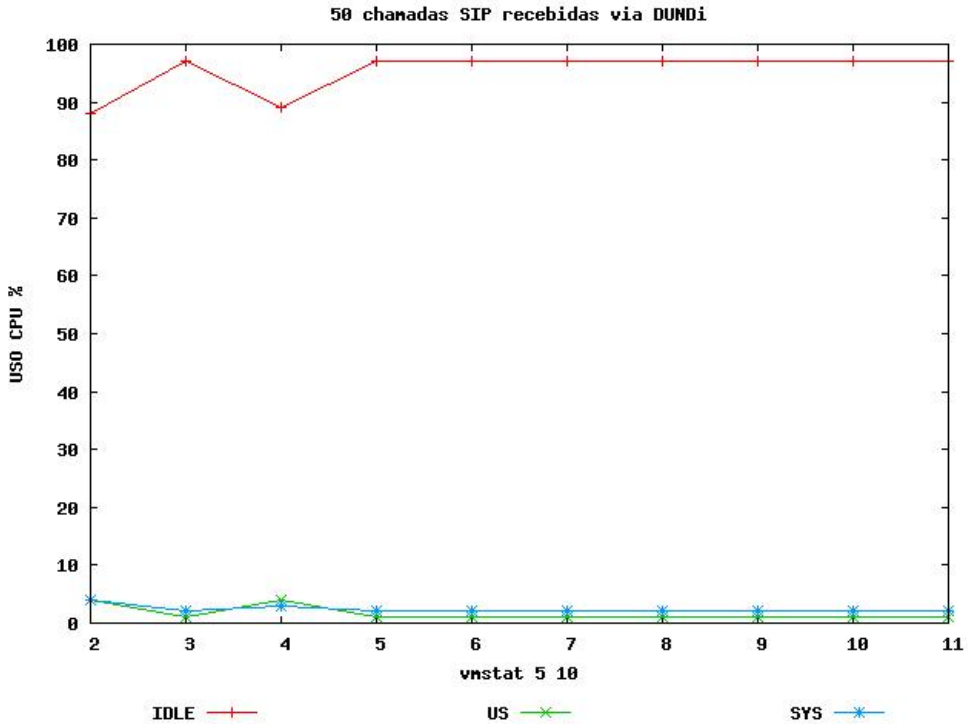

Figura 7: 50 chamadas SIP recebidas via DUNDi

Ao analisar esse gráfico, percebe-se que o PABX2 foi capaz de suportar a carga imposta sem dificuldades, com um nível de processamento livre acima de 90%. Nessa condição, é possível ouvir nitidamente o som da URA, e ainda é possível ter outros serviços rodando sem comprometer a qualidade do sistema.

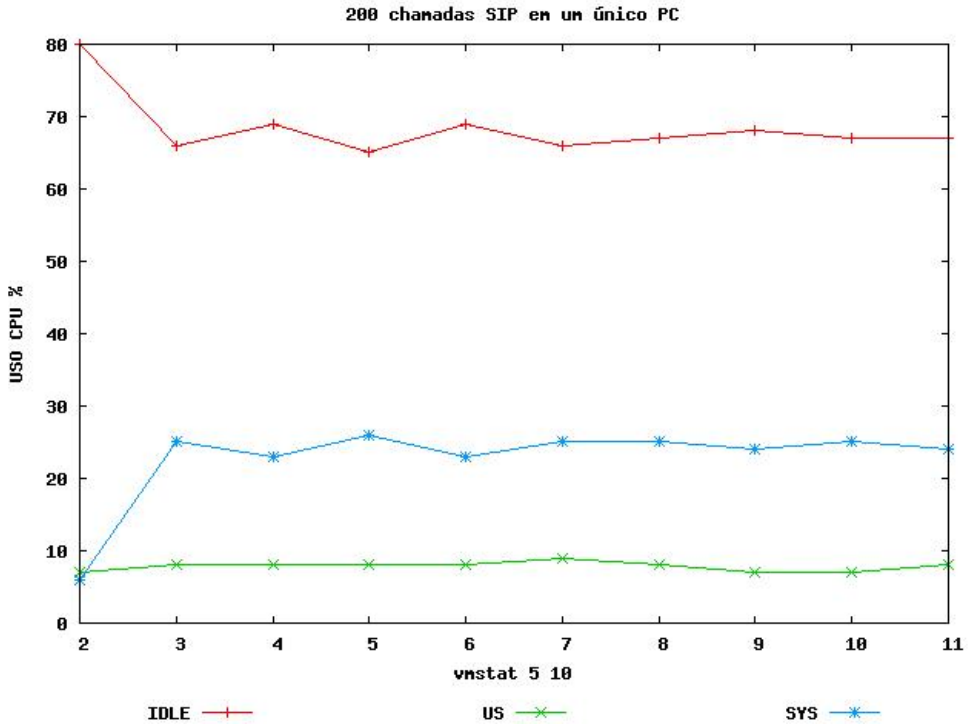

Figura 8: 200 chamadas SIP em um único Servidor

Ao analisar esse gráfico, percebe-se que o PABX1 foi capaz de suportar a carga imposta sem dificuldades, com um nível de processamento livre acima de 60%. Nessa condição, é possível ouvir nitidamente o som da URA, e ainda é possível ter outros serviços rodando sem comprometer a qualidade do sistema.

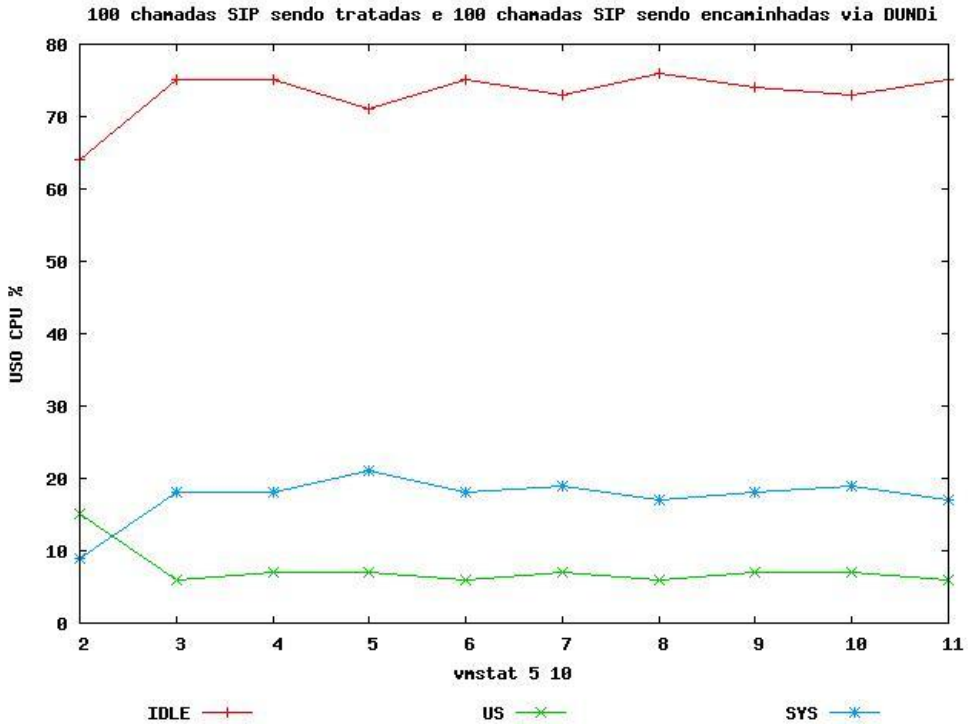

Figura 9: 100 chamadas SIP sendo tratadas e 100 sendo encaminhadas via DUNDi

Ao analisar esse gráfico, percebe-se que o PABX1 foi capaz de suportar a carga imposta sem dificuldades, com um nível de processamento livre acima de 70%. Nessa condição, é possível ouvir nitidamente o som da URA, e ainda é possível ter outros serviços rodando sem comprometer a qualidade do sistema.

