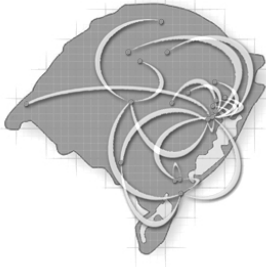


I Escola Regional de Redes de Computadores – ERRC 2003

**Porto Alegre, RS - Brasil
22, 23 e 24 de setembro de 2003**



Anais

Editores

Lisandro Zambenedetti Granville

Juergen Rochol

Luciano Paschoal Gaspar

Fernando Luis Dotti

João Cesar Netto

Jorge Guedes

Coordenação

Universidade Federal do Rio Grande do Sul (UFRGS)

Organização

Pontifícia Universidade Católica do Rio Grande do Sul (PUC-RS)

Universidade do Vale do Rio dos Sinos (UNISINOS)

Promoção

Sociedade Brasileira de Computação, Secretaria do Rio Grande do Sul – SBC-RS

Impressão
C V Artes Gráficas Ltda.

Tiragem
100 exemplares

I Escola Regional de Redes de Computadores – ERRC 2003
(2. : 2003 22 a 24 de setembro: Porto Alegre, RS)

Anais / I Escola Regional de Redes de Computadores – ERRC 2003; editores Lisandro Zambenedetti Granville, Juergen Rochol, Luciano Paschoal Gaspary, Fernando Luis Dotti, João Cesar Netto e Jorge Guedes, Porto Alegre, 22 a 24 de setembro de 2003. Porto Alegre : Sociedade Brasileira de Computação, 2003.

x, 156p. ; 21 cm

1. Redes 2. Redes de Computadores 3. Programa I. Título II. Granville, Lisandro Zambenedetti. III. Rochol, Juergen. IV. Gaspary, Luciano Paschoal. V. Dotti, Fernando Luis. VI. Netto, João Cesar. VII. Guedes, Jorge.

Prefácio

Esta primeira edição da **Escola Regional de Redes de Computadores (ERRC)** é resultado do esforço conjunto e da iniciativa da Universidade Federal do Rio Grande do Sul (UFRGS), da Pontifícia Universidade Católica do Rio Grande do Sul (PUC-RS) e da Universidade do Vale do Rio dos Sinos (UNISINOS). O objetivo principal da Escola é permitir o debate e a discussão das atividades relacionadas à área de redes de computadores no Rio Grande do Sul. O público alvo da escola é formado por professores, profissionais e alunos de graduação e pós-graduação em redes. Entre os dias 22 a 24 de setembro de 2003 diversas atividades serão desenvolvidas nos prédios 11 e 40 da PUC-RS. A Escola é composta por minicursos, sessões técnicas, oficinas, fóruns de debate e palestras.

Nos mini-cursos são abordados temas atuais e relevantes da área de redes, apresentados por um ministrante principal. Artigos técnicos relatando os trabalhos desenvolvidos nos centros de pesquisa e universidades são apresentados pelos seus autores em sessões técnicas temáticas. Já nas oficinas, conduzidas por uma equipe, os participantes tem contato com os tópicos de redes através de uma abordagem prática. Por fim, as palestras e fóruns têm o objetivo de criar um ambiente para o debate de temas relevantes.

Esperamos que esta ERRC seja a primeira de uma série de Escolas voltadas à difusão, discussão e promoção da área de redes de computadores na região. Esta primeira edição só foi possível graças à colaboração de várias pessoas. Gostaríamos de agradecer aos coordenadores do evento: Luciano Paschoal Gaspary (coordenador do comitê de programa), Fernando Luis Dotti (coordenador de minicursos), João Cesar Netto (coordenador de oficinas) e Jorge Guedes (coordenador de palestras). Não poderíamos de deixar de agradecer também aos alunos de graduação e pós-graduação que trabalharam incansavelmente para que a ERRC fosse possível: a todos nosso muito obrigado. Agradecemos também o importante apoio dado pela SBC, através de sua secretaria do Rio Grande do Sul, a esta iniciativa. Por fim, desejamos a todos os participantes um ótimo e proveitoso evento.

Lisandro Zambenedetti Granville, UFRGS
Juergen Rochol, UFRGS
Coordenadores Gerais da ERRC

Porto Alegre, setembro de 2003

Carta do Coordenador do Comitê de Programa

A **Escola Regional de Redes de Computadores (ERRC)** é um evento que tem como principal objetivo propiciar a professores, profissionais e alunos de graduação e pós-graduação um fórum para discussão e divulgação das atividades e pesquisas relacionadas à área de redes de computadores no Rio Grande do Sul.

Já na sua primeira edição a comunidade legitima o evento com um elevado número de submissões. De um total de 52 artigos submetidos, o comitê de programa da ERRC 2003 selecionou 24 para apresentação e publicação. Recebemos trabalhos de autores vinculados a entidades públicas e privadas espalhadas pelos estados do Rio Grande do Sul, Santa Catarina, São Paulo, Rio de Janeiro, Minas Gerais e Bahia, bem como do Distrito Federal.

O processo de avaliação dos artigos contou com a participação direta dos vinte e três membros do comitê de programa e de mais oito revisores associados. Cada artigo foi revisado por, pelo menos, dois especialistas na área.

Nestes Anais encontram-se os textos completos dos artigos selecionados, organizados em seis sessões técnicas: (1) gerência e operação de redes, (2) aplicações e estudos de caso, (3) segurança, (4) redes peer-to-peer, qualidade de serviço e roteamento em redes IP, (5) comunicação sem fio e (6) mobilidade e tolerância a falhas.

Meus sinceros agradecimentos aos membros do comitê de programa e revisores pelo árduo trabalho realizado em um curto espaço de tempo. É com esse processo cuidadoso de revisão que poderemos manter o alto nível de qualidade da Escola Regional de Redes de Computadores.

Luciano Paschoal Gaspary, UNISINOS
Coordenador do Comitê de Programa

São Leopoldo, setembro de 2003

Carta do Coordenador de Minicursos

A primeira **Escola Regional de Redes de Computadores (ERRC)**, entre seus vários objetivos, pretende criar um espaço para complementação na formação de estudantes das áreas afins. Uma excelente forma de fazer isto é trazendo minicursos de interesse da comunidade para a programação. Neste sentido, um espaço importante da Escola é dedicado aos minicursos selecionados.

Em “Avanços rumo à Integração de Tecnologias de Gerenciamento de Redes e Segurança”, Luciano Paschoal Gaspary e Leonardo Lemes Fagundes (UNISINOS) apresentam técnicas que permitem aproximar as tecnologias de gerência das de segurança, discutindo aspectos favoráveis e não favoráveis dessa integração, e apresentam uma abordagem para detectar diversos tipos de ataques utilizando o protocolo SNMP e a observação de objetos das MIBs II, RMON e RMON2.

Em “Comunicação Multicast em Middleware CORBA”, Lau Cheuk Lung (PUC-PR, PPGIA), Alysson Bessani Neves e Joni da Silva Fraga (UFSC, DAS) introduzem conceitos e abstrações de comunicação de grupo, apresentando os tipos de difusão (multicast) possíveis, e como eles podem ser introduzidos na arquitetura CORBA.

Ana Cristina Benso da Silva (PUCRS; FACIN), Andrei Oliveira da Silva e Fabrício D'Avila Cabral (Centro de Pesquisa em Software Embarcado - convênio PUCRS-HP) apresentam “Redes Wireless Bluetooth”, abordando aspectos históricos, tecnológicos e práticos, e visando proporcionar conhecimento básico sobre esta tecnologia de tal forma que, ao final do minicurso, os alunos estarão capacitados a instalar e configurar uma rede Bluetooth.

Os tópicos são variados e ricos e certamente serão de proveito para um grande número de estudantes e profissionais.

Por fim, antes de desejar a todos um bom proveito desta Escola e especificamente dos minicursos, eu gostaria de agradecer aos coordenadores gerais deste evento pelo convite para coordenação dos Minicursos da ERRC.

Fernando Luis Dotti, PUC-RS
Coordenador de Minicursos

Porto Alegre, setembro de 2003

Carta do Coordenador de Oficinas

A **Escola Regional de Redes de Computadores (ERRC)**, em sua primeira edição, visa congrega alunos, professores e pesquisadores da área, e fomentar contatos e a troca de idéias desta grande comunidade. Neste contexto estão as OFICINAS. Oficina é um lugar onde as ferramentas estão disponíveis, onde se fazem testes, onde se observa o funcionamento das coisas. Enfim, onde através da prática, colocamos questionamentos para futuros estudos.

As oficinas do ERRC 2003 abrangem 3 tópicos:

- **Simulação.** Ferramental utilizado e quase imprescindível no projeto de novos protocolos e tecnologias, bem como para a avaliação de desempenho das tecnologias já disponíveis;
- **Monitoração de redes, através de *probes* RMON-2.** Este tipo de observação permite que seja traçada a assinatura de uma rede (comportamento esperado), bem como a detecção antecipada de anomalias e verificação do desempenho da rede;
- **Qualidade de serviço (QoS).** Assunto atual e determinante na Internet 2. A experimentação das técnicas disponíveis permite verificar sob quais condições tais técnicas devem ser aplicadas, além de fomentar o estudo desta área em amplo desenvolvimento científico e tecnológico.

Desejo um bom proveito àqueles que seguirem as práticas, e um voto de louvor aos pesquisadores, assistentes e técnicos que possibilitaram que esta oportunidade nos fosse apresentada.

João Cesar Netto, UFRGS
Coordenador de Oficinas

Porto Alegre, setembro de 2003

Coordenação

Coordenação Geral

Lisandro Zambenedetti Granville, UFRGS

Juergen Rochol, UFRGS

Coordenação do Comitê de Programa

Luciano Paschoal Gaspar, UNISINOS

Coordenação de Minicursos

Fernando Luis Dotti, PUC-RS

Coordenação de Oficinas

João Cesar Netto, UFRGS

Coordenação de Palestras

Jorge Guedes, PUC-RS

Comitê de Programa

Alexandre da Silva Carissimi, UFRGS

Ana Cristina Benso da Silva, PUC-RS

Avelino Francisco Zorzo, PUC-RS

Benhur de Oliveira Stein, UFSM

Cláudio Fernando Resin Geyer, UFRGS

Cristian Koliver, UCS

Fernando Luis Dotti, PUC-RS

Gerson Geraldo Homrich Cavalheiro, UNISINOS

Ingrid Eleonora Schreiber Jansch-Pôrto, UFRGS

João Cesar Netto, UFRGS

Jorge Guedes Silveira, PUC-RS

Jorge Luis Victória Barbosa, UCPEL

Juergen Rochol, UFRGS

Katia Barbosa Saikoski, PUC-RS

Liane Margarida Rockenbach Tarouco, UFRGS

Lisandro Zambenedetti Granville, UFRGS

Luciano Paschoal Gaspar, UNISINOS

Marco Antônio Sandini Trentin, UPF

Maria Janilce Bosquiroli Almeida, UFRGS

Marinho Pilla Barcellos, UNISINOS

Taisy Silva Weber, UFRGS

Vinícius da Silveira Serafim, UFRGS

Valter Roesler, UNISINOS

Comitê de Organização

Evandro Della Vecchia Pereira, UFRGS

Fernando Luis Dotti, UNISINOS

Laura Silva Lopes Gonçalves, UFRGS

Liane Margarida Rockenbach Tarouco, UFRGS

Maria Janilce Bosquiroli Almeida, UFRGS

Marinho Pilla Barcellos, UNISINOS

Michelle Denise Leonhardt, UFRGS

Valter Roesler, UNISINOS

Revisores

Alexandre da Silva Carissimi, UFRGS
Ana Cristina Silva, PUC-RS
Antonio Candia, UFSM
Avelino Francisco Zorzo, PUC-RS
Benhur de Oliveira Stein, UFSM
Cláudio Fernando Resin Geyer, UFRGS
Cristian Koliver, UCS
Diego Kreutz, UFSM
Fernando Luis Dotti, PUC-RS
Gabriela Silva, UFSM
Gerson Geraldo Homrich Cavalheiro, UNISINOS
Ingrid Eleonora Schreiber Jansch-Pôrto, UFRGS
João Cesar Netto, UFRGS
Jorge Barbosa, UNISINOS
Jorge Guedes, PUC-RS
Juergen Rochol, UFRGS
Katia Barbosa Saikoski, PUC-RS
Liane Margarida Rockenbach Tarouco, UFRGS
Lisandro Zambenedetti Granville, UFRGS
Luciano Paschoal Gaspar, UNISINOS
Lúcio Braga, UNISINOS
Márcia Pasin, UFRGS
Marco Antônio Sandini Trentin, UPF
Maria Janilce Bosquioli Almeida, UFRGS
Marinho Pilla Barcellos, UNISINOS
Ricardo Neisse, UFRGS
Taisy Silva Weber, UFRGS
Tiago Fioreze, UFRGS
Valter Roesler, UNISINOS
Vinicius Serafim, UFRGS