Houve um aumento no nível de processamento livre de aproximadamente 6% nesse servidor, o que não justificaria o uso de uma solução assim.

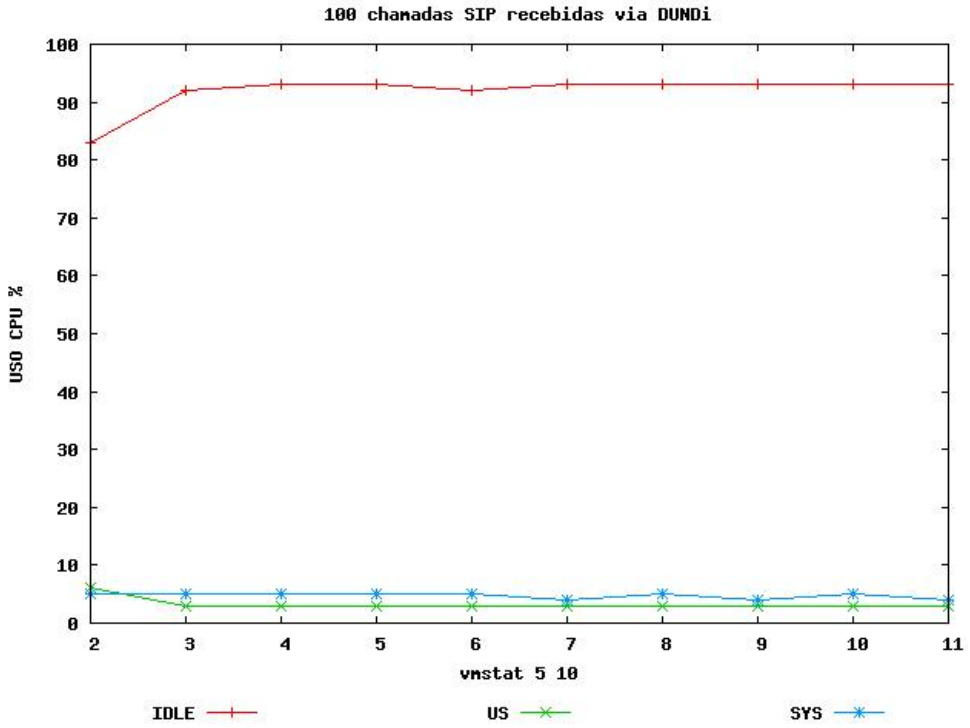

Figura 10: 100 chamadas SIP recebidas via DUNDi

Ao analisar esse gráfico, percebe-se que o PABX2 foi capaz de suportar a carga imposta sem dificuldades, com um nível de processamento livre acima de 90%. Nessa condição, é possível ouvir nitidamente o som da URA, e ainda é possível ter outros serviços rodando sem comprometer a qualidade do sistema.

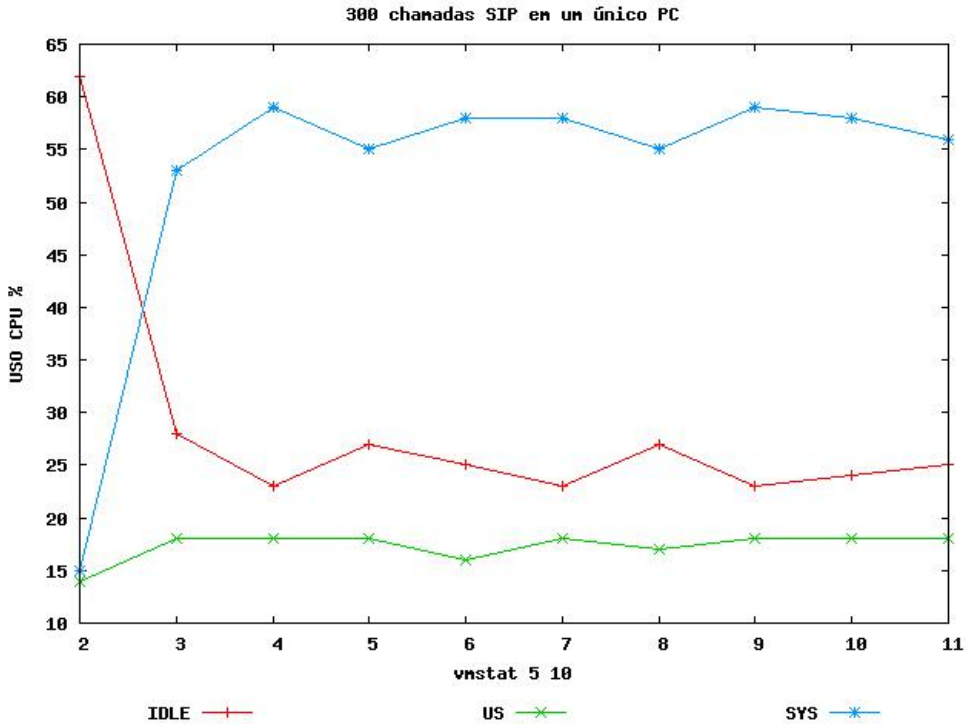

Figura 11: 300 chamadas SIP em um único Servidor

Ao analisar esse gráfico, percebe-se que o PABX1 foi capaz de suportar a carga imposta com grandes dificuldades, com um nível de processamento livre em torno de 25%. Nessa condição, não é possível ouvir nitidamente o som da URA, já que apesar de ter em média 25% de processamento livre, ocorrem alguns picos de processamento. Como o processador nesse caso não tem condições de processar tudo no instante do pico, acaba por fazer o áudio da URA sair com algumas falhas (picotamento).

Para o tráfego em questão, não é aconselhável usar esse servidor sozinho de forma alguma.

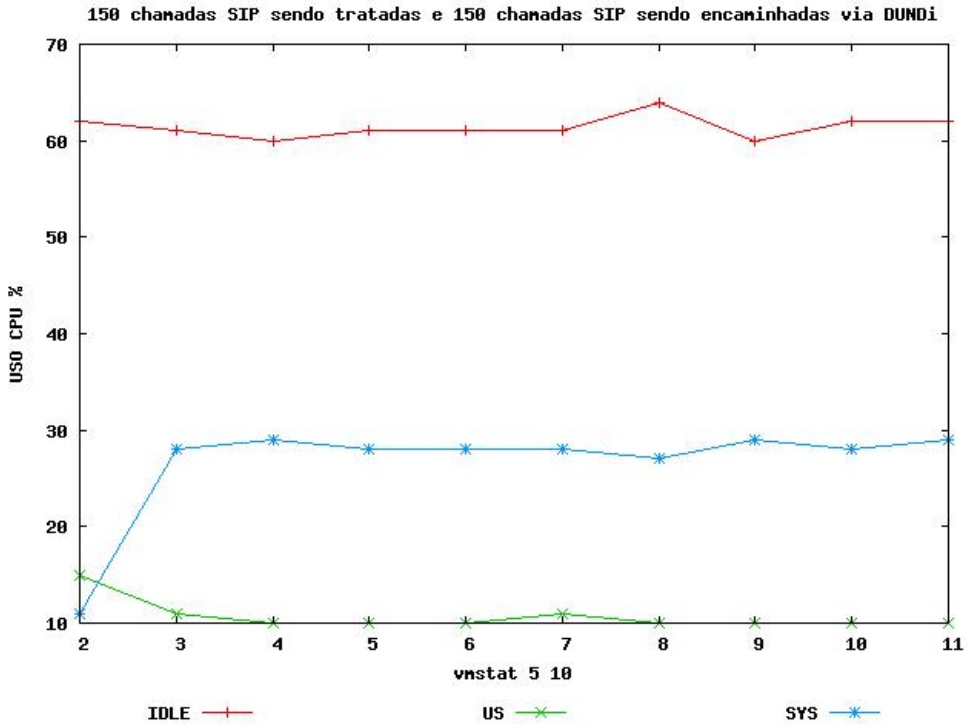

Figura 12: 150 chamadas SIP sendo tratadas e 150 sendo encaminhadas via DUNDi

Ao analisar esse gráfico, percebe-se que o PABX1 foi capaz de suportar a carga imposta sem dificuldades, com um nível de processamento livre em torno de 60%. Nessa condição, é possível ouvir nitidamente o som da URA, e ainda é possível ter outros serviços rodando sem comprometer a qualidade do sistema, desde que os outros serviços não consumissem muito processamento.

Houve um aumento no nível de processamento livre de aproximadamente 35% nesse servidor, além do fato de ser possível ouvir sem falhas o áudio da URA. Isso justificaria o uso de uma solução assim.

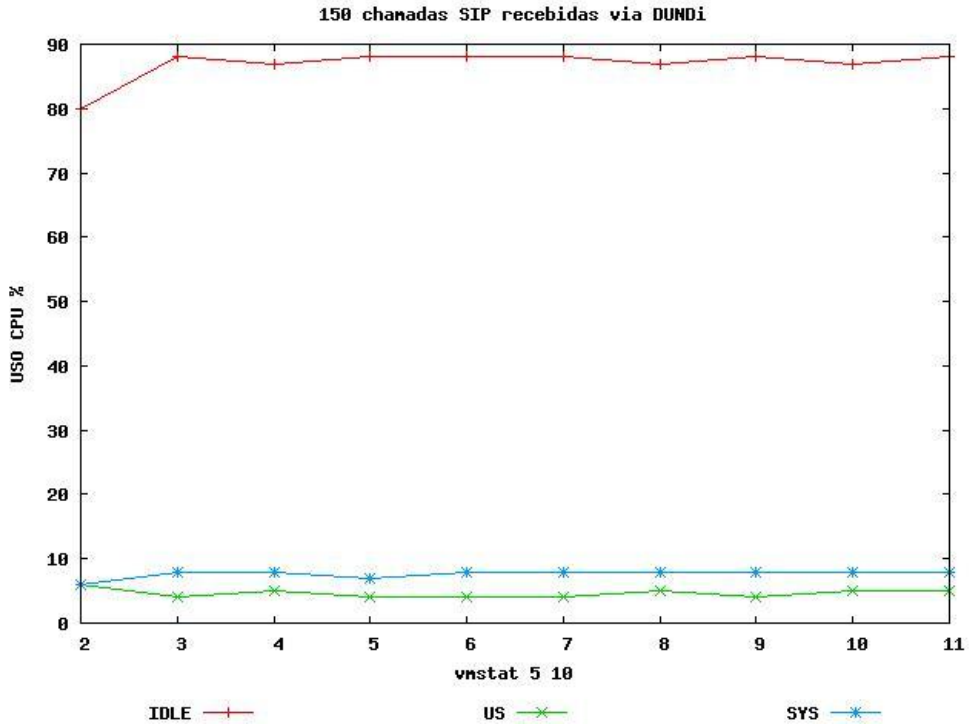

Figura 13: 150 chamadas SIP recebidas via DUNDi

Ao analisar esse gráfico, percebe-se que o PABX2 foi capaz de suportar a carga imposta sem dificuldades, com um nível de processamento livre próximo de 90%. Nessa condição, é possível ouvir nitidamente o som da URA, e ainda é possível ter outros serviços rodando sem comprometer a qualidade do sistema.

Observando os gráficos, percebe-se que quanto maior o volume de tráfego, maior o ganho de processamento com o DUNDi. Mas apesar dos servidores com o DUNDi estarem relativamente folgadas com 150 chamadas cada, um fator fez com que não se avançasse sobre os testes. O módulo "chan_iax", que implementa o IAX, limita o buffer de dados com tamanho suficiente de dados para 200 chamadas. Isso fez com que, ao testar com 200 ligações via DUNDi e usar mais um ramal somente para escutar a URA como forma de verificar a qualidade do áudio, o Asterisk exibisse um "warning"e o áudio não saísse de forma adequada. Abaixo segue o aviso:

```
Feb 18 16:14:25 WARNING[5430]: chan\_iax2.c:3806 iax2\_trunk\_queue:
Maximum trunk data space exceeded to 192.168.130.10:4569
```

Para contornar o problema, podem-se seguir as soluções apresentadas no link abaixo:

http://bugs.digium.com/view.php?id=8267

Outra solução para o caso seja necessário mais de 200 chamadas via DUNDi é configura-lo com o SIP. Perde-se o suporte a criptografia, mas se ganha na capacidade de ligações sem configurações adicionais.

5 Conclusões

As técnicas de Cluster estudadas nesse trabalho mostraram cada uma a sua eficiência.

O DNS Round-Robin foi capaz de distribuir os ramais entre os dois servidores configurados para responder ao domínio usado no teste. Como já é conhecido pelos seus adeptos, ele não faz por si só o failover, porém é uma solução muito simples de ser implementada.

O DUNDi tem um ganho mínimo quando o volume de ligações simultâneas é baixo (100 ligações).

Porém com o aumento do volume de tráfego (200 a 300 chamadas), o DUNDi se mostrou bastante eficiente, pois conseguiu melhorar significativamente o processamento.

Nas referências usadas para auxiliar a configuração do DUNDi, normalmente ele vem associado ao Asterisk Realtime, porém por ser uma solução muito complexa de implementar e não implementar sozinho nenhuma forma de Cluster, não foi integrada ao DUNDi.

Apesar de poucos, os materiais que ensinam a configurar o DUNDi são suficientes para realizar a sua implementação. Quando é feita uma nova instalação do Asterisk para configurar o DUNDi, a tarefa se torna um pouco simples. Porém quando se tem um sistema pronto que não foi concebido com o DUNDi, nesse caso o DISC-OS, a tarefa de implementar se torna mais complexa. Isso porque adaptar o DUNDi a um PABX já configurado exige o conhecimento do sistema, além de necessitar de cuidado para que não se percam as funcionalidades do PABX em função do uso do DUNDi.

ANEXO A – Anexos

A.1 Configuração do Bind 9 para permitir o uso do DNS Round-Robin

Para habilitar o DNS Round-Robin no Bind 9 (BIND, 2000), deve-se apenas colocar os endereços dos servidores e relaciona-los com o mesmo nome de servidor:

```
registro IN  A    192.168.130.102
registro IN  A    192.168.65.100
```

A.2 Instalação do DUNDi no DISC-OS

O DISC-OS não foi planejado para usar o DUNDi por padrão. Porém não há nada que impessa o DISC-OS de usar o DUNDi.

Para usar o DUNDi no DISC-OS deve-se carregar o seu módulo no arquivo "/etc/asterisk/modules.conf":

```
load => pbx_dundi.so
```

Além dele, usaremos mais dois módulos no sistema:

```
load => res_crypto.so
load => app_chanisavail.so
```

O módulo "res_crypto.so" é o responsável pelo suporte a criptografia no Asterisk. Já o módulo "app_chanisavail.so" permite o uso da aplicação "ChanIsAvail" do dialplain do Asterisk.

Após colocar os módulos no arquivo, reinicia-se o Asterisk para carregar os módulos. Opcionalmente, pode-se carregar os módulos usando o comando "load" no CLI do Asterisk. Porém ao reiniciar o asterisk, esta configuração será perdida.

Depois de carregados os módulos, é necessário criar o arquivo "/etc/asterisk/dundi.conf", pois o mesmo não vem por padrão no DISC-OS. O arquivo pode ser encontrado em:

http://www.voip-info.org/wiki/index.php?page=Asterisk+config+dundi.conf

O arquivo deve ter o dono "asterisk" e o grupo "asterisk".