Sumário

Sessão Técnica 1: Gerência e Operação de Redes

<i>Melhoria de Gerência de um Ambiente de Rede TPCI com Auxílio das Facilidades de uma Rede de Computadores</i> Arlete Cardoso Duarte Oliveira (UnB), Mário A. R. Dantas (UFSC)	2
<i>Gerenciamento Integrado dos Recursos do Sistema para obter QoS Fim-a-Fim</i> Sílvia Cristina Sardela Bianchi (UFSC), Carlos Becker Westphall (UFSC), André de Barros Sales (Université Paul Sabatier), Michelle Sibilla (Université Paul Sabatier), Carla Merkle Westphall (UFSC)	8
<i>BGP – Sub-Agente para Gerenciamento do Protocolo BGP</i> Andrey Vedana Andreoli (POP-RS), Leandro Márcio Bertholdo (POP-RS), Liane Tarouco (POP-RS), Ana Benso da Silva (PUC-RS), Fábio Rodrigues (PUC-RS)	14
<i>Monitoração de Disponibilidade e Desempenho de Servidores Críticos usando uma Abordagem Descentralizada</i> Juliano Valentini (UNISINOS), Luciano Paschoal Gaspary (UNISINOS)	20

Sessão Técnica 2: Aplicações e Estudos de Caso

<i>Security Site – Desenvolvendo um Ambiente Seguro para E-Commerce e E-Business</i> Thiago de Moraes Pereira (UEMG), Alessandro de Castro Borges (UEMG)	27
<i>Análise de Aplicações de Voz em Redes IP</i> Adriana Patrícia de Oliveira (UnB), Mário A. R. Dantas (UFSC)	33
<i>Desenvolvimento de Aplicações 3G - Perspectivas</i> Lucas Mello Schnorr (UFRGS), Juergen Rochol (UFRGS)	39
<i>Sistema de Hardware e Software para Videomonitoração Através de Telefones Celulares</i> Jorge Guedes (PUC-RS), Rafael Rehm (PUC-RS), Fernando Thiesen (PUC-RS), Francisco Souza (PUC-RS), Luciano Azevedo (PUC-RS)	45

Sessão Técnica 3: Segurança

<i>Uma Solução de Autenticação Fim a Fim para o LDP (Label Distribution Protocol)</i> Morvan D. Muller (UFRGS), Carlos Becker Westphall (UFRGS), Carla Merckle Westphall (UFSC)	52
<i>Análise dos Processos de Segurança em Sistemas Móveis de 3ª Geração</i> Fabrício Jorge Lopes Ribeiro (UFRJ), Jaime Cesar Ribeiro Lopes (UFRJ), Aloysio de Castro P. Pedroza (UFRJ)	59
<i>Análise das ferramentas de IDS SNORT e PRELUDE quanto à eficácia na detecção de ataques e na proteção quanto à evasões</i> Julio Steffen Junior (FEEVALE), Eduardo Leivas Bastos (FEEVALE)	65
<i>Compreendendo Ataques Denial of Services</i> Leandro Márcio Bertholdo (CERT-RS), Andrey Vedana Andreoli (CERT-RS), Liane Tarouco (CERT-RS)	71

Sessão Técnica 4: Rede Peer-to-Peer, Qualidade de Serviço e Roteamento em Redes IP

<i>Localização de Conteúdo em Redes Peer-to-Peer</i> André Detsch (UNISINOS)	78
<i>A Tecnologia Peer-to-Peer como Ferramenta para Comunicação em Redes Ad Hoc Móveis</i> Felipe Jung Vilanova (UFRGS), Juergen Rochol (UFRGS), Maria Janilce Bosquiroli Almeida (UFRGS), Lisandro Zambenedetti Granville (UFRGS)	84
<i>Arquiteturas Híbridas de QoS em Redes IP</i> Adelmo Jerônimo Silva (UnB), Mário A. R. Dantas (UFSC)	90
<i>Roteando redes Ipv6 e Multicast com MP-BGP</i> Andrey Vedana Andreoli (POP-RS), Leandro Márcio Bertholdo (POP-RS), Liane Tarouco (POP-RS)	96

Sessão Técnica 5: Comunicação sem Fio e Mobilidade

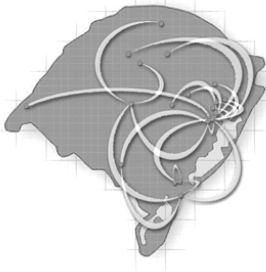
<i>Roteamento com Agregação de Dados em Redes de Sensores</i> Jorgito Matiuzy Stochero (UFRJ), Antonio José Gonçalves Pinto (UFRJ), José Ferreira de Rezende (UFRJ)	103
<i>A Tecnologia CDMA: Revisão Teórica e Aplicações em Sistemas Wireless</i> Diego Moreira da Rosa (UFRGS), Fabio Irigon Pereira (UFRGS), Juergen Rochol (UFRGS)	109
<i>Efeito da Precisão Numérica do Conversor sobre a Taxa de Erros de um Software Radio</i> Diego Moreira da Rosa (UFRGS), Luigi Carro (UFRGS)	115
<i>Modelo de Arquitetura para simulação de redes móveis sem fio ad hoc no Simmcast</i> Daniela Saccol Peranconi (UNISINOS), Hisham H. Muhammad (UNISINOS), Marinho P. Barcellos (UNISINOS)	121

Sessão Técnica 6: Tolerância a Falhas

<i>Tolerância a Falhas em Sistemas de Agentes Móveis</i> Tiago Fioreze (UFRGS), Ingrid Jansch-Pôrto (UFRGS), Lisandro Zambenedetti Granville (UFRGS)	128
<i>Alta Disponibilidade aplicada a Computação Móvel</i> Hélio Antônio Miranda da Silva (UFRGS), Ingrid Jansch-Pôrto (UFRGS)	134
<i>Tolerância a Falhas em Sistemas de Armazenamento de Dados</i> Márcio Joel Barth (PROCERGS, FEEVALE), Edvar Bergmann Araújo (FEEVALE)	140
<i>Uma Estratégia para Validação Experimental de um Sistema de Comunicação de Grupo</i> Gabriela Jacques da Silva (UFRGS), Taisy Silva Weber (UFRGS)	146

Resumos

Minicursos	153
Oficinas	155



Sessão Técnica 1

Gerência e Operação de Redes

Melhoria de Gerência de um Ambiente de Rede TPCI com Auxílio das Facilidades de uma Rede de Computadores

Arlete Cardoso Duarte Oliveira¹, M.A.R. Dantas²

¹Departamento de Engenharia Elétrica – Laboratório de Engenharia de Redes
(UnB/ENE/LabRedes) – Universidade de Brasília
Av.L3 Norte – FT – ENE – LabRedes – Sala B1-01 – Asa Norte – CEP:70910-900 – Brasília-
DF – Brasil

²Departamento de Informática e Estatística (INE) – Universidade Federal de Santa Catarina
(UFSC)
Caixa Postal 476 – Trindade – Florianópolis – SC – 88040-900 – Brasil
arlete@brt14.com.br, mario@inf.ufsc.br

Abstract – *The available facilities in computer networks can serve as an excellent tool for the management of some configurations in the telephony nets. The public telephony with the device of inductive card, as was enacted by the ANATEL [1], spreads the access at the telecommunications to all the far regions of the Brazil. The viability of the attendance with successfully of the proposal is based on the full functioning of the plant in service, fact this that is guaranteed by the supervision system. However, some restrictions in the visualization of the information collected for the system exist. In this context, we present in this article a study of the stored data and as we obtain information that could mitigate the income evasion through reports centered on this goal. Our proposal it is interesting, because it uses resources of the computer network that does not require bulky budgets, or extra costs for the operators companies and using the existing technology.*

Resumo – *As facilidades disponíveis nas redes de computadores podem servir como uma excelente ferramenta para o gerenciamento de algumas configurações nas redes de telefonia. A telefonia pública com o aparelho de cartão indutivo, como foi regulamentada pela ANATEL [1], faz chegar o acesso das telecomunicações a todos os pontos do Brasil. A viabilidade do atendimento com sucesso da proposta, está baseada no pleno funcionamento da planta em serviço, fato este que é garantido pelo sistema de supervisão. Porém, existem algumas restrições na visualização das informações coletadas pelo sistema. Neste contexto, apresentamos neste artigo um estudo dos dados armazenados e como retiramos informações que possibilitam diminuir a evasão da receita com relatórios direcionados a esse objetivo. Nossa proposta é interessante, pois emprega recursos da rede de computadores que não requerem vultosos investimentos, nem custos adicionais para as operadoras e utilizamos a tecnologia existente.*

1. Introdução

A grande concorrência entre empresas que prestam serviços de telecomunicações, fazem surgir importantes questões sobre comunicações. Dentre muitos aspectos podemos citar a redução de custos operacionais, otimização de tarefas, um conhecimento dos clientes e a capacidade de satisfazê-los. Garantir a efetiva receita na prestação de serviços de telecomunicações é um processo bastante complexo para uma operadora. Este fator é principalmente relevante, quando tratamos do telefone público a cartão indutivo [2].

Nossa proposta é um estudo de caso do sistema de supervisão remota de telefones públicos a cartão indutivo [4] [5] [6]. Para tal utilizamos facilidades de uma rede de computadores para um melhor gerenciamento de um ambiente de rede de TPCI. Nossa abordagem tem como foco trabalhar o complemento das informações do módulo cliente da supervisão através de um banco de dados convencional disponível em nosso ambiente de trabalho (exemplo é o Microsoft SQL Server 7 [7]). Utilizamos como ferramentas *scripts/queries* [8,9] de análise acessando diretamente o banco de dados.

Nosso objetivo é obtenção e agrupamento de uma forma melhorada das informações relevantes. Com estas informações é esperada uma análise mais eficaz, seguida de uma posterior tomada de decisão quanto a manutenção ou posicionamento da tecnologia em serviço. Desta forma, é possível se atingir uma diminuição da evasão de receita com a utilização de pacotes de software existentes.

A relevância de nosso trabalho está em diminuir a evasão da receita com relatórios direcionados a esse objetivo. Em outras palavras, buscamos apresentar relatórios que obtém do banco de dados informações originadas dos TPCI (Telefone Público de Cartão Indutivo) [2] em campo e repassadas para o sistema de supervisão. Todavia, no módulo de relatórios do sistema de supervisão estas informações não são visíveis. Assim, nossa abordagem pode trazer uma longevidade maior à tecnologia hoje existente, sem vultosos investimentos e sem custos adicionais para as operadoras. Conseqüentemente este fato pode nos levar a uma prestação de serviço de maior alcance. Esta solução é interessante em especial nos casos das comunidades carentes, nas áreas de grande congestionamento e, finalmente, levar as operadoras a considerar uma ampliação maior da planta do que é exigido por regulamento, uma vez que a mesma passará a ser mais atrativa.

Como abordagem acima demandaria em diversas ações e pesquisas, algumas as quais não teríamos acesso para quantificarmos posteriormente, delimitamos o experimento na diminuição da evasão de receita.

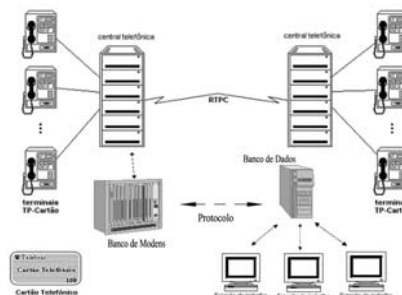


Figura 1. Ambiente Simulado [5]

Para um melhor entendimento este artigo foi estruturado da seguinte forma: na segunda seção apresentamos uma visão rápida da estrutura de funcionamento do TPCI e o ambiente montado para o experimento; na terceira seção mostramos os experimentos e seus resultados; a quarta seção com as conclusões e propostas para pesquisas futuras.

2. Ambiente Experimental

Para a realização de nossos experimentos, fez necessária a criação de uma configuração similar ao serviço ilustrado na Figura 1 [5]. Nosso objetivo com a configuração foi a obtenção de um volume de dados grande para iniciarmos com nossa pesquisa e posteriormente aplicarmos a um banco de dados real.

Podemos observar na Figura 1, que a rede física dos TPCIs está estruturada de uma forma similar a qualquer aparelho fixo de telefonia, estando interligados fisicamente as centrais telefônicas. A interligação lógica com um sistema de supervisão através de banco de modems é o diferencial dos TPCIs.

A interligação é efetuada através de um protocolo específico de comunicação [10], o qual prove acesso ao banco de dados de sua supervisão. As diversas informações do ambiente (tais como chamadas originadas e recebidas, cartões [12] [13] [14] [15], falhas e o status do terminal, no momento da comunicação) são assim armazenadas.

Interessante observar que a figura exemplifica o modelo e o meio de transmissão utilizado. O enlace de comunicação deve ser genérico[11], podemos ter a ligação via rádio celular, via enlace de rádio ou até mesmo via satélite. No caso do satélite, em canal de transmissão que possibilite sinal de voz e transmissão de dados dentro da faixa de voz, sem duplo salto.

Para nossa pesquisa, focamos a atenção em uma máquina central onde instalamos o banco de dados SQL Server 7 [7] e suas ferramentas administrativas.