Pode-se então configurar o entroncamento entre as centrais via interface do DISC. Neste caso, foi feito usando IAX.

Depois, edita-se o "/etc/asterisk/iax.conf" e adicionamos o seguinte parâmetro no tronco:

```
dbsecret=dundi/secret
```

O parâmetro "dbsecret" define a senha que será usada nas requisições para localização do ramal.

Então configura-se uma rota de entrada, uma rota de saída para o tronco e os ramais:

Depois de terminado o trabalho com a interface do DISC-OS, passa-se então para a configuração do DUNDi e configurações adicionais. Deve ser adicionado tanto no "sip.conf" como no "iax.conf" o seguinte parâmetro dentro do contexto "general":

```
regcontext=dundiextens
```

regcontext: contexto que será dinamicamente criado (se o mesmo não existir) e será incluso uma extensão com o comando "NoOp" quando um ramal se registrar no DISC-OS.

```
dundi.conf

[general]

entityid=00:13:72:02:F5:F6
cachetime=5
ttl=1
```

```
autokill=yes
secretpath=dundi

[mappings]

dundi => dundiextens,0,IAX2,dundi:${SECRET}@192.168.65.100/${NUMBER},nopartial

[00:13:72:02:F5:E3]
model = symmetric
host = 192.168.130.102
inkey = priv
outkey = priv
include = all
permit = all
qualify = yes
order = primary
```

Parâmetros:

entityid: Define o identificador do peer na rede DUNDi. O padrão é o endereço MAC da primeira interface de rede, mas é recomendável especificar esse valor manualmente.

cachetime: Define o tempo de validade de uma consulta DUNDi.

ttl: define o número de saltos que o pedido DUNDi pode ter antes de ser descartado.

autokill: Define o que acontece quando um ACK não é recebido depois de 2 segundos. Quando setado em "yes", ele cancela a requisição.

secretpath: Define a família da chave criada. O padrão é "dundi" sendo que a chave será mantida em "secret/dundi".

A linha de mapeamento segue a seguinte sintaxe:

```
dundi_context => local_context,weight,tech,dest[,options]
```

Parâmetros:

dundi_context: é o nome do contexto que será usado nas consultas DUNDi.

local_context: é o nome do contexto onde o servidor local buscará o número para responder sobre a existência ou não desse número.

weight: é o peso para a resposta da requisição. Quanto menor o número, maior a prioridade da sua resposta. Podem ser usados valores de 0 a 60000.

tech: indica a tecnologia do ramal a ser buscado. Pode ser SIP, IAX2 ou H323.

dest: define os valores para que o destino encontre o número desejado. Três valores são repassados:

$NUMBER: O número requisitado

$IPADDR: O endereço IP a se conectar

$SECRET: A senha que está sendo usada no momento.

Algumas opções também podem ser selecionadas. No nosso caso, foi usada a opção "nopartial", que faz com que as respostas indiquem somente o valor exato que foi pedido, sem pesquisas parciais.

Por último, vem a identificação do outro peer. O primeiro valor é o entityid do outro peer. Os outros parâmetros são:

model: Indica se a conexão DUNDi será só de envio, recebimento ou de envio e recebimento.

host: É o endereço IP do outro peer.

inkey: É a chave que o outro peer usará para autenticar com este peer.

outkey: É a chave que será usada por este peer para autenticar com o outro peer.

include: Define a permissão que o outro peer terá de fazer consultas em um contexto particular deste peer. O valor "all" permite que o outro peer faça consultas em todos os contextos.

permit: Define a permissão que o outro peer terá de fazer consultas no contexto DUNDi deste peer. O valor "all" permite que o outro peer faça consultas em todos os contextos.

qualify: Tem a mesma função do autokill, porém é especificado na própria identificação do peer.

order: Define a ordem da pesquisa. Pode ser primário, secundário, terceário e quaternário.

Após configurar o DUNDi, passa-se para as configurações do plano de discagem:

extensions_disc.conf:

```
[disc-ext-local]
exten => 2002,1,ChanIsAvail(SIP/2002|sj)
exten => 2002,2,Goto(disc-ext-local,2002,4)
exten => 2002,3,Hangup
exten => 2002,102,Goto(lookupdundi,2002,1)
exten => 2002,103,Hangup
exten => 2002,4,Macro(disc-exten-vm,novm,2002)
exten => 2002,5,Hangup

[disc-inrt-from_taho]
exten => _[*#0-9].,1,Set(FROM_DID=${EXTEN})
exten => _[*#0-9].,2,Set(CDR(userfield)=from_taho)
exten => _[*#0-9].,3,Macro(disc-blacklist,,${CALLERID(NUM)},in)
exten => _[*#0-9].,4,Goto(disc-ext-local,${FROM_DID},1)
```

extensions_local.conf:

```
[lookupdundi]
switch => DUNDi/dundi
```

No contexto local do DISC-OS é usada a aplicação "ChanIsAvail" para verificar se o ramal está registrado neste servidor ou não. Se estiver, ele é direcionado para o dialplain do DISC-OS para que seja realizada a chamada. Caso o ramal não esteja registrado, ele é enviado para o contexto "lookupdundi", que será responsável por fazer a consulta DUNDi. Na rota entrante, foi feita uma alteração para que as chamadas vindas do tronco usado para o DUNDi sejam direcionadas para o contexto local do DISC-OS.

Lógica do plano de discagem:

```
[disc-ext-local]
exten => 2002,1,ChanIsAvail(SIP/2002|sj)
```

Testa se o ramal (2002 SIP) está registrado e disponível para receber uma chamada. Caso esteja ativo, irá para a extensão n+1 (2). Se não estiver, irá para a extensão n+101 (102).

```
exten => 2002,2,Goto(disc-ext-local,2002,4)
```

A chamada é redirecionada para a prioridade "4" desse mesmo contexto.

```
exten => 2002,3,Hangup
```

Desliga a chamada.

```
exten => 2002,102,Goto(lookupdundi,2002,1)
```

A chamada é redirecionada para o contexto "lookupdundi" procurando pelo ramal "2002" na prioridade "1" desse contexto.

```
exten => 2002,103,Hangup
```

Desliga a chamada.

```
exten => 2002,4,Macro(disc-exten-vm,novm,2002)
```

É chamada a macro "disc-exten-vm" informando que o mesmo não possuí voicemail habilitado e que se trata da extensão "2002". Essa é a primeira sequência original do DISC-OS.

```
exten => 2002,5,Hangup
```

Desliga a chamada.

```
[disc-inrt-from_taho]
exten => _[*#0-9].,1,Set(FROM_DID=${EXTEN})
```

Seta a variável "FROM_DID" com o valor da variável "EXTEN".

```
exten => _[*#0-9].,2,Set(CDR(userfield)=from_taho)
```

Seta a variável "CDR(userfield)" com o nome da rota de entrada.

```
exten => _[*#0-9].,3,Macro(disc-blacklist,,${CALLERID(NUM)},in)
```

Chama a macro "disc-blacklist" para testar se essa ligação não deve ser bloqueada.

```
exten => _[*#0-9].,4,Goto(disc-ext-local,${FROM_DID},1)
```

Envia a chamada para o contexto local, no número discado na prioridade "1".

```
[lookupdundi]
switch => DUNDi/dundi
```

A partir daqui o DUNDi se encarrega de fazer a consulta no contexto de consultas.

Após essas configurações, reiniciamos o Asterisk, de forma a faze-lo reconhecer as alterações realizadas.

A.2.1 Comandos para Testes

Alguns comandos do CLI do Asterisk podem ser usados para verificar o funcionamento das configurações realizadas:

dundi show peers: Mostra o estado das conexões com os peers.

Ex:

```
discOS2*CLI> dundi show peers
EID                              Host                    Model            AvgTime  Sta
00:13:72:02:f5:e3    192.168.130.102  (S) Symmetric  Unavail     OK (1 ms)
1 dundi peers [1 online, 0 offline, 0 unmonitored]
discOS2*CLI>
```

dundi lookup extensão (Consulta o outro peer se a extensão extensão está registrada nele)

Ex:

```
discOS2*CLI> dundi lookup 2010@dundi
  1.     0 IAX2/dundi:SGUBCZR+0GPV9MkTuQwsHA@192.168.130.102/2010 (EXISTS)
```

```
     from 00:13:72:02:f5:e3, expires in 5 s
DUNDi lookup completed in 15 ms
discOS2*CLI>
```

A.3 Saída dos comandos vmstat e sipp

procs		memory				swap		io		system		cpu			
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa
0	0	0	432588	5268	38416	0	0	592	59	1699	1194	14	8	67	11
0	0	0	432268	5308	38692	0	0	60	22	2118	5787	4	12	83	1
0	0	0	432268	5348	38712	0	0	7	22	2080	5784	4	13	82	1
0	0	0	432012	5380	38864	0	0	34	72	2073	5797	4	11	84	1
0	0	0	431820	5428	39004	0	0	31	61	2056	5790	4	12	83	1
0	0	0	431756	5464	39032	0	0	10	20	2046	5781	4	11	84	1
15	0	0	431564	5496	39184	0	0	34	26	2048	5798	4	12	83	1
0	0	0	431500	5544	39204	0	0	9	22	2044	5795	4	11	84	1
0	0	0	431308	5588	39352	0	0	34	22	2052	5780	4	12	83	1
0	0	0	429116	5740	41236	0	0	389	75	2076	5903	11	13	68	8

Figura 14: Saída do comando “vmstat 5 10” para 100 chamadas SIP em um único PC

```
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5061      64.56 s          100  192.168.130.10:5060(UDP)

  0 new calls during 0.548 s period      4 ms scheduler resolution
  100 calls (limit 100)                  Peak was 100 calls, after 10 s
  0 Running, 100 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets

                                 Messages  Retrans   Timeout   Unexpected-Msg
     INVITE ---------->          100       0         0
        100 <----------          100       0         0         0
        180 <----------          0         0         0         0
        183 <----------          0         0         0         0
        200 <----------  E-RTD1  100       0         0         0
        ACK ---------->          100       0
      Pause [     1:40]          100                           0
        BYE ---------->          0         0         0
        200 <----------          0         0         0         0

------------------------------ Test Terminated --------------------------------
```

```
---------------------------- Statistics Screen ------- [1-9]: Change Screen --
 Start Time              | 2008-02-15   15:17:35:370    1203095855.370580
 Last Reset Time         | 2008-02-15   15:18:50:421    1203095930.421470
 Current Time            | 2008-02-15   15:18:51:230    1203095931.230711
-------------------------+----------------------------+----------------------------
 Counter Name            | Periodic value             | Cumulative value
-------------------------+----------------------------+----------------------------
 Elapsed Time            | 00:00:00:809               | 00:01:15:860
 Call Rate               |    0.000 cps               |    0.659 cps
-------------------------+----------------------------+----------------------------
 Incoming call created   |        0                   |        0
 OutGoing call created   |        0                   |       50
 Total Call created      |                            |       50
 Current Call            |       50                   |
-------------------------+----------------------------+----------------------------
 Successful call         |        0                   |        0
 Failed call             |        0                   |        0
-------------------------+----------------------------+----------------------------
 Response Time 1         | 00:00:00:000               | 00:00:00:001
 Call Length             | 00:00:00:000               | 00:00:00:000
---------------------------- Test Terminated --------------------------------
```

Figura 15: Saída de status do SIPP para 100 chamadas SIP em um único PC

```
procs -----------memory---------- ---swap-- -----io---- --system-- ----cpu----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in    cs us sy id wa
 0  0      0 430436   5520  38472    0    0   404    55 1821   815 10  6 76  8
 0  0      0 430116   5568  38596    0    0    30    24 2352  4503  3  9 87  1
 1  0      0 429860   5612  38872    0    0    59    24 2390  4540  4 11 84  1
 0  0      0 429732   5652  38896    0    0     8    22 2361  4506  3 10 86  1
 0  0      0 429540   5680  39044    0    0    34    10 2410  4555  4 12 83  1
 0  0      0 429476   5720  39064    0    0     7    78 2363  4504  3  9 87  1
 0  0      0 429284   5756  39212    0    0    34    17 2392  4591  4 10 85  1
 0  0      0 429092   5796  39360    0    0    33    26 2355  4500  4 11 84  1
 0  0      0 429028   5836  39380    0    0     7    22 2401  4581  4 10 84  1
 0  0      0 428772   5872  39532    0    0    33    22 2348  4492  4 10 85  1
```

Figura 16: Saída do comando “vmstat 5 10” para 50 chamadas SIP recebidas e 50 encaminhadas via DUNDi

```
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5061      75.85 s           50  192.168.130.10:5060(UDP)

  0 new calls during 0.808 s period      4 ms scheduler resolution
  50 calls (limit 50)                    Peak was 50 calls, after 5 s
  0 Running, 50 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets

                                 Messages  Retrans   Timeout   Unexpected-Msg
      INVITE ---------->         50        0         0
        100 <----------          50        0         0         0
        180 <----------          0         0         0         0
        183 <----------          0         0         0         0
        200 <----------  E-RTD1  50        0         0         0
        ACK ---------->          50        0
      Pause [     1:40]          50                            0
        BYE ---------->          0         0         0
        200 <----------          0         0         0         0

------------------------------ Test Terminated --------------------------------


------------------------------ Statistics Screen ------- [1-9]: Change Screen --
  Start Time              | 2008-02-15   15:17:35:370    1203095855.370580
  Last Reset Time         | 2008-02-15   15:18:50:421    1203095930.421470
  Current Time            | 2008-02-15   15:18:51:230    1203095931.230711
--------------------------+---------------------------+---------------------------
  Counter Name            | Periodic value            | Cumulative value
--------------------------+---------------------------+---------------------------
  Elapsed Time            | 00:00:00:809              | 00:01:15:860
  Call Rate               |    0.000 cps              |    0.659 cps
--------------------------+---------------------------+---------------------------
  Incoming call created   |        0                  |        0
  OutGoing call created   |        0                  |       50
  Total Call created      |                           |       50
- Current Call            |       50                  |
--------------------------+---------------------------+---------------------------
- Successful call         |        0                  |        0
  Failed call             |        0                  |        0
--------------------------+---------------------------+---------------------------
- Response Time 1         | 00:00:00:000              | 00:00:00:001
  Call Length             | 00:00:00:000              | 00:00:00:000
------------------------------ Test Terminated --------------------------------
```

Figura 17: Saída de status do SIPP para 50 chamadas SIP recebidas e 50 encaminhadas via DUNDi

```
procs -----------memory---------- ---swap-- -----io---- --system-- ----cpu----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in    cs us sy id wa
 0  0      0 428408   4244  37356    0    0   201    20  912   426  4  4 88  5
 0  0      0 428152   4264  37596    0    0    47    10 2198  3634  1  2 97  0
 0  0      0 425512   4360  39580    0    0   374     2 2220  3757  4  3 89  4
 0  0      0 425576   4384  39556    0    0     0    74 2221  3647  1  2 97  0
 0  0      0 425512   4384  39816    0    0    26     0 2210  3628  1  2 97  0
 0  0      0 425464   4396  39804    0    0    26    74 2221  3626  1  2 97  0
 0  0      0 425528   4404  39796    0    0     0     2 2204  3632  1  2 97  0
 0  0      0 425528   4416  39784    0    0    26     4 2216  3644  1  2 97  0
 0  0      0 425528   4424  39776    0    0     0     2 2213  3635  1  2 97  0
 0  0      0 425400   4432  40028    0    0    26     6 2203  3624  1  2 97  0
```