Criamos um banco de dados como *primário* e neste armazenamos informações vindas da comunicação do telefone público à cartão indutivo [13], colhidas no protocolo de comunicação. Visando uma analogia com o projeto do sistema de supervisão [5], criamos um segundo banco denominado de *repositório*. Este segundo banco é uma imagem do banco *principal*. A figura 2, nos permite visualizar pelo Enterprise Manager [7] os bancos ora descritos.

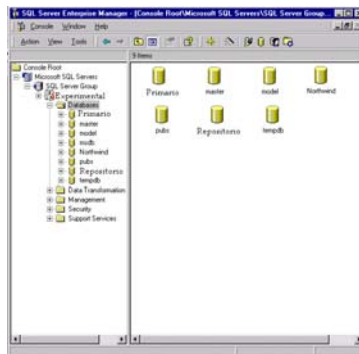


Figura 2. Banco de Dados do Ambiente Experimental

3. Experimentos

Devido a enorme quantidade de informações estabelecemos um limite para nossos testes em função: (a) da diminuição da evasão de receita através de uma monitoração de falhas na planta e (b) evasão de receita causada por ações intencionais sobre o terminal e/ou linha telefônica associada.

3.1. Diminuição da Evasão de Receita pela Monitoração de Falhas nos TPCIs

Uma causa observada para a evasão de receita são as falhas ocorridas na tecnologia TPCI [10], ou na rede de telecomunicações. Algumas dessas falhas envolvem o processo de tarifação das chamadas [3], outras deixando o terminal fora de serviço por longo período de tempo e por último, algumas falhas que não chegam nem ao conhecimento do operador do sistema de supervisão [5] por falta de mecanismos de visualização.

Nossos laboratórios para este tópico foram direcionados à elaboração de consultas em SQL que retornasse relatórios com informações de falhas relacionadas acima, somente dos TPCIs que estivessem com esse *status* no momento da emissão dos relatórios. Ao final nossos relatórios apontaram com sucesso para uma quantidade de TPCIs que se enquadraram em nossa seleção.

A relevância dessas informações está numa visualização única das informações de falhas, em ordem crescente de quantificação de dias do ocorrido ou o número de ocorrências; trazendo informações de forma de tarifação, do fabricante e da versão do software do TPCI; aferindo as cadências aplicadas; as chamadas expurgadas do sistema e as causas que levaram a isso; o desempenho do banco de modems; as chamadas que não foram tarifadas por falta de recebimento do pulso da central; isso de forma real time ao momento de execução do mesmo na ferramenta administrativa Query Analyser [7], sendo que o resultado pode ser manipulado por qualquer operador através da importação do arquivo da forma que o mesmo preferir.

3.2. Compartilhamento de Mercado

Com o surgimento da escolha de CSP (Código de Seleção da Prestadora) [18], foi observado no mercado um crescimento em opções de operadoras e por este motivo uma maior malha de interconexão. Este fato pode significar uma fonte de renda ou de evasão de receita, dependendo da eficiência do trabalho de cada operadora.

O interessante dessa informação está em se fazer um mapeamento dos pontos onde cada empresa aparece como mais atuante. Desta forma, uma determinada operadora é capaz de realizar uma recuperação da divisão do mercado, através do direcionamento do uso de mídia seja do display do TPCI, como televisiva ou através do rádio com uma maior eficiência dos dispositivos.

O atrativo da proposta foi a demonstração da possibilidade de reversão de escolha da prestadora, através de ações de cunho explícito de esclarecimento ao usuário sobre a abrangência e o valor da tarifa em cada ligação feita através do CSP da empresa ao qual trabalhamos em parceria.

O ápice desse enfoque foi ter todos os pontos mapeados, podendo-se mensurar individualmente ou por regiões, o impacto do compartilhamento de mercado, pelo volume de chamadas, na receita cessante da operadora em questão.

3.3. Detecção de Problemas na Coleta de Créditos

Problemas na coleta de créditos, durante uma chamada, podem ter várias origens. Citando algumas, temos a extensão ou derivação da linha telefônica por outro terminal, as perturbações intencionais na detecção do pulso enviado pela central, as chamadas realizadas através do acesso interno ao TPCI.

Para esse experimento propusemos a recuperação de receita num valor aproximado ao da evasão encontrada, entre os créditos queimados e o valor pago para outra prestadora na interconexão das chamadas, baseando nossas indicações de ações operacionais, nos resultados de nosso experimento.

Devido a um problema na tecnologia, optamos por fechar as informações em valores recebidos durante todas as chamadas de um determinado TPCI, num período de tempo e valores pagos posteriormente para a empresa do CSP, para esse mesmo período e TPCI, inclusive, impostos. Solução essa que trouxe não só agilidade ao processo, como também simplificou o entendimento e a comprovação da evasão de receita, sendo esse último o fator decisivo para sua implantação.

Para tanto, foi necessário a criação de uma consulta só com dados de bilhetes originados, com um filtro para uma determinada operadora, realizando busca no banco de dados do sistema de supervisão e fazendo o cruzamento do resultado com os bilhetes recebidos em fatura para a mesma operadora do filtro de análise.

Em nossas consultas chegamos a uma conclusão que o valor pago em interconexão para uma operadora, a evasão de receita representou 19% do total, o que demonstrou que nosso experimento tem uma importância grande na indicação para o processo de reversão de receita. Uma vez que nossa proposta se atém a cobrir alguns meses anteriores, no mês subsequente ocorrendo uma concretização de ações que apontamos como corretivas, houve uma queda na evasão de 22,5% do valor em relação ao mês anterior.

4. Conclusões

O objetivo desse artigo foi demonstrar que com o uso de uma ambiente de banco de dados, disponível em uma rede de computadores, foi possível melhorar o gerenciamento das operações dos TPCI. O gerenciamento diferenciado trouxe uma melhoria no controle de evasão de receitas, pois foi possível realizar uma melhor análise que auxiliou na tomada de decisão quanto à manutenção, ou posicionamento da tecnologia em serviço.

As consultas que realizamos, quando lançadas para as equipes de manutenção e de consultoria de uma operadora de telefonia fixa, demonstraram sua eficácia na solução do problema, seja de falha ou de evasão de receita.

Em adição, neste trabalho verificamos potencialidades no TPCI e também nos deparamos com problemas desta tecnologia que devem ser consideradas como: a falta de sincronismo entre o relógio dos bilhetadores das centrais e o relógio interno do TPCI, fazendo com os bilhetes gerados no último, tornarem-se difíceis de relacionar-se sem que haja um algoritmo de tratamento dos mesmos.

Como trabalhos futuros seria interessante incluir outras possibilidades tais como a consolidação de cartões fabricados, com os vendidos e os utilizados; outro trabalho que poderia acrescentar satisfação ao cliente, seria monitorar, por região, o tipo de cartão (quantidade de créditos) que possui uma melhor aceitação de mercado; outra possibilidade seria experimentos de adaptação do telefone público de forma a atender, de uma forma

maximizada, os diversos tipos de deficiências, conforme orienta o novo Plano Geral de Metas de Universalização [18] onde temos que “*os portadores de deficiência poderão, diretamente, ou por meio de quem os represente, solicitar adaptação do Terminal de Uso Público, referida no caput, de acordo com a deficiência ...*”

Bibliografia

- [1] Plano Geral de Metas de Universalização – PGMU, aprovado pelo Decreto nº2.592, de 15 de maio de 1998.
- [2] SDT 245-300-707 (PADRÃO) – Série Engenharia – Especificação de Aparelho Telefônico Público a Cartão Indutivo. Emissão 04/07/1995.
- [3] http://www.anatel.gov.br/index.asp?link=/telefonia_fixa/stfc/tarifas98.htm. Data de consulta 05/05/2003.
- [4] SDT 560-400-501 (PADRÃO) – Série Planta – Procedimentos de Gerência e Supervisão de Aparelho Telefônico Público e Semipúblico a Cartão Indutivo. Emissão 01/12/1991.
- [5] Sistema de Supervisão Remota para Telefones Públicos a Cartão Indutivo. Módulo Administrador. Fundação Centro de Pesquisa e Desenvolvimento, 08/1999.
- [6] SDT 235-710-701 (PADRÃO) – Série Engenharia – Especificação do Centro de Supervisão Automatizada para Telefones Públicos a Cartão Indutivo. Emissão 03/02/1994.
- [7] Microsoft SQL Server Introduction. Microsoft Corporation, U.S.A., 1998.
- [8] RAMALHO, José Antônio Alves. SQL: A Linguagem dos Bancos de Dados. São Paulo, Berkeley Brasil, 1999.
- [9] DATE, C. J. Introdução a Sistemas de Bancos de Dados. Editora Campus, 2000.
- [10] SDT 245-300-709 (PADRÃO) – Série Engenharia - Especificação do Protocolo de Comunicação Entre o Centro de Supervisão Automatizada e o Aparelho Telefônico Público a Cartão Indutivo. Emissão 03/02/1994.
- [11] SDT 245-300-710 (PADRÃO) – Série Engenharia - Especificação do Aparelho Telefônico Público Celular a Cartão Indutivo. Emissão 01/10/1995.
- [12] TR13-UD3014-P-02. Equitel Telecomunicações. Módulo Cartão Indutivo.
- [13] Resolução Nº 334, de 16 de abril de 2003, ANATEL.
- [14] Mapa do Cartão Indutivo Codificado – emissão 02/04/1999.
- [15] SDT 245-300-708 (PADRÃO) – Série Engenharia - Especificação do Cartão Indutivo para Telefones Públicos e Semipúblicos. Emissão 02/12/1992.
- [16] Resolução Nº 327, de 13 de Dezembro de 2002, ANATEL.
- [17] http://www.anatel.gov.br/biblioteca/Contrato/Concessao/novos/contratos_novos.asp. Data de consulta em 05/08/03.
- [18] Resolução Nº 263, de 08 de junho de 2001, ANATEL.

Gerenciamento Integrado dos Recursos do Sistema para obter QoS Fim-a-Fim

Silvia Cristina Sardela Bianchi¹, Carlos Becker Westphall¹, André de Barros Sales², Michelle Sibilla², Carla Merkle Westphall¹

¹Laboratório de Redes e Gerência – Universidade Federal de Santa Catarina (UFSC)
Caixa Postal 476 – 88.049-970 – Florianópolis – SC – Brazil

²Institut de Recherche en Informatique de Toulouse - Université Paul Sabatier (UPS)
{silvia,westphal,carla}@lrg.ufsc.br, {barros,sibilla}@irit.fr

Abstract. *From the system manager viewpoint, a distributed system is composed of four layers: application, middleware, operating system and network. To reach an adequate quality of service it is possible to combine several standard technologies to provide an integrated management of heterogeneous distributed systems. This paper proposes a management model of system resources in an integrated management of distributed systems using WBEM architecture, CIM management information model, communication infrastructure CORBA and Java language to the application management development.*

Resumo. *Do ponto de vista da gerência de sistemas, um sistema distribuído é composto por quatro camadas: aplicação, middleware, sistema operacional e rede. Para atingir uma qualidade de serviço adequada pode-se combinar várias tecnologias para fornecer um gerenciamento integrado de sistemas distribuídos heterogêneos. Este artigo propõe um modelo de gerenciamento dos recursos do sistema sob o ponto de vista do gerenciamento integrado em um ambiente distribuído utilizando arquitetura WBEM, modelo CIM de informação, infra-estrutura de comunicação CORBA e Java como linguagem de programação para o desenvolvimento da aplicação de gerenciamento.*

1. Introdução

Com o aumento da disponibilidade, complexidade e heterogeneidade dos recursos nos sistemas distribuídos, surge a necessidade de novas técnicas de gerenciamento que possam garantir que as necessidades dos usuários sejam atendidas, proporcionando uma utilização eficiente dos recursos e de soluções dinâmicas [Abdu *et al* 2001].

Do ponto de vista de um gerente de sistemas as capacidades de gerenciamento em sistemas distribuídos são compostas por quatro camadas: aplicação, *middleware*, sistema operacional e rede. Em um ambiente distribuído as aplicações podem estar rodando em mais de uma máquina e podem precisar trocar mensagens ou dados entre elas. Para isso os dados são passados à camada de *middleware*, que os envia ao sistema operacional e deste para a rede. Cada camada possui pontos próprios que precisam ser gerenciados. Desta forma, cada uma demanda um gerenciamento específico e possui seus próprios parâmetros de qualidade de serviço.

Procurando um gerenciamento de QoS (qualidade de serviço) fim-a-fim, pesquisas têm sido feitas nas quatro camadas. As atividades de gerenciamento das camadas de Aplicação, *Middleware* e Rede têm sido realizadas no *Institut de Recherche en Informatique de Toulouse (IRIT) da Universidade Paul Sabatier* [Sibilla *et al* 2001] [Benech *et al* 2002].

Esse trabalho tem como objetivo abordar as atividades de gerenciamento relacionadas à camada do Sistema Operacional baseando-se no modelo CIM (*Common Information Model*) de informação [Dmtf 2003]. Assim, o artigo está organizado da seguinte forma: na seção 2 são apresentados gerenciamento integrado e os trabalhos correlatos; o modelo CIM de informação é apresentado brevemente na seção 3; na seção 4 é proposto um modelo de gerenciamento dos recursos do sistema para garantir uma qualidade de serviço adequada; em seguida, na seção 5, é mostrada a implementação e o ambiente experimental onde os testes foram realizados em um ambiente distribuído; e, finalmente, na seção 6 apresentamos as conclusões e trabalhos futuros.