Figura 18: Saída do commando “vmstat 5 10” para 50 chamadas recebidas via DUNDi

```
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5062      75.26 s           50  192.168.130.10:5060(UDP)

  0 new calls during 0.240 s period      3 ms scheduler resolution
  50 calls (limit 50)                    Peak was 50 calls, after 5 s
  0 Running, 50 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets

                                 Messages  Retrans   Timeout   Unexpected-Msg
     INVITE ---------->          50        0         0
        100 <----------          50        0         0         0
        180 <----------          0         0         0         0
        183 <----------          0         0         0         0
        200 <----------  E-RTD1  50        0         0         0
        ACK ---------->          50        0
      Pause [     1:40]          50                            0
        BYE ---------->          0         0         0
        200 <----------          0         0         0         0

------------------------------ Test Terminated --------------------------------


------------------------------ Statistics Screen ------- [1-9]: Change Screen --
  Start Time              | 2008-02-15   15:17:36:829    1203095856.829373
  Last Reset Time         | 2008-02-15   15:18:51:857    1203095931.857406
  Current Time            | 2008-02-15   15:18:52:102    1203095932.102370
--------------------------+---------------------------+--------------------------
  Counter Name            | Periodic value            | Cumulative value
--------------------------+---------------------------+--------------------------
  Elapsed Time            | 00:00:00:244              | 00:01:15:272
  Call Rate               |    0.000 cps              |    0.664 cps
--------------------------+---------------------------+--------------------------
  Incoming call created   |        0                  |        0
  OutGoing call created   |        0                  |       50
  Total Call created      |                           |       50
  Current Call            |       50                  |
--------------------------+---------------------------+--------------------------
  Successful call         |        0                  |        0
  Failed call             |        0                  |        0
--------------------------+---------------------------+--------------------------
  Response Time 1         | 00:00:00:000              | 00:00:00:011
  Call Length             | 00:00:00:000              | 00:00:00:000
------------------------------ Test Terminated --------------------------------
```

Figura 19: Saída de status do SIPP para 50 chamadas recebidas via DUNDi

```
procs -----------memory---------- ---swap-- -----io---- --system-- ----cpu----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in    cs us sy id wa
 0  0      0 419012   5912  40776    0    0   299    42 1887  1696  7  6 80  6
 0  0      0 418820   5952  40924    0    0    33   162 2081 11445  8 25 66  1
 0  0      0 418628   5996  41064    0    0    32    22 2050 11452  8 23 69  1
 0  0      0 418500   6028  41096    0    0    10     9 2047 11446  8 26 65  1
 0  0      0 418372   6064  41244    0    0    32    22 2055 11450  8 23 69  0
 0  0      0 418180   6088  41268    0    0     6    33 2054 11450  9 25 66  1
 0  0      0 418052   6124  41416    0    0    34    29 2062 11447  8 25 67  1
 0  0      0 417860   6168  41568    0    0    34    22 2058 11453  7 24 68  1
 0  0      0 417732   6204  41588    0    0     6    23 2051 11451  7 25 67  1
 0  0      0 417604   6248  41740    0    0    34    25 2074 11451  8 24 67  1
```

Figura 20: Saída do comando “vmstat 5 10” para 200 chamadas SIP em um único PC

```
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5061      75.32 s          200  192.168.130.10:5060(UDP)

  0 new calls during 0.304 s period      4 ms scheduler resolution
  200 calls (limit 200)                  Peak was 200 calls, after 20 s
  0 Running, 200 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets

                                 Messages  Retrans   Timeout   Unexpected-Msg
     INVITE ---------->          200       0         0
        100 <----------          200       0         0         0
        180 <----------          0         0         0         0
        183 <----------          0         0         0         0
        200 <----------  E-RTD1  200       0         0         0
        ACK ---------->          200       0
      Pause [     1:40]          200                           0
        BYE ---------->          0         0         0
        200 <----------          0         0         0         0

------------------------------ Test Terminated --------------------------------


------------------------------ Statistics Screen ------- [1-9]: Change Screen --
  Start Time                | 2008-02-18   15:49:58:986    1203360598.986311
  Last Reset Time           | 2008-02-18   15:51:14:005    1203360674.005862
  Current Time              | 2008-02-18   15:51:14:312    1203360674.312152
--------------------------+---------------------------+--------------------------
  Counter Name              | Periodic value            | Cumulative value
--------------------------+---------------------------+--------------------------
  Elapsed Time              | 00:00:00:306              | 00:01:15:325
  Call Rate                 |    0.000 cps              |    2.655 cps
--------------------------+---------------------------+--------------------------
  Incoming call created     |        0                  |        0
  OutGoing call created     |        0                  |      200
  Total Call created        |                           |      200
  Current Call              |      200                  |
--------------------------+---------------------------+--------------------------
  Successful call           |        0                  |        0
  Failed call               |        0                  |        0
--------------------------+---------------------------+--------------------------
  Response Time 1           | 00:00:00:000              | 00:00:00:002
  Call Length               | 00:00:00:000              | 00:00:00:000
------------------------------ Test Terminated --------------------------------
```

Figura 21: Saída de status do SIPP para 200 chamadas SIP em um único PC

procs		memory				swap		io		system		cpu			
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa
0	0	0	416740	5416	40096	0	0	638	70	1734	1399	15	9	64	12
0	0	0	416100	5464	40372	0	0	60	24	2667	7288	6	18	75	1
0	0	0	416036	5484	40392	0	0	6	6	2677	7302	7	18	75	1
0	0	0	415844	5528	40536	0	0	33	114	2717	7305	7	21	71	1
0	0	0	415780	5564	40556	0	0	6	25	2692	7303	6	18	75	1
0	0	0	415524	5608	40708	0	0	34	29	2681	7312	7	19	73	1
0	0	0	415396	5652	40856	0	0	34	22	2679	7337	6	17	76	1
0	0	0	415332	5692	40876	0	0	7	26	2697	7312	7	18	74	1
0	0	0	415076	5736	41024	0	0	34	22	2693	7312	7	19	73	1
0	0	0	414884	5764	41164	0	0	32	2	2671	7299	6	17	75	1

Figura 22: Saída do comando “vmstat 5 10” para 100 chamadas SIP recebidas e 100 encaminhadas via DUNDi

```
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5061      59.96 s          100  192.168.130.10:5060(UDP)

  0 new calls during 0.947 s period      4 ms scheduler resolution
  100 calls (limit 100)                  Peak was 100 calls, after 10 s
  0 Running, 100 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets

                                 Messages  Retrans   Timeout   Unexpected-Msg
     INVITE ---------->          100       0         0
        100 <----------          100       0         0         0
        180 <----------          0         0         0         0
        183 <----------          0         0         0         0
        200 <----------  E-RTD1  100       0         0         0
        ACK ---------->          100       0
      Pause [     1:40]          100                           0
        BYE ---------->          0         0         0
        200 <----------          0         0         0         0

------------------------------ Test Terminated --------------------------------


------------------------------ Statistics Screen ------- [1-9]: Change Screen --
  Start Time              | 2008-02-18   16:04:44:018    1203361484.018009
  Last Reset Time         | 2008-02-18   16:05:43:036    1203361543.036017
  Current Time            | 2008-02-18   16:05:43:986    1203361543.986691
--------------------------+---------------------------+--------------------------
  Counter Name            | Periodic value            | Cumulative value
--------------------------+---------------------------+--------------------------
  Elapsed Time            | 00:00:00:950              | 00:00:59:968
  Call Rate               |    0.000 cps              |    1.668 cps
--------------------------+---------------------------+--------------------------
  Incoming call created   |        0                  |        0
  OutGoing call created   |        0                  |      100
  Total Call created      |                           |      100
  Current Call            |      100                  |
--------------------------+---------------------------+--------------------------
  Successful call         |        0                  |        0
  Failed call             |        0                  |        0
--------------------------+---------------------------+--------------------------
  Response Time 1         | 00:00:00:000              | 00:00:00:004
  Call Length             | 00:00:00:000              | 00:00:00:000
------------------------------ Test Terminated --------------------------------
```