2. Gerenciamento Integrado e Trabalhos Correlatos

Em um ambiente distribuído, a camada do *middleware* mascara a heterogeneidade de arquiteturas de computadores, sistemas operacionais, linguagens de programação e tecnologias de redes para facilitar o desenvolvimento de aplicações e gerenciamento.

O gerenciamento da aplicação necessita ter visibilidade sobre todos os recursos gerenciáveis em um ambiente distribuído. E, mais importante, necessita saber as dependências e relacionamentos pelos serviços fornecidos nas diferentes camadas.

Geiths (2001) tem sido um dos primeiros pesquisadores a apresentar questões para um gerenciamento integrado em arquiteturas de *middleware*. A fim de obter conhecimento do ambiente, o *middleware* deve ter informações das outras camadas, que permitirá a capacidade de suportar as garantias de qualidade de serviço como envio de mensagens e tratamento de erros.

Para obter a visibilidade sobre todos os recursos gerenciáveis em um ambiente distribuído heterogêneo e as suas dependências e relações, esses sistemas precisam de um modelo unificado para o gerenciamento dos recursos.

Um modelo de gerenciamento integrado e distribuído pode reduzir os custos e aumentar as capacidades de gerenciamento. Isto pode ser feito homogeneizando as informações de gerenciamento através da utilização de padrões e tecnologias orientadas a objeto.

Para que os usuários tenham um maior conforto, as tarefas devem ter um desempenho adequado com um mínimo de falhas e tempo de resposta adequado. Para atingir os requisitos específicos de cada comunidade de usuários, define-se esses requisitos pelo QoS (segurança, tempo de resposta, fácil de usar). Esses requisitos implicam em recursos gerenciáveis nos sistemas finais e nós da rede. O trabalho do Bacon [Bacon *et al* 2000] propõe extensões do *middleware* para resolver problemas como: eficiência, segurança, facilidade de uso, robustez e confiabilidade. Outra opção para resolver esses problemas seria integrar padrões para obter um desempenho melhor.

Os padrões tradicionais existentes são ineficientes na tentativa de cobrir todos os domínios de gerenciamento. Alguns deles são mais adequados em determinados níveis

de gerenciamento que outros. Mesmo sendo possível utilizar o SNMP (*Simple Network Management Protocol*) em quase todos os domínios, esse é mais apropriado para o gerenciamento da rede.

A iniciativa WBEM (*Web-Based Enterprise Management*) cobre todo o ambiente de gerenciamento, incluindo instrumentação, plataformas de gerenciamento, distribuição de dados, e fornece um modelo de dados compreensível, coerente e extensível. O maior problema no gerenciamento de sistemas é a existência de múltiplos *frameworks* que resultam em pouca integração. O WBEM não tem a intenção de substituir padrões existentes ou protocolos. O objetivo do WBEM é integrar com padrões existentes como SNMP, DMI, CMIP. Essa integração permite que qualquer aplicação de gerenciamento seja independente de específico API ou padrões usados para manusear entidades de gerenciamento [Schott *et al* 2002] [Thompson 1998].

Diversas alternativas propondo essa integração foram apresentadas. Um exemplo é a proposta de um *framework* para alcançar um gerenciamento global e distribuído integrando tecnologias WBEM/CIM, CORBA, SNMP para o gerenciamento da rede e Java [Benech *et al* 2000]. Outro trabalho é a possibilidade de integração do WBEM baseado em agentes, modelo de informação CIM e OSI para gerenciamento de aplicações e plataformas [Festor *et al* 1999].

No artigo proposto por Wunnana [Wunnava e Hambissa 2002] é realizado um estudo das tecnologias de gerenciamento da rede. Observa-se uma necessidade de integração do gerenciamento de redes corporativas e domésticas onde o nível de integração e escalabilidade é atingido através de padrões como WBEM, SNMP, CORBA e Java.

3. Modelo CIM de Informação

O WBEM define um modelo de informação, CIM, que fornece um padrão unificado com o conceito de *framework*, descrevendo objetos físicos e lógicos em um ambiente gerenciável. Para fornecer um *framework* comum, CIM define um conjunto de classes, propriedades, associações e métodos. O CIM define os seguintes modelos de informação:

Core Model: consiste no nível mais alto das classes, incluindo suas propriedades e associações.

Common Model: estende o *Core Model* para domínios gerenciáveis: sistema, aplicação, rede, dispositivos, físico. Incorpora noção comum para um domínio específico, independente de tecnologia ou implementação particular.

Extension Model: estende do *Common Model* para gerenciar ambientes específicos dependentes de tecnologia.

Assim, o CIM tem como objetivo modelar todos os ambientes de gerenciamento, definindo objetos e suas propriedades, métodos e associações em uma maneira uniforme. Assim, informações de redes, de dispositivos, de sistemas e de aplicações podem ser acessadas mais facilmente por uma aplicação de gerenciamento e tornar mais poderosas as capacidades de gerenciamento [Thompson 1998].

4. Gerenciamento dos Recursos do Sistema

Em um sistema distribuído o monitoramento dos recursos do sistema é essencial para um bom funcionamento do ambiente. Se uma aplicação faz uma requisição a um objeto remoto, ela fica aguardando até a chegada da resposta. Se, por algum motivo, ocorre um mau funcionamento do sistema, o objeto não consegue enviar a resposta à aplicação. Para garantir uma qualidade de serviço, alguns requisitos na gerência do sistema operacional são necessários como: controlar os recursos de espaço de disco, memória, CPU, swap, usuários e impressoras.

Na Figura 1 é apresentado o modelo de gerenciamento do sistema operacional em um ambiente distribuído. O modelo propõe a implementação de uma aplicação de gerenciamento que coleta informações dos recursos. As informações de gerenciamento são modeladas segundo o modelo CIM de informação para homogeneizar a informação e satisfazer os requisitos de unificação e integração dos sistemas de gerenciamento. Com base nas informações coletadas, a aplicação de gerenciamento pode efetuar um monitoramento dos recursos dos vários sistemas operacionais diferentes e enviar uma notificação, se necessária.

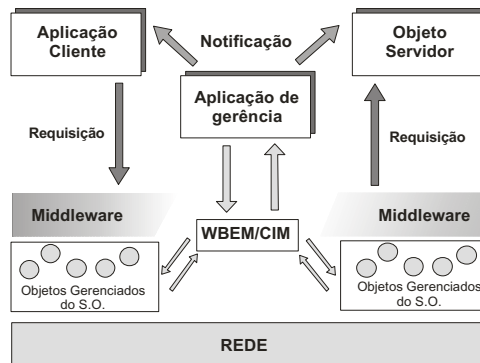


Figura 1. Modelo proposto de gerenciamento dos recursos do sistema.

O *CIM schema* define classes com propriedades e associações utilizando a sintaxe MOF (*Managed Object Format*). Assim, as informações dos objetos gerenciados são coletadas e modeladas segundo o *CIM schema*.

A especificação *System Specification 2.8* [System Specification 2.8 2003] define classes para o gerenciamento do sistema operacional e seus recursos. O envio da notificação é baseado na análise dos dados coletados. Para efetuar a notificação, a especificação *Event Specification 2.8* [Event Specification 2.8 2003] e a recomendação ITU/X.733 [Itu 1992] definem os tipos de notificação e em que circunstâncias elas devem ser enviadas.

5. Implementação e Testes

Para a implementação do modelo proposto foi utilizado o *Windows Management Instrumentation* (WMI) [Wmi Overview 1998] como implementação do WBEM e a

linguagem Java para o desenvolvimento da aplicação de gerenciamento. O WMI é uma implementação para plataformas Windows e permite o gerenciamento dos recursos como descrito no modelo.

Para a implementação do modelo proposto foram considerados dois estudos de caso. O primeiro estudo de caso é baseado no monitoramento da memória utilizada pelo processo e o segundo é o monitoramento do processo e envio de um alarme em caso de término do processo.

O monitoramento da memória é baseado no alarme enviado ao cliente quando a memória utilizada pelo processo está próxima do limite estabelecido pelo sistema operacional. Quando o cliente executa a aplicação, deve ser passado como parâmetro o tempo de *polling* para a coleta de informações dos recursos gerenciados. Quando a memória utilizada pelo processo está próxima de 90% do limite estabelecido pelo sistema operacional, um alarme é enviado para quem requisitou o serviço.

O segundo estudo de caso, um alarme é enviado quando um processo termina. O monitoramento do processo é realizado em ambos os lados do cliente e do servidor. Enquanto o processo está executando, informações sobre o processo são coletadas para monitorar o desempenho do processo.

Os testes foram realizados em um ambiente distribuído com plataforma Windows XP e como ORB usou-se o Visibroker. Como implementação do WBEM foi usado o *Windows Management Instrumentation* (WMI) para plataformas Microsoft. Utilizamos uma aplicação distribuída para testes que faz a conversão de *Euros*. Esta aplicação foi utilizada pelo laboratório *IRIT* da Universidade *Paul Sabatier* para a realização dos testes de gerenciamento das outras camadas em um ambiente distribuído.

6. Conclusão e Trabalhos Futuros

O monitoramento dos processos e o gerenciamento da memória utilizada por esses, permitiu uma melhoria no desempenho do sistema distribuído. Com o envio de um alarme quando a utilização da memória está próxima ao limite da utilização permitida pelo sistema operacional, é possível evitar o término do processo por indisponibilidade de memória para sua execução. Em uma situação em que o cliente espera por uma resposta de uma requisição a um objeto no servidor, se o processo no servidor terminou, ambos receberão um alerta e o cliente não ficará esperando pela resposta.

Dentro do modelo proposto de gerenciamento integrado de sistemas distribuídos é possível estender essa implementação para gerenciar recursos de outros sistemas operacionais. Implementações do WBEM para diferentes plataformas podem ser integradas a aplicação de gerenciamento permitindo o gerenciamento de recursos heterogêneos. Outra implementação da arquitetura WBEM é o WBEMServices 2.4 implementada pela Sun Microsystems [Sun Microsystems 2001]. Assim, como proposta de trabalhos futuros sugerimos a utilização dessas tecnologias para extensão da aplicação de gerenciamento, baseada no modelo de informação CIM, dos recursos do sistema em ambientes heterogêneos. Outra proposta é o gerenciamento de outros recursos do sistema, obtendo-se um gerenciamento do sistema operacional mais eficiente.

Referências

- Abdu, H., Lutfiyya, H. e Bauer, M. A. (2001) "Towards Efficient Resource Allocation In Distributed Systems Management", In: IPDPS-01, San Francisco, CA.
- Bacon, J., Moody, K., Bates, J., Hayton, R., Ma, C., Mcneil, A., Seidel, O. e Spiteri, M. (2000) "Generic Support for Distributed Applications", In: IEEE Computer, vol.33, n.3, p. 68-76.
- Benech, D., Jocteur-Monrozier, F. e Riviere, A. (2000) "Supervision of the CORBA Environment with SUMO: a WBEM/CIM-Based Management Framework", In: International Symposium on Distributed Objects and Applications, p. 241-250.
- Benech, D., Desprats, T., Sibilla, M., Sales, A. e Steff, Y. (2002) "Corba Management Services", Response to Request for Information, http://www.omg.org/techprocess/meetings/schedule/CORBA_Management_RFI.html.
- Dmtf. (2003) "Distributed management Task Force", <http://www.dmtf.org>
- Event Specification 2.8. (2003), CIM Schema: Version 2.8, http://www.dmtf.org/standards/documents/CIM/CIM_Schema28/CIM_Event28-Prelim.pdf.
- Festor, O., Festor, P., Youssef, B. e Andrey, L. (1999) "Integration of WBEM-based Management Agents in the OSI Framework", In: IFIP/IEEE International Symposium on Distributed Management for the Networked Millennium, p. 49-64.
- Geihs, K. (2001) "Middleware Challenges Ahead", In: IEEE Computer, vol.34, n.6 p. 24-31.
- Information Technology – Open Systems Interconnection – Systems Management: Alarm Reporting Function. ITU-T Recommendation X.733, 1992.
- Sibilla, M., Sales, A. B., Jocteur-Monrozier, F. e Riviere, A. (2001) "Towards End-To-End QoS: CIM/WBEM Management of CORBA QoS", In: Proceedings of HP Openview University Association, 8th Plenary Workshop, Berlin.
- Schott, J., Westerinen, A., Martin-Flatin, J. P. e Rivera, P. (2002) "Common Information vs. Information Overload", In: Network Operations and Management Symposium, Italy, p. 767-781.
- Sun Microsystems, Solaris WBEM Services 2.4. (2001), <http://www.sun.com/software/solaris/ds/ds-wbem24>.
- System Specification 2.8. (2003), CIM Schema: Version 2.8, http://www.dmtf.org/standards/documents/CIM/CIM_Schema28/CIM_System28-Prelim.pdf.
- Thompson, J. P. (1998) "Web-Based Enterprise Management Architecture", In: IEEE Communications Magazine, vol.36, n.3, p. 80-86.
- Wmi Overview. (1998) "Windows Management Instrumentation: Background and Overview". White Paper, <http://www.microsoft.com>.
- Wunnava, S. V. e Hambissa, Y. (2002) "Information Management Using the Centralized and Distributed Schemes" In: IEEE Press, p. 10-14.