Figura 23: Saída de status do SIPP para 100 chamadas SIP recebidas e 100 encaminhadas via DUNDi

procs		memory				swap		io		system		cpu			
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa
0	0	0	421344	4184	37936	0	0	284	27	880	793	6	5	83	6
0	0	0	420832	4220	37900	0	0	34	31	2469	7004	3	5	92	1
0	0	0	420832	4236	37884	0	0	0	16	2512	7077	3	5	93	0
0	0	0	420768	4244	38136	0	0	26	67	2506	7073	3	5	93	0
0	0	0	420704	4256	38384	0	0	26	5	2486	7060	3	5	92	0
0	0	0	420704	4264	38376	0	0	0	2	2481	7040	3	4	93	0
0	0	0	420576	4280	38360	0	0	26	5	2495	7063	3	5	93	0
0	0	0	420448	4288	38352	0	0	26	2	2492	7094	3	4	93	0
0	0	0	420448	4300	38340	0	0	0	3	2480	7077	3	5	93	0
1	0	0	420384	4308	38592	0	0	26	2	2476	7018	3	4	93	0

Figura 24: Saída do commando “vmstat 5 10” para 100 chamadas recebidas via DUNDi

```
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5061      59.96 s          100  192.168.130.10:5060(UDP)

  0 new calls during 0.947 s period      4 ms scheduler resolution
  100 calls (limit 100)                  Peak was 100 calls, after 10 s
  0 Running, 100 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5062      59.56 s          100  192.168.130.10:5060(UDP)

  0 new calls during 0.540 s period      3 ms scheduler resolution
  100 calls (limit 100)                  Peak was 100 calls, after 10 s
  0 Running, 100 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets

                                Messages  Retrans   Timeout   Unexpected-Msg
     INVITE ---------->         100       0         0
        100 <----------         100       0         0         0
        180 <----------         0         0         0         0
        183 <----------         0         0         0         0
        200 <----------  E-RTD1 100       0         0         0
        ACK ---------->         100       0
      Pause [     1:40]         100                           0
        BYE ---------->         0         0         0
        200 <----------         0         0         0         0

------------------------------ Test Terminated --------------------------------


------------------------------ Statistics Screen ------- [1-9]: Change Screen --
  Start Time              | 2008-02-18   16:04:46:093    1203361486.093572
  Last Reset Time         | 2008-02-18   16:05:45:119    1203361545.119849
  Current Time            | 2008-02-18   16:05:45:663    1203361545.663721
-------------------------+---------------------------+--------------------------
  Counter Name            | Periodic value            | Cumulative value
-------------------------+---------------------------+--------------------------
  Elapsed Time            | 00:00:00:543              | 00:00:59:570
  Call Rate               |    0.000 cps              |    1.679 cps
-------------------------+---------------------------+--------------------------
  Incoming call created   |        0                  |        0
  OutGoing call created   |        0                  |      100
```

Figura 25: Saída de status do SIPP para 100 chamadas recebidas via DUNDi

```
procs -----------memory---------- ---swap-- -----io---- --system-- ----cpu----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in    cs us sy id wa
 0  0      0 409100   5532  40608    0    0   499    60 1781  3160 14 15 62  9
 0  0      0 408908   5564  40760    0    0    34    31 2051 15335 18 53 28  1
 0  0      0 408652   5600  40900    0    0    30    34 2044 15327 18 59 23  0
 0  0      0 408652   5644  40924    0    0     9    22 2057 15393 18 55 27  1
 0  0      0 408460   5684  41072    0    0    34     9 2051 15334 16 58 25  1
 1  0      0 408204   5716  41196    0    0    28    16 2096 15434 18 58 23  1
 0  0      0 408076   5764  41244    0    0    14    24 2162 15586 17 55 27  1
 0  0      0 407812   5808  41400    0    0    34    26 2156 15646 18 59 23  0
 0  0      0 407684   5848  41424    0    0     8    47 2156 15581 18 58 24  1
 0  0      0 407492   5900  41568    0    0    33    26 2145 15588 18 56 25  0
```

Figura 26: Saída do comando “vmstat 5 10” para 300 chamadas SIP em um único PC

```
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5061      59.96 s          100  192.168.130.10:5060(UDP)

  0 new calls during 0.947 s period      4 ms scheduler resolution
  100 calls (limit 100)                  Peak was 100 calls, after 10 s
  0 Running, 100 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5061      80.54 s          300  192.168.130.10:5060(UDP)

  0 new calls during 0.528 s period      3 ms scheduler resolution
  300 calls (limit 300)                  Peak was 300 calls, after 30 s
  0 Running, 300 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets

                                 Messages  Retrans   Timeout   Unexpected-Msg
     INVITE ---------->          300       0         0
        100 <----------          300       0         0         0
        180 <----------          0         0         0         0
        183 <----------          0         0         0         0
        200 <----------  E-RTD1  300       0         0         0
        ACK ---------->          300       0
      Pause [     1:40]          300                           0
        BYE ---------->          0         0         0
        200 <----------          0         0         0         0

------------------------------ Test Terminated --------------------------------


---------------------------- Statistics Screen ------- [1-9]: Change Screen --
  Start Time              | 2008-02-18   16:09:38:516    1203361778.516353
  Last Reset Time         | 2008-02-18   16:10:58:536    1203361858.536366
  Current Time            | 2008-02-18   16:10:59:066    1203361859.066782
--------------------------+---------------------------+--------------------------
  Counter Name            | Periodic value            | Cumulative value
--------------------------+---------------------------+--------------------------
  Elapsed Time            | 00:00:00:530              | 00:01:20:550
  Call Rate               |    0.000 cps              |    3.724 cps
--------------------------+---------------------------+--------------------------
  Incoming call created   |        0                  |        0
  OutGoing call created   |        0                  |      300
  Total Call created      |                           |      300
```

Figura 27: Saída de status do SIPP para 300 chamadas SIP em um único PC

```
procs -----------memory---------- ---swap-- -----io---- --system-- ----cpu----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in    cs us sy id wa
 0  0      0 408412   5312  38488    0    0   618    65 1772  1850 15 11 62 12
 0  0      0 405428   5348  38620    0    0    27    31 2875  9727 11 28 61  0
 1  0      0 405364   5372  38756    0    0    29    16 2896  9645 10 29 60  0
 0  0      0 405364   5404  38764    0    0     3    21 2912  9612 10 28 61  0
 0  1      0 404988   5440  38928    0    0    35    60 2916  9624 10 28 61  1
 0  0      0 404988   5464  39064    0    0    28    10 2912  9650 11 28 61  0
 0  0      0 404924   5492  39068    0    0     2    26 2903  9642 10 27 64  0
 0  0      0 404668   5516  39204    0    0    29    16 2934  9628 10 29 60  1
 0  0      0 404476   5544  39208    0    0     2    19 2939  9606 10 28 62  0
 0  0      0 404348   5552  39340    0    0    26     3 2903  9576 10 29 62  0
```