BGP_e - Sub-Agente para Gerenciamento do Protocolo BGP

Andrey Vedana Andreoli¹, Leandro Márcio Bertholdo¹, Liane Tarouco¹,

Ana Benso da Silva² e Fábio Rodrigues²

¹Ponto de Presença da RNP no Rio Grande do Sul – POP-RS
RSiX – Ponto de Troca de Tráfego do Rio Grande do Sul
Centro de Processamento de Dados da Universidade Federal do Rio Grande do Sul
Rua Ramiro Barcelos, 2574 – Porto Alegre – RS – Brasil

²Faculdade de Informática – Universidade Católica do Rio Grande do Sul (PUCRS)
Avenida Ipiranga, 6681 – 91.501-970 – Porto Alegre – RS – Brasil
{andrey,berthold,liane}@penta.ufrgs.br, {benso,ji202379}@inf.pucrs.br

Resumo. *Este artigo descreve e analisa as necessidades adicionais do protocolo BGP no RSiX, demonstrando a implementação e validação de um sub-agente (BGP_e) que complementa o conjunto de informações importantes para uma gerência efetiva. Estes complementos podem auxiliar diretamente na operação de PTTs, possibilitando também sua utilização em pontos onde há grande concentração de peers BGP. Dentre os resultados obtidos, o artigo propõe modificações na MIB BGP para incluir as funcionalidades alcançadas pelo sub-agente BGP_e.*

Palavras chave: Roteamento, BGP, Pontos de Troca de Tráfego, Gerência de redes.

1. Informações Gerais

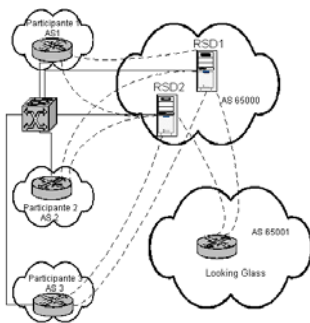
Com o grande crescimento da Internet desde seus primórdios, o modelo conhecido como hierárquico, constituído por uma rede centralizada que conectava redes secundárias, foi substituído por um modelo distribuído. A partir dessa mudança foram surgindo grandes redes de provedores comerciais, interconectadas através dos chamados Pontos de Troca de Tráfego (PTT). A necessidade desses pontos tem se tornado cada vez mais evidente, já que com o leque de aplicações que a Internet suporta hoje, o volume de tráfego agregado na ordem de gigabytes e o tempo de acesso tem se tornado um recurso valioso, sem contar a necessidade de minimizar os custos com conexões de longa distância.

A partir destas mudanças, grandes redes e backbones passaram a ser vistos perante a Internet como sistemas autônomos, também chamados de ASes. Tal definição faz com que o que ocorre internamente a um AS não seja conhecido pelos demais ASes, diminuindo a complexidade da Internet global. O protocolo que iniciou a utilização desse conceito e permanece até os dias de hoje é o BGP (Border Gateway Protocol)[RFC 1771].

Perante tais características, o número de PTTs tem aumentado consideravelmente. No Brasil existem mais de 6 PTTs, sendo que o primeiro destes foi implementado em 1998 pela ANSP e no ano de 2000 o RSiX [RSIX 2003] iniciou sua operação. Outros PTTs mais recentes também em operação, administrados pela RNP ou seus pontos de presença são o PRiX o FiX. Esse crescimento é reforçado pela orientação do Comitê Gestor da Internet-BR [CGIB 2000].

Os pontos de troca de tráfego são formados basicamente por um switch que atua como comutador, interligando roteadores de diferentes sistemas autônomos com o intuito de trocar tráfego. As sessões BGP entre os participantes de um sistema autônomo podem ser estabelecidas diretamente entre os participantes do PTT, formando uma relação de troca de tráfego bilateral, ou pode existir um componente no PTT chamado de Route Server. Tal componente faz com que as sessões BGP sejam estabelecidas diretamente com o Route Server e este anuncie aos demais ASes. Essa relação é definida como troca de tráfego multilateral, já que os Route Servers divulgam os anúncios a todos os participantes que pertencem a determinado Acordo de Tráfego Multilateral (ATM). Grande parte dos dos PTTs existentes no Brasil adota a política de troca de tráfego multilateral.

Na figura abaixo são apresentados dois Route Servers (RSD1 e RSD2), onde cada participante estabelece sessões BGP com estes afim de anunciar seus prefixos e receber os



anúncios dos demais participantes. Vale lembrar que apenas o tráfego do protocolo BGP passa pelos Route Servers. O tráfego trocado entre os participantes passa diretamente entre os roteadores dos participantes envolvidos na troca de tráfego. O componente Looking Glass (LG) é utilizado para verificar os anúncios e conectividade do PTT por parte dos participantes e por possíveis interessados na entrada do PTT. Em geral é permitida a consulta dessas informações via uma interface WEB.

Quanto ao acesso aos equipamentos do PTT, cada participante é responsável pela configuração e suporte ao roteador presente no PTT, não sendo acessível por parte da equipe de administração do

PTT. Neste contexto, apenas os Route Servers, o Switch e o Looking Glass estão sob o domínio da equipe de administração do PTT. Isso significa que as informações que poderiam ser utilizadas para a gerência do PTT devem ser coletadas em um destes três componentes. Cada um destes componentes possui as seguintes informações:

- Switch: Atuando em nível 2, este componente pode fornecer o volume de tráfego de cada participante, já que ele faz o papel de comutador da rede local do PTT. Essas informações são obtidas através da MIB II [RFC 1213].
- Looking Glass: Este componente possui todos os anúncios presentes no PTT, visto que possui sessões BGP com os dois Route Servers. Isso poderia ser obtido através da MIB-BGP [RFC 1657], caso o equipamento suporte tal recurso.
- Route Servers: Possui todas as informações que podem ser fornecidas pelo Looking Glass, com adição do estado das sessões BGP de cada participante, já que ele próprio estabelece sessões com todos os participantes. Estas informações também são acessíveis via MIB-BGP, desde que o componente suporte tal recurso. No caso do RSiX, é utilizado o software Zebra que tem suporte a MIB-BGP e torna possível o acesso a essas informações. Adicionalmente, já que ele pode implementar recursos de número máximo de prefixos, outros recursos poderiam ser explorados neste componente.

2. Necessidade de gerência de um PTT

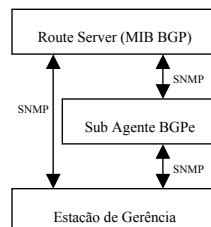
A partir da definição das possíveis fontes de informação da administração do PTT, enumeram-se algumas informações relevantes do protocolo BGP que poderiam ser utilizadas para a efetiva administração do PTT. Algumas delas podem ser encontradas em [KRAHE 2001]. São elas:

- Contabilização dos anúncios de cada participante e do número total de anúncios do PTT. Isso facilita a relação dos anúncios com o padrão de tráfego de cada participante, além de permitir que seja analisado o crescimento do PTT.
- Contabilização dos anúncios do PTT classificados pela sua máscara, afim de permitir que seja fornecido um panorama específico dos anúncios do PTT de cada participante e seu crescimento. Também auxilia o controle e histórico da ocorrência de anúncios muito específicos, de acordo com as regras do PTT.
- Monitoramento das sessões BGP como um todo, ou seja, contabilização do número de sessões agrupadas por seu estado, afim de identificar facilmente problemas isolados de problemas globais.

Através do acesso pela console dos Route Servers, parte dessas informações poderia ser obtida facilmente, mas a necessidade é que tais informações estejam disponíveis através de uma interface SNMP para permitir que diferentes ferramentas possam obtê-las e analisá-las, permitindo também a possibilidade de alguma forma de histórico destes valores.

3. Propondo uma solução

Analisando as fontes de informação e as necessidades listadas anteriormente, verifica-se que a MIB-BGP não fornece tais informações de forma direta. Entretanto, através da MIB BGP é possível obter todos os anúncios e atributos da tabela BGP, além do estado de cada sessão BGP. Baseado nessas informações, seria necessário processá-las para obter as informações desejadas. Surge então a necessidade de um sub-agente que faça acesso da MIB-BGP nos Route Servers e forneça via SNMP as informações desejadas, fazendo o processamento necessário. O sub-agente foi chamado de BGPe.



Os objetos fornecidos pelo sub-agente são apresentados abaixo:

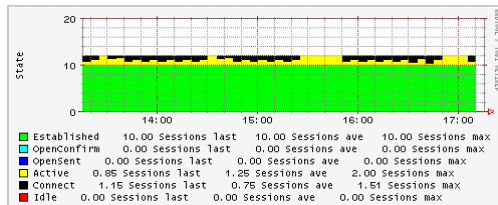
- BGPeTotalPathsGlobal: Total de anúncios globais ao PTT;
- BGPeLenGlobal: Total de anúncios do PTT agrupados por seu tamanho, a partir da especificação do tamanho do bloco desejado em notação CIDR.
- BGPeStates: Total de sessões BGP de um dos 6 estados possíveis definidos da máquina de estados do protocolo BGP [HALABI 2000].
- BGPeTotalPaths: Total de anúncios agrupados por cada participante do PTT.
- BGPeLen: Total de anúncios de determinado tamanho, agrupados também por cada participante do PTT.

A implementação deste sub-agente foi feita em linguagem C, adaptado como MIB estendida ao NET-SNMP [NETSNMP 2003].

4. Resultados obtidos

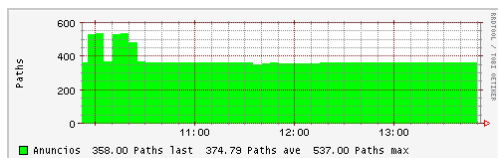
A partir da utilização deste sub-agente, através do software RRDTOol [RRDTOOL 2003] foi possível criar alguns gráficos que vem sendo utilizados para a administração do RSiX atualmente. Tais informações têm auxiliado em 90% dos problemas envolvendo participantes do PTT. Abaixo são exibidos alguns exemplos:

Número de sessões BGP agrupadas por seu status.



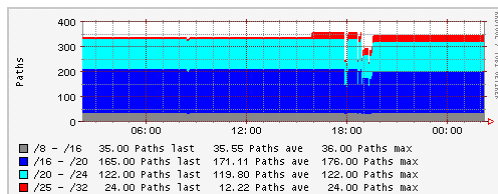
Tal representação é muito útil pois reflete o estado das sessões BGP de todos os peers de forma sintetizada, demonstrando facilmente tanto problemas específicos a apenas um peer, como problemas envolvendo mais de um peer.

Total de anúncios da tabela BGP.



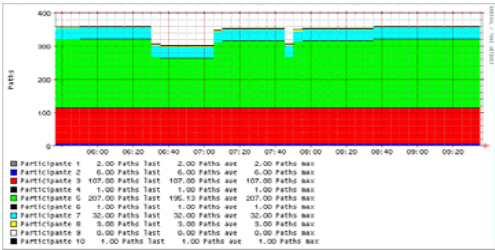
O total de anúncios da tabela BGP, principalmente em PTTs é muito importante, pois a estabilidade de tal informação reflete na maioria dos casos a estabilidade do BGP e do backbone de seus peers. Mudanças nesses números podem servir de justificativa para mudanças no tráfego e, se não esperadas, podem ser indícios de problemas. O gráfico acima reflete dois picos de anúncios que na ocasião foram provocados por problemas na configuração de filtros de um dos peers.

Contabilização de anúncios BGP de acordo com sua profundidade.



Da mesma forma que é desejável saber o número de anúncios, é desejável obter de forma gráfica informações mais específicas, como os tamanhos dos anúncios. Isso tem grande valor pois de forma muito frequente surgem anúncios muito específicos que excedem o tamanho máximo permitido. No caso da ocasião ilustrada na figura acima, um peer deixou passar por seus filtros anúncios mais específicos que, na notação CIDR, seriam maiores que /25. Estrategicamente foi escolhida a cor vermelha para esses anúncios, afim de rapidamente serem identificados.

Total de anúncios de cada participante do PTT



Da mesma forma que o total de anúncios global é útil, surge a necessidade também de se ter um controle de número de anúncios específico para cada participante. Assim torna-se possível saber qual participante tem oscilado ou apresentado problemas em seus anúncios. Sendo assim, o troubleshooting já inicia sabendo qual o peer que está apresentando problemas e qual sua variação de anúncios, sem citar o histórico que a representação gráfica pode fornecer para todos os gráficos apresentados.

Diversos outros gráficos de diferentes objetos podem ser montados a partir das informações fornecidas pelo sub-agente, dependendo da necessidade de gerência e dos pontos mais críticos de cada ponto BGP.

4. Proposta de modificação da MIB BGP

Atualmente, os equipamentos que implementam o protocolo BGP tem condições de fornecer, além dos dados da MIB BGP, as informações adicionais demonstradas no sub-agente, porém apenas através da console. Esse detalhe parece desprezível, mas é exatamente o ponto onde o presente artigo busca ressaltar, já que esse detalhe inviabiliza diversas formas de gerência e monitoração, como é o caso do SNMP. O custo para incluir as funcionalidades do sub-agente BGPe na MIB BGP é relativamente simples, visto que os referidos equipamentos já manipulam as mensagens do protocolo e a tabela de anúncios BGP, já possuindo as informações, bastando apenas instancia-las como objeto da MIB.

Tais mudanças não provocariam nenhum grande impacto na performance do produto e principalmente tornaria a MIB BGP uma opção ainda mais consolidada na gerência efetiva do protocolo BGP, fornecendo ainda mais objetos úteis para tal tarefa.

Não se pretende incluir um grande número de objetos na MIB, pois não é o objetivo desta MIB fornecer um mecanismo único para gerência do BGP, mas apenas disponibilizar os objetos demonstrados no sub-agente. Dessa forma, a MIB BGP poderá ser um recurso ainda mais útil para o monitoramento do BGP, sem a adição de nenhum outro componente na gerência. Tais informações mostrarão de forma muito contundente as mudanças no roteamento BGP, sinalizando a equipe de gerência para tomar as ações necessárias.

5. Conclusões

Como conclusões deste estudo e validação, pode-se afirmar que o sub-agente tem exercido um papel importante no auxílio na gerência do protocolo BGP principalmente no RSiX. Os gráficos e monitorações permitidas pelo sub-agente têm exercido um papel fundamental na detecção de anormalidades no roteamento BGP. Uma vez que as anormalidades são detectadas, os procedimentos convencionais de troubleshooting e verificação mais

detalhada da anormalidade devem ser tratados pela equipe de suporte, não pertencendo mais ao escopo deste trabalho.

Na implementação e validação, observou-se que ação do sub-agente BGPe sobre a MIB BGP apresenta um overhead, já que busca dos dados da tabela BGP é feita via SNMP. Esse tempo de processamento aumenta de acordo com o tamanho da tabela de roteamento BGP. Já o overhead para o processamento destes dados é desprezível. Essa é uma limitação que irá existir até que as funcionalidades do sub-agente não estejam incorporadas a MIB BGP. Uma vez que as funcionalidades do sub-agente sejam incluídas na MIB BGP, a limitação de processamento torna-se também desprezível, pois os dados já estarão sendo contabilizados, restando apenas repassar as informações solicitadas ao requisitante.

Por fim, vale ressaltar que as informações identificadas e propostas pelo sub-agente não beneficiam apenas Pontos de Troca de Tráfego [ANDREOLI 2003], mas qualquer ponto onde há concentração de peers BGP. O valor de tais informações de fato é de grande valia para a gerência efetiva em nível de roteamento BGP.

6. Referências bibliográficas

- [RFC 1771] A Border Gateway Protocol 4 (BGP-4) – RFC 1771 – on line – 2003 - <http://www.ietf.org/rfc/rfc1771.txt>
- [RSIX 2003] RSIX - Ponto de Troca de Tráfego Internet – on line - 2003 – www.rsix.tc.br
- [CGIB 2000] Comitê Gestor Internet/BR – Operação e Administração de PTTs no Brasil – on line – 2000 - http://www.cg.org.br/grupo/operacao_ptt_v1.1.htm
- [RFC 1213] Management Information Base for Network Management of TCP/IP-based internets: MIB II - RFC 1213 – on line - <http://www.faqs.org/ftp/rfc/rfc1213.txt>
- [RFC 1657] Definitions of Managed Objects for the Fourth Version of the Border Gateway Protocol (BGP-4) using SMIPv2 – RFC 1657 - on line – 2003 - <http://www.faqs.org/ftp/rfc/rfc1657.txt>
- [KRAHE 2001] Segurança em Pontos de Troca de Tráfego – 14ª Reunião do Grupo de Trabalho em Engenharia de Redes – GTER 14 – 2001 – on line - [ftp://ftp.registro.br/pub/gter/gter14/aspectos_sec_ptt_bertholdo.ppt](http://ftp.registro.br/pub/gter/gter14/aspectos_sec_ptt_bertholdo.ppt)
- [HALABI 2000] Sam Halabi, Danny McPerson. Internet Routing Architectures, Second Edition. Indianapolis – USA : Cisco Press, 2000
- [NETSNMP 2003] NetSNMP Software – The Net SNMP Project Home Page – on line – 2003 – <http://www.netsnmp.org>
- [RRDTOOL 2003] RRD Tool Software – on line – 2003 – <http://www.rrdtool.com>
- [ANDREOLI 2003] Controle do Protocolo BGP em PTT's – 15ª Reunião do Grupo de Trabalho em Engenharia de Redes – GTER 15 – 2003 – on line – [ftp://ftp.registro.br/pub/gter/gter15/gter15-bgpe-rsux.pdf](http://ftp.registro.br/pub/gter/gter15/gter15-bgpe-rsux.pdf)

Monitoração de Disponibilidade e Desempenho de Servidores Críticos usando uma Abordagem Descentralizada

Juliano Valentini, Luciano Paschoal Gasparly

¹Programa Interdisciplinar de Pós-Graduação em Computação Aplicada – PIPCA
Universidade do Vale do Rio dos Sinos – UNISINOS
Av. Unisinos, 500 – CEP 93.022-000 – São Leopoldo, RS

{julianov, paschoal}@exatas.unisinos.br

Abstract. *Due to the growth of size and number of services provided by current computer networks, the centralized management paradigm tends to lead to a lot of management traffic and to the overload of the central management station (when a large number of devices is supposed to be monitored). Some approaches have been proposed by the scientific community to solve this problem. This paper describes the usage of Script MIB, proposed by the IETF, to monitor in a decentralized fashion the availability and performance of critical servers.*

Resumo. *Devido ao aumento do tamanho e da quantidade de serviços oferecidos pelas redes de computadores atuais, o paradigma centralizado de gerenciamento tende a provocar um aumento do tráfego de gerenciamento e a sobrecarga da estação central (quando um grande número de dispositivos deve ser monitorado). Para amenizar essa problemática, algumas abordagens de gerenciamento distribuídas têm sido propostas pela comunidade científica. Este artigo descreve a utilização da MIB Script, proposta pelo IETF, para a monitoração descentralizada de disponibilidade e desempenho de servidores críticos.*

1. Introdução

As redes de computadores estão passando por um grande crescimento no tamanho e no número de serviços que oferecem. Normalmente são gerenciadas por um sistema central, que é responsável por controlar um grande número de elementos. Com o acréscimo do tamanho da rede, o gerente central acaba ficando sobrecarregado, a ponto de não ser mais capaz de monitorar todos os elementos da mesma. O problema da escalabilidade torna-se pior quando ocorre um problema e, além de continuar com a monitoração, o gerente precisa executar procedimentos de recuperação, sobrecarregando-o ainda mais. Outra limitação de um sistema centralizado está relacionada com o tráfego de gerenciamento, que muitas vezes é enviado de uma ponta a outra da rede, consumindo banda (em geral escassa) dos canais de longa distância.

Este trabalho relata um experimento utilizando uma solução alternativa ao gerenciamento centralizado. Se os problemas do gerenciamento centralizado são a sobrecarga do gerente central e o tráfego de gerenciamento, o que se busca é distribuir as tarefas de gerenciamento entre várias entidades da rede. Nesse contexto, o presente trabalho descreve a utilização da MIB Script [3], proposta pelo IETF, para a monitoração descentralizada de disponibilidade e desempenho de servidores críticos.

O artigo está organizado da seguinte forma: a seção 2 descreve alguns conceitos e classificações de gerenciamento distribuído. A seção 3 revisa a MIB Script, incluindo sua

estrutura e forma de utilização. Na seção 4 descreve-se o experimento realizado. O artigo encerra na seção 5 com as considerações finais.

2. Gerenciamento Distribuído

Uma solução de gerenciamento distribuído de redes, em geral, é composta por gerentes, agentes e entidades que desempenham ambos os papéis, chamadas de gerentes intermediários. Os gerentes são responsáveis pela delegação de tarefas de gerenciamento da rede. Os gerentes intermediários recebem essas tarefas e são responsáveis pela execução das mesmas. Os agentes monitoram variáveis e contabilizam informações sobre as estações da rede, que são armazenadas em uma MIB (*Management Information Base*). Os gerentes podem consultar informações ou receber alertas dos gerentes intermediários sobre as tarefas que foram delegadas e estão sendo executadas.

A figura 1 ilustra que as definições acima (gerente, gerente intermediário e agente) implicam uma estrutura hierárquica do sistema de gerenciamento. No topo da hierarquia está o gerente. Embaixo encontram-se os agentes. Os gerentes intermediários ficam entre o gerente e os agentes. Na próxima seção serão apresentadas as classes de sistemas de gerenciamento, onde ficará evidenciado que nem sempre é necessário instanciar todas essas entidades.

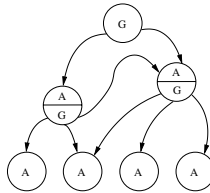


Figura 1: Ilustração de um sistema de gerenciamento. Círculo com a letra G representa um gerente e círculo com a letra A, um agente. Os círculos divididos são os gerentes intermediários.

2.1. Classes de Sistemas de Gerenciamento de Rede

Os sistemas de gerenciamento de rede podem ser classificados de acordo com o número de gerentes, gerentes intermediários e agentes que possuem. A seguir, são apresentadas quatro classes de sistemas de gerenciamento de rede, extraídas de [5]:

- $m = 1$: gerenciamento centralizado
- $1 < m \ll n$: gerenciamento fracamente distribuído
- $1 \ll m < n$: gerenciamento fortemente distribuído
- $m \approx n$: gerenciamento cooperativo

No esquema acima, m representa o número total de gerentes e gerentes intermediários e n o número total de elementos no sistema de gerenciamento, ou seja, a soma de m e o número de agentes.

Para sair do gerenciamento centralizado em direção ao cooperativo, aumenta-se o grau de complexidade do sistema de gerenciamento. Para que seja possível a descentralização é necessário um mecanismo que permita a distribuição das tarefas. Atualmente, a comunidade tem investigado muitas dessas tecnologias [1, 2, 8]. Nas seções

seguintes, será abordado uma dessas tecnologias – a MIB Script – utilizada como peça-chave da nossa arquitetura de gerenciamento por ser padronizada pelo IETF e compatível com SNMP (*Simple Network Management Protocol*).

3. MIB Script

A MIB Script [3] é um conjunto de objetos de gerenciamento que permite a delegação de *scripts* de gerenciamento para gerentes distribuídos fornecendo os seguintes atributos:

- transferência de *scripts* de gerenciamento para locais distribuídos;
- inicialização, suspensão, reinicialização e finalização dos *scripts* nesses locais;
- passagem de parâmetros para os *scripts* de gerenciamento;
- monitoração e controle dos *scripts* de gerenciamento que estão executando;
- transferência para o gerente dos resultados produzidos pelos *scripts* de gerenciamento que foram executados.

Os *scripts* a serem executados podem ser escritos em qualquer linguagem de programação, compilada ou interpretada, desde que esta seja suportada pela MIB Script. É possível estender a MIB Script de forma a suportar uma determinada linguagem que não é suportada nativamente.

3.1. Uso da MIB Script

Observando a figura 2, a partir da estação de gerenciamento (1), o gerente de rede define *scripts* de monitoração e de ação. Esses *scripts* são transmitidos a um servidor Web (2) que funciona como um repositório de *scripts* (3). O gerente deverá, ainda, instalar os *scripts* que deseja executar na MIB Script via requisições SNMP (4). Nessas requisições são enviadas ao agente SNMP informações sobre os *scripts* como: nome, parâmetros, URL do *script*, linguagem em que foi escrito, descrição, entre outros. A MIB Script faz *download* do *script* (5, 6). Instalado o *script*, o gerente poderá, através de uma requisição SNMP (4), disparar e controlar a execução do mesmo.

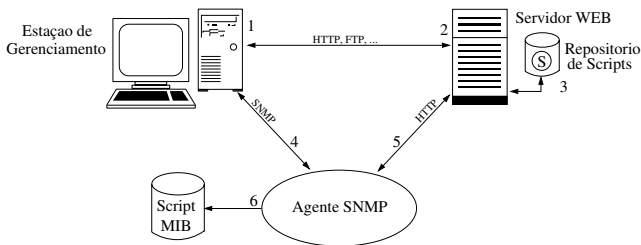


Figura 2: Uso da MIB Script

3.2. Cenários de Uso

Nesta seção serão descritos alguns cenários de gerenciamento distribuído que poderiam ser implantados com base na MIB Script. Um dos objetivos da MIB Script é servir como suporte a gerentes intermediários em um sistema de gerenciamento distribuído. A idéia é implementar funcionalidades de gerentes intermediários através de *scripts* e executá-los dinamicamente em segmentos de rede remotos.