Figura 28: Saída do comando "vmstat 5 10" para 150 chamadas SIP recebidas e 150 encaminhadas via DUNDi

```
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5061      59.96 s          100  192.168.130.10:5060(UDP)

  0 new calls during 0.947 s period      4 ms scheduler resolution
  100 calls (limit 100)                  Peak was 100 calls, after 10 s
  0 Running, 100 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5061      63.38 s          150  192.168.130.10:5060(UDP)

  0 new calls during 0.376 s period      4 ms scheduler resolution
  150 calls (limit 150)                  Peak was 150 calls, after 15 s
  0 Running, 150 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets

                                 Messages  Retrans   Timeout   Unexpected-Msg
     INVITE ---------->          150       0         0
        100 <----------          150       0         0         0
        180 <----------          0         0         0         0
        183 <----------          0         0         0         0
        200 <----------  E-RTD1  150       0         0         0
        ACK ---------->          150       0
      Pause [     1:40]          150                           0
        BYE ---------->          0         0         0
        200 <----------          0         0         0         0

------------------------------ Test Terminated --------------------------------


------------------------------ Statistics Screen ------- [1-9]: Change Screen --
  Start Time              | 2008-02-18   16:13:12:393    1203361992.393318
  Last Reset Time         | 2008-02-18   16:14:15:408    1203362055.408149
  Current Time            | 2008-02-18   16:14:15:787    1203362055.787414
--------------------------+---------------------------+--------------------------
  Counter Name            | Periodic value            | Cumulative value
--------------------------+---------------------------+--------------------------
  Elapsed Time            | 00:00:00:379              | 00:01:03:394
  Call Rate               |    0.000 cps              |    2.366 cps
--------------------------+---------------------------+--------------------------
  Incoming call created   |        0                  |        0
  OutGoing call created   |        0                  |      150
  Total Call created      |                           |      150
```

Figura 29: Saída de status do SIPP para 150 chamadas SIP recebidas e 150 encaminhadas via DUNDi

```
procs -----------memory---------- ---swap-- -----io---- --system-- ----cpu----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in    cs us sy id wa
 0  0      0 412192   4312  39888    0    0   302    32  901  1171  6  6 80  7
 0  0      0 411592   4336  39864    0    0     0    20 2879 10097  4  8 88  0
 0  0      0 410936   4344  40116    0    0    26     4 2894 10103  5  8 87  0
 0  0      0 410464   4352  40108    0    0    26    29 2886 10090  4  7 88  0
 0  0      0 409920   4368  40092    0    0     0    11 2901 10148  4  8 88  0
 0  0      0 409272   4384  40336    0    0    26     6 2872 10101  4  8 88  0
 0  0      0 408936   4408  40312    0    0     0    33 2928 10133  5  8 87  0
 0  0      0 408664   4416  40304    0    0    26     2 2894 10090  4  8 88  0
 0  0      0 408536   4432  40548    0    0    26     6 2893 10095  5  8 87  0
 0  0      0 408528   4448  40532    0    0     0     6 2880 10126  5  8 88  0
```

Figura 30: Saída do commando “vmstat 5 10” para 150 chamadas recebidas via DUNDi

```
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5061      59.96 s          100  192.168.130.10:5060(UDP)

  0 new calls during 0.947 s period      4 ms scheduler resolution
  100 calls (limit 100)                  Peak was 100 calls, after 10 s
  0 Running, 100 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets
------------------------------ Scenario Screen -------- [1-9]: Change Screen --
  Call-rate(length)     Port   Total-time  Total-calls  Remote-host
10.0(100000 ms)/1.000s   5062      64.46 s          150  192.168.130.10:5060(UDP)

  0 new calls during 0.452 s period      3 ms scheduler resolution
  150 calls (limit 150)                  Peak was 150 calls, after 15 s
  0 Running, 150 Paused, 0 Woken up
  0 out-of-call msg (discarded)
  1 open sockets

                                 Messages  Retrans   Timeout   Unexpected-Msg
     INVITE ---------->          150       9         0
        100 <----------          149       9         0         0
        180 <----------          0         0         0         0
        183 <----------          0         0         0         0
        200 <----------  E-RTD1  150       7         0         0
        ACK ---------->          150       0
      Pause [     1:40]          150                           0
        BYE ---------->          0         0         0
        200 <----------          0         0         0         0

------------------------------ Test Terminated --------------------------------


------------------------------ Statistics Screen ------- [1-9]: Change Screen --
  Start Time              | 2008-02-18   16:13:15:194    1203361995.194548
  Last Reset Time         | 2008-02-18   16:14:19:207    1203362059.207957
  Current Time            | 2008-02-18   16:14:19:661    1203362059.661241
--------------------------+---------------------------+--------------------------
  Counter Name            | Periodic value            | Cumulative value
--------------------------+---------------------------+--------------------------
  Elapsed Time            | 00:00:00:453              | 00:01:04:466
  Call Rate               |    0.000 cps              |    2.327 cps
--------------------------+---------------------------+--------------------------
  Incoming call created   |        0                  |        0
  OutGoing call created   |        0                  |      150
  Total Call created      |                           |      150
```

Figura 31: Saída de status do SIPP para 150 chamadas recebidas via DUNDi

Referências

BAKER, M. Cluster computing white paper. In: PORTSMOUTH University of Portsmouth. 2000. Disponível em: <http://arxiv.org/ftp/cs/papers/0004/0004014.pdf>. Acesso em: 4 mar. 2008.

BIND. [s.n.], 2000. Disponível em: <http://www.isc.org/index.pl?/sw/bind/index.php>. Acesso em: 4 mar. 2008.

DISC-OS. [s.n.], 2007. Disponível em: <http://www.disc-os.org>. Acesso em: 4 mar. 2008.

FRIEDMAN, R. *OpenMP Architecture Review Board*. [s.n.], 1997. Disponível em: <http://www.openmp.org/blog/about>. Acesso em: 27 nov. 2007.

GNUPLOT. [s.n.], 1999. Disponível em: <http://www.gnuplot.info>. Acesso em: 4 mar. 2008.

GROPP, W. D.; LUSK, E. *The Message Passing Interface (MPI) standard*. [s.n.], 2005. Disponível em: <http://www-unix.mcs.anl.gov/mpi>. Acesso em: 27 nov. 2007.

KNOX, B. *The OpenMosix Project*. [s.n.], 2002. Disponível em: <http://openmosix.sourceforge.net>. Acesso em: 27 nov. 2007.

MEGGELEN, J. V.; SMITH, J.; MADSEN, L. Asterisk: O futuro da Telefonia. In: _____. *Tradução de Armando Figueiredo e Betina Macedo*. [S.l.]: Alta Books, Rio de Janeiro, RJ, 2005.

MERKEY, P. *Beowulf History*. [s.n.], 2004. Disponível em: <http://www.beowulf.org/overview/history.html>. Acesso em: 8 fev. 2008.

PVM. *PVM Paralelal Virtual Machine*. [s.n.], 2007. Disponível em: <http://www.csm.ornl.gov/pvm>. Acesso em: 27 nov. 2007.

RICHARDSON, J. *DUNDi, So Easy a Caverman Could Do It*. [s.n.], 2006. Disponível em: <http://www.voip-info.org/wiki-DUNDi>. Acesso em: 16 jan. 2008.

RICHARDSON, J. *JR Richardson Writepaper*. [s.n.], 2006. Disponível em: <http://www.voip-info.org/wiki-Asterisk+DUNDi+Call+Routing>. Acesso em: 11 nov. 2007.

SILVA, G. P. Clusters. In: RIO DE JANEIRO,Universidade Federal do Rio de Janeiro. 2005. Disponível em: <http://equipe.nce.ufrj.br/gabriel/sispar/ArqPar6.pdf>. Acesso em: 25 fev. 2008.

SIPP. [s.n.], 2004. Disponível em: <http://sipp.sourceforge.net>. Acesso em: 4 mar. 2008.

SPENCER, M. Universal number discovery (dundi) draft-mspencer-dundi-01. In: DIGIUM. 2004. Disponível em: <http://www.dundi.com/dundi.txt>. Acesso em: 25 fev. 2008.

SUSIN, G. Análise de desempenho de um cluster para execução do modelo de previsão do tempo arps. In: FLORIAÓPOLIS Universidade Federal de Santa Catarina. 2001. Disponível em: <http://www.tede.ufsc.br/teses/PGCC0470.pdf>. Acesso em: 22 jan. 2008.