1. *Scripts para monitoração*: um gerente intermediário poderia monitorar um conjunto de variáveis de uma MIB em vários elementos da rede com o objetivo de detectar comportamentos irregulares ou para fins de contabilizar informações. O gerente intermediário poderia ser consultado pelo gerente ou reportar os resultados a ele enviando notificações.
2. *Scripts para teste de serviços*: um cenário empregando um gerente intermediário mais complexo é o que realiza teste de serviços. Esses testes incluem verificação da disponibilidade do serviço e os parâmetros estáticos e dinâmicos passados ao serviço. Uma maneira para implementar o teste serviços é simular o comportamento de usuários. Isso pode ser útil, por exemplo, quando a acessibilidade de certos documentos de um servidor web é essencial. Um *script* pode ser instalado em muitos locais remotos para verificar a disponibilidade e tempo de resposta regularmente e notificar o gerente em caso de falha.
3. *Scripts para gerenciamento e controle de serviços*: um serviço de rede pode ser implementado para executar processos de forma distribuída em várias instâncias da rede. Esses processos podem ser controlados por um gerente intermediário local executando no mesmo segmento de rede que o processo. Esses gerentes intermediários podem iniciar todos os processos pertencentes ao serviço, receber mensagens de erro desses processos, monitorar suas operações, reiniciar ou abortar o processo, se necessário, e notificar o gerente em caso de problemas sérios.

4. Ambiente de Monitoração Configurado

O ambiente configurado tem como objetivo realizar a monitoração de servidores críticos de uma rede corporativa. A idéia é distribuir gerentes intermediários em locais estratégicos de forma que o gerente central possa delegar tarefas a eles relacionadas à análise de disponibilidade e desempenho desses servidores. Assim, esse gerente central passa a receber somente informações que são relevantes (por exemplo, notificação de que um serviço está indisponível).

Para que o gerente intermediário pudesse receber *scripts* do gerente central e executá-los foi preciso instalar dois *softwares* principais nas estações que desempenham esse papel. O primeiro foi o NET-SNMP [4], agente SNMP responsável por processar requisições e enviar respostas. O segundo foi o *Jasmin* [6], uma implementação da MIB Script. Por outro lado, para obter as informações dos servidores críticos foi utilizado um agente SNMP que implementa a MIB Host Resources (em cada servidor), definida em [7].

A MIB Host Resources define um conjunto de objetos para gerenciamento de computadores. As informações armazenadas por essa MIB são independentes de sistema operacional, serviços de rede ou qualquer programa aplicativo. Os dados estão organizados em tabelas que contêm informações gerais sobre o sistema, armazenamento, dispositivos instalados, programas e outras informações relevantes ao administrador de sistema. Dispondo das informações da MIB Host Resources, acessadas remotamente via requisições SNMP, o gerente pode tomar decisões em relação ao *host*.

O experimento realizado monitora as variáveis *hrSystemUptime* que acumula o tempo desde a última vez que o *host* foi inicializado, e *hrProcessorLoad* que mantém a porcentagem de carga média do processador durante o último minuto. Para realizar essa monitoração foi criado um *script* Java, ilustrado na figura 3, que gera um alarme caso detecte que o sistema foi reiniciado ou quando o processador ficou com uma carga superior a 90%.

```
{
  ...

  SnmpPeer peer = new SnmpPeer("YODA",
                                InetAddress.getByName("200.132.73.112"),
                                "public" );
  SnmpConnection connection = new SnmpConnection(peer);
  Vector varbinds = new Vector();
  OID oid = new OID("1.3.6.1.2.1.25.1.1");
  varbinds.addElement(oid);
  oid = new OID("1.3.6.1.2.1.25.3.1.2");
  varbinds.addElement(oid);
  while (true) {
    Varbind[] result = connection.getNextRequest(varbinds);
    if (result == null) {
      out.println("No response");
    }
    else {
      alarm1.clear();
      alarm1 = getAlarm("reboot");
      newUptime = Double.parseDouble(result[0].value.content);
      if (newUptime < oldUptime) {
        alarm1.update("Host has been rebooted"); }
      else {
        alarm1.clear(); }
      oldUptime = newUptime;
      alarm2.clear();
      alarm2 = getAlarm("overload");
      ProcessorLoad = Integer.parseInt(result[1].value.content);
      if (ProcessorLoad > 90) {
        alarm2.update("Host has been overloaded"); }
      else {
        alarm2.clear(); }
    }
  }
}
```

Figura 3: Fragmento do *script* Java executado pelo gerente intermediário

Conforme pode ser observado na figura 4, a partir da estação de gerenciamento (1) o *script* de monitoração foi instalado e, em seguida, inicializado na MIB Script do gerente intermediário (2). Neste momento, o *host* (3) passou a ser monitorado continuamente pelo gerente intermediário.

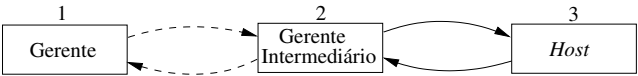


Figura 4: Ilustração do cenário do experimento. Configuração do gerente: CPU Intel Pentium 4 1,7GHz com 512MB de RAM e sistema operacional Linux Slackware 8.1; Configuração do gerente intermediário: CPU Intel Pentium 1 200MHz com 48MB de RAM e sistema operacional Linux Slackware 7.1; Configuração do *Host*: CPU Intel Pentium 4 1,7GHz com 512MB de RAM e sistema operacional Windows XP. As linhas contínuas representam tráfego constante e as linhas tracejadas representam tráfego eventual.

Durante o experimento, foi medido o tamanho das requisições SNMP. Observou-se que para monitorar a variável *hrSystemUptime*, por exemplo, consome-se 168 bytes (82 bytes para GET e 86 bytes para RESPONSE). Fazendo uma projeção, em uma abordagem de gerenciamento centralizado, de um cenário onde o gerente central é responsável pela monitoração de 120 servidores e que, em cada servidor, devem ser monitoradas 20 variáveis do mesmo tamanho, chega-se a um total de 403.200 bytes de tráfego por ciclo de monitoração. Em uma abordagem de gerenciamento distribuído, introduzindo-se 6 gerentes intermediários, os quais, cada um, ficaria responsável por gerenciar 20 dos 120 servidores, obtém-se uma redução de aproximadamente 83,34% do tráfego entre os gerentes intermediários e os servidores (*hosts*), ou seja, carga dividida entre os 6 gerentes

intermediários. Já o tráfego entre o gerente central e os gerentes intermediários pode ser considerado desprezível, tendo em vista que o gerente central faz apenas duas requisições. Uma para instalar os *scripts* de monitoração e de ação na MIB Script e outra para disparar a execução dos mesmos. Quando ocorre algum problema na rede o gerente central apenas recebe um alerta.

A sobrecarga originalmente concentrada na estação de gerenciamento é, agora, distribuída entre os gerentes intermediários. A carga de cada gerente intermediário está diretamente relacionada com o número de tarefas que lhe forem delegadas. Assim, é importante destacar a relevância da organização dos domínios de gerenciamento, a fim de balancear as tarefas a serem executadas entre os gerentes intermediários e evitar que estes sejam sobrecarregados.

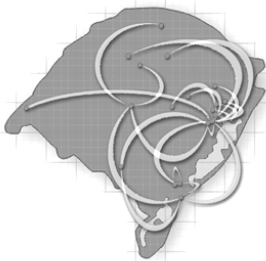
5. Conclusões

O processo de gerenciamento de grandes de redes utilizando gerenciamento centralizado é uma tarefa cada vez mais complexa, tendo em vista o grande conjunto de serviços e o volume de tráfego que o gerenciamento demanda. Existe, também, a dificuldade de uma única estação dar conta de processar toda as requisições de monitoração e, ainda, executar as tarefas de reparo. Este artigo descreveu um experimento envolvendo o uso da MIB Script para implantar tarefas de gerenciamento descentralizadas.

A maior dificuldade encontrada neste trabalho foi a instalação e a configuração da MIB Script – o Jasmin. O projeto desta implementação foi concluído no ano de 2001. Portanto, é necessário configurar um ambiente utilizando os *softwares* e tecnologias daquela época, entre elas o JDK 1.1, por exemplo. Apesar da vasta documentação teórica sobre o Jasmin, a documentação da parte prática deixa a desejar, tornando-se difícil saber exatamente que versões de *software* são necessárias. A MIB Script necessita, ainda, do UCD-SNMP, que atualmente foi migrado para o NET-SNMP. Por tudo isso, conclui-se que o Jasmin precisa passar por uma grande atualização permitindo o seu uso junto com as novas tecnologias que surgiram desde então. Ao contrário disso, o seu uso acabará se tornando incompatível e inviável.

Referências

- [1] GOLDSZMIDT, German S. **Distributed Management by Delegation**. 1996. Ph.D Thesis – Graduate School of Arts and Sciences, Columbia University, New York.
- [2] KAHANI, M. and BEADLE, H.W. P. Decentralized Approaches for Network Management. **Computer Communications Review**, v. 27, n. 3, p. 36-47, July 1997.
- [3] LEVI, D. and SCHÖNWÄLDER, Jürgen. **Definitions of Managed Objects for the Delegation of Management Scripts**, Nortel Networks, TU Braunschweig, RFC 3165, August 2001. Disponível em <<http://www.ietf.org>>.
- [4] NET-SNMP. **NET-SNMP Project Homepage**. Disponível em <<http://net-snmp.sourceforge.net>>.
- [5] SCHÖNWÄLDER, J. et al. Building Distributed Management Applications with the IETF Script MIB. **IEEE Journal on Selected Areas in Communications**, New York, v. 18, n. 5, p. 702-714, March 2000.
- [6] TU BRAUNSCHWEIG, NEC C&C RESEARCH LABORATORIES. **Jasmin – A Script MIB Implementation**. Disponível em <<http://www.ibr.cu.tu-bs.de/projects/jasmin>>.
- [7] WALDBUSSER, S., GRILLO, P. **Host Resources MIB**, Lucent Technologies Inc., WeSync.com, RFC 2790, March 2000. Disponível em <<http://www.ietf.org>>.
- [8] SIEGL, M., TRAUSMUTH, G. Hierarchical Network Management: A Concept and Its Prototype in SNMPv2. **Computer Networks and ISDN Systems**, v. 28, n. 4, p. 441-452, 1996.



Sessão Técnica 2

Aplicações e Estudos de Caso

Security Site – Desenvolvendo um Ambiente Seguro para E-Commerce e E-Business

Thiago de Moraes Pereira, Prof. MSc Alessandro de Castro Borges

Departamento de Computação – Universidade do Estado de Minas Gerais (UEMG)
Caixa Postal 03 – 37.900-106 – Passos – MG – Brazil

{thiagomp, acborges}@passosuemg.br

Abstract. *This paper describes a propose for a security policy in order to provide confidentiality in e-commerce and e-business transactions, with high security patterns for total control on transfered informations. The pretended solution is suited for electronic commerce, but it works too for enterpriser data transference on the Internet, or either on a Intranet and Extranet either. The presented project is on test and was developed with VPN, SSL, S-MIME, FIREWALL and ANTI-VIRUS technology.*

Resumo. *Este artigo descreve uma proposta para o desenvolvimento de uma política de segurança com o objetivo de prover confidencialidade em transações de e-commerce e e-business, atendendo aos mais altos padrões de segurança visando o controle total das informações. A solução a ser abordada não se aplica apenas ao e-commerce, mas também às empresas que necessitam fazer transferência de dados pela Internet ou até mesmo por uma Intranet e Extranet. O projeto apresentado está em fase de testes e foi desenvolvido com as tecnologias VPN, SSL, S-MIME, FIREWALL, ANTI-VÍRUS.*

1 Introdução

Com o crescimento da Internet as empresas, de modo geral, perceberam que poderiam usufruir desse ambiente para comercializar seus produtos globalmente e assim expandir seus negócios de forma abrangente, transferindo parte, ou quase a totalidade, de suas tarefas para a Internet.

Este artigo descreve o projeto Security Site, destinado a tornar seguras transações envolvendo valores, permitindo relacionamentos sem danos entre corporações e seus clientes.

1.1 Motivação

A área de segurança relacionada ao e-commerce ainda é muito incipiente no Brasil. Neste trabalho o foco não é dado apenas à venda dos produtos, mas a todos os aspectos correlacionados envolvendo clientes, comerciantes, bancos e serviços prestados. Foi desenvolvida uma proposta para solucionar deficiências de segurança no e-commerce, aumentando o grau de confiança entre clientes, comerciantes e bancos quando efetuam compras, vendas, pagamentos ou entregas pela Internet.

A solução abordada não se aplica apenas ao e-commerce, mas também às empresas que necessitam fazer transferência de dados via Internet, Intranet e Extranet.

1.2 Objetivos

O objetivo deste projeto é desenvolver uma política de segurança sólida com aplicação no e-commerce, incluindo procedimentos usados para implementar segurança em transações on-line. Outro aspecto importante é o levantamento de aspectos referentes ao tema e as variadas políticas desenvolvidas para prover segurança nas corporações.

Este trabalho solucionou divergências com relação à segurança no e-commerce e e-business. Apresentamos uma proposta de segurança às empresas que necessitam fazer transferência pela Internet com tranquilidade e credibilidade no envio dos seus dados.

2 Projeto Security Site

O Security Site foi proposto como um trabalho para prover segurança a outro projeto de iniciação científica desenvolvido na instituição, com objetivo principal de focar todos os pontos necessários na construção de um e-commerce rentável, seguro e acessível às empresas de nossa região (a maioria de pequeno porte), aderindo à idéia de atingir um público maior, disponibilizando suas mercadorias na Internet.

O sucesso de uma organização depende em muito da maneira que trata suas informações com relação à segurança e da forma que disponibiliza seus dados. Pensando nisso, propomos uma política de segurança eficiente, eficaz e barata. Foi feita uma pesquisa minuciosa sobre os procedimentos utilizados em sites comerciais. A idéia é garantir a segurança das informações usando tecnologias vigentes no mercado.

2.1 Tecnologias Utilizadas

Foram pesquisadas várias formas de tornar um site de e-commerce seguro, levando em conta também testes em diferentes tecnologias. Abaixo estão relacionadas as que apresentaram melhores índices de credibilidade:

- **SSL** (Secure Sockets Layer) - interessante devido à sua funcionalidade em criar conexões “criptografadas” entre o servidor web e o browser, provendo segurança entre o TCP/IP e os protocolos de aplicação [Terada 2000]; [Gomes 1995].
- **VPN** (Virtual Private Network) - conexões seguras baseadas em tecnologias de criptografia, autenticação e tunelamento, [Scott 1999].
- **S-MIME** (Secure Multimedia Internet Mail Extensions) - Para um e-commerce bem sucedido, é essencial o uso do correio eletrônico no envio de mensagens entre empresas e clientes. Com base nessa idéia, a pesquisa apontou o S-MIME como uma ótima ferramenta para segurança de e-mail, pois explora uma sintaxe de mensagem “criptografada” num ambiente de Internet MIME [Smith 1997].
- **FIREWALLS** - garante integridade das informações. Impõe uma barreira entre a rede privada da organização e a rede externa, baseado na combinação de hardware e software ou somente em software [Cooper 2001], [Siyan 1995].

3 Descrição do Projeto

A proposta desenvolvida possui como um de seus critérios o poder aquisitivo de cada empresa, restringindo o nível das tecnologias e formas de proteção disponíveis de acordo com seus valores.

Em linhas gerais, o projeto coordena desde o envio das informações pelo cliente até a segurança dos dados armazenados em um determinado servidor.

3.1 Arquitetura

Com base no levantamento realizado, foi implementado um firewall para a proteção dos servidores do projeto, barrando a entrada de pacotes não identificados na rede. Na máquina firewall foram implementadas restrições acentuadas com relação à entrada dos pacotes, pois nessa máquina somente podem trafegar pacotes relacionados à porta 80, restringindo conexões de SSH, TELNET, FTP, SMTP, RIP, entre outras.

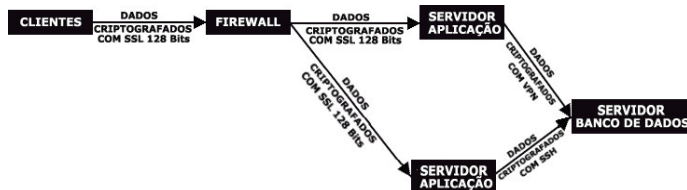


Figura 1. Diagrama de Fluxo de Dados – Tráfego da informação criptografada

Foram implementadas também no firewall, restrições quanto a defensivas contra “ping da morte”, “spoofing”, entre outras que poderiam colocar em risco o bom funcionamento do projeto como um todo. Nessa máquina também foi implementado um sistema de balanceamento de carga entre os dois servidores de aplicação utilizados, ou seja, quando um servidor está com uma grande quantidade de serviço, requisições são redirecionadas para o segundo servidor de aplicação. Outra vantagem desse sistema se dá no caso de “negação de serviço” por um dos servidores de aplicação, quando então o outro começa imediatamente a responder todas as requisições (Tabela 1, Figura 2).

Para avaliar a aplicabilidade do balanceamento de carga, foram efetuadas grandes quantidades de requisições aos servidores de aplicação, onde os mesmos responderam de forma balanceada e dividida. Outra avaliação feita com relação ao balanceamento de carga foi a negação de serviço, onde em funcionamento um dos servidores foi reiniciado. Imediatamente o outro servidor de aplicação começou a responder as requisições que estavam sendo processadas pelo servidor reiniciado (Tabela 1).

Em resumo, quando os dois servidores estão funcionando em paralelo, a aplicação de e-commerce pode atender a uma quantidade de conexões elevada, pois se um servidor web disponibiliza uma quantidade de conexões por servidor, o balanceamento de carga irá dobrar o número de conexões, trazendo então uma maior rapidez e eficiência na disponibilização dos produtos de uma determinada empresa.

O projeto foi implementado e testado em uma rede broadcast mantendo a preocupação com relação à utilização de sniffers, pois qualquer computador em modo promíscuo poderia capturar pacotes sendo transferidos pelos servidores. Para sanar esse problema, a política de segurança descreveu a utilização de VPN e SSH entre os servidores envolvidos. Nesse caso, mesmo que o sniffer capturasse todos os pacotes,

estes estariam criptografados, conservando a transferência entre os servidores de aplicação e o servidor de banco de dados assegurada (Figura 1.).

Outra tecnologia utilizada para interligar um dos servidores foi a VPN, por ser estável e segura com relação à criptografia. As chaves geradas para “criptografar” os dados é muito grande, dificultando a quebra de sigilo do algoritmo utilizado.

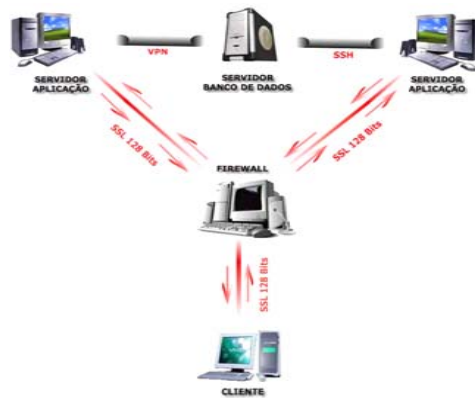


Figura 2. Solução proposta para um e-commerce e e-business seguro

O SSH foi utilizado por prover conexão “tunelada” e segura entre duas máquinas. A idéia foi criptografar os dados transferidos entre um dos servidores de aplicação e o servidor de banco de dados. Os servidores de aplicação disponibilizam uma loja de e-commerce desenvolvida em JSP, onde o cliente pode cadastrar seus dados e comprar mercadorias disponíveis. Os dados submetidos pelo cliente são enviados para o servidor de banco de dados por uma conexão “tunelada” provida pela VPN, que dá todo respaldo necessário com relação à criptografia e segurança dos dados (Figura 3).

Tabela 1. Exemplo de balanceamento de carga entre os servidores de aplicação

Requisição	Número de Requisições	% de Atendimento do Servidor de Aplicação I	% de Atendimento do Servidor de Aplicação II
Inserção	25	13	12
Consulta	25	12	13
Alteração	20	10	10
Exclusão	30	15	15
Total	100	50	50

Para maior segurança da proposta especificada, foi utilizado em conjunto com todas as tecnologias descritas acima o SSL de 128 bits, com o papel de criptografar os dados desde o browser do cliente até o servidor de aplicação. Para reforçar ainda mais a segurança dos servidores, foi usado IP falso. Assim, mesmo sabendo o número IP dos servidores de aplicação e também o de banco de dados, o atacante não conseguiria invadir as máquinas, pois somente o firewall responde às requisições. Quando o administrador necessita buscar algo na Internet, ou até mesmo na rede interna, por

intermédio de um servidor de aplicação, o pedido é feito, mas quem responde pelo servidor de aplicação é o firewall, e o seu é o único número IP que pode ser rastreado.

Com os mecanismos levantados uma determinada empresa pode implementar uma política de segurança levando em conta todos os aspectos citados neste artigo, atingindo altos padrões de segurança de informação com baixo custo.

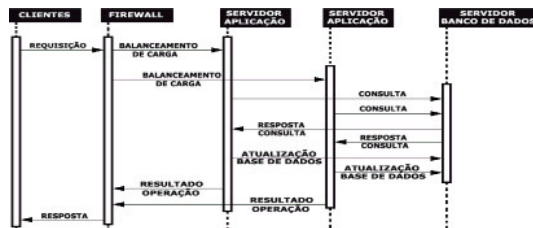


Figura 3. Diagrama de sequência – Interação completa do usuário com a loja virtual

3.2 Resultados

Nesta seção, são apresentados os resultados obtidos a partir da simulação do modelo proposto quando submetido a várias requisições simultâneas, ataques à estrutura implementada, balanceamento de carga e negação de serviço. O estudo feito se concentra sobre a aplicação de métodos que possam assegurar transações no momento em que o cliente envia informações sigilosas para o servidor.

Na simulação, foram feitas várias requisições aos servidores de aplicação que, em conjunto com o sistema de balanceamento de carga, responderam de forma dividida e simplificada como na Tabela 1.

Após as requisições foram instalados “trojans” nos servidores, abrindo portas para invasão. Entretanto não houve efeito devido ao fato do sistema de IP falso ser capaz de proteger o restante da infra-estrutura. A título de simulação, um sniffer foi instalado em uma máquina à parte e conectada à rede dos servidores (interna ao firewall) para que pudesse capturar dados transferidos entre eles. Foram feitas novas requisições após a existência do sniffer, e todos os pacotes transferidos do cliente para os servidores foram “criptografados” conforme a análise do relatório gerado pelo sniffer.

Outra simulação tentou estabelecer conexões SSH, barradas pelas restrições aplicadas ao firewall. Conexões TELNET e FTP, bem como o “ping da morte”, também foram barradas, trazendo maior confiabilidade ao restante dos servidores.

4 Conclusão

Nosso trabalho focou, de forma ampla e abrangente, tecnologias disponíveis no mercado que provêm segurança em sites de e-commerce e e-business, despontando como mais utilizadas (VPN, SSL, FIREWALL, ANTI-VÍRUS), citando também a política de segurança para os meios físicos onde a aplicação será implementada.

Este trabalho descreve uma proposta para acesso a uma loja de e-commerce, permitindo diversos níveis de segurança para com os clientes e para com a infraestrutura envolvida, validada através de simulações como compras na loja implementada, e

também tentativas de ataques ao firewall e aos servidores de aplicação. Com base nos valores para aquisição das tecnologias avaliadas foi desenvolvida uma política de segurança acessível às empresas da região da pesquisa.

Acreditamos que o resultado deste trabalho tem muito a contribuir com as empresas da região, oferecendo-lhes condições de ganho de mercado (usando a Internet) de forma segura e com custo aceitável.

5 Trabalhos Futuros

O próximo passo em busca de integridade nas transações entre cliente/servidor é agregar a idéia de Sistemas Distribuídos ao projeto, efetuando, por intermédio de uma rede, rastreamento em máquinas clientes. O trabalho terá como funcionalidade:

- Manter no servidor uma lista com a cópia de todos os arquivos que compõem uma determinada máquina na rede.
- Efetuar uma varredura completa em busca de todos os arquivos instalados e gerar um valor para controle sempre que a máquina cliente for iniciada.
- Se o valor recém-processado diferir daquele no servidor, verificar o motivo da diferença comparando a lista do servidor com uma lista de arquivos do cliente.
- Se constatados arquivos no cliente além dos presentes na lista do servidor, apagar os excedentes, tendo em vista que os arquivos necessários para o bom funcionamento da máquina sempre serão os mesmos.
- Se constatada, entretanto, a falta de arquivos na máquina cliente, verificar na lista quais arquivos, e imediatamente iniciar as cópias necessárias.
- No caso de listas iguais, iniciar varredura em busca de vírus e vulnerabilidades que permitam interceptação de dados ainda não “criptografados”.
- Se encontrada alguma vulnerabilidade, vírus ou trojan, gerar um arquivo de notificação (log) para o administrador da rede. A máquina fica restrita ao uso até o administrador desbloquear o sistema efetuando as devidas correções.
- Feitas as correções requerer o reinício da máquina, provocando nova verificação.

6 Referências Bibliográficas

- Cooper, S. Construindo um firewall para Internet. Rio de Janeiro: Campus, 2001.
- Gomes O. Segurança Total: protegendo-se contra hackers. São Paulo: Makron Books, 1995.
- Scott, C. Virtual Private Networks. Beijing: O’illy, 1999.
- Siyam, Katarine Internet Firewalls and Network Security. Indianapolis: New Riders Publishing, 1995.
- Smith, R. E. Internet Cryptography. Addison Wesley Longman, Massachusetts: 1997.
- Terada, R. Segurança de Dados: criptografia em redes. São Paulo: Edgard Blucher, 2000.
- Scambray, J. Hackers Expostos: 2ª Edição. Makron Books, 2001.

Análise de Aplicações de Voz em Redes IP

Adriana Patrícia de Oliveira¹, M.A.R. Dantas²

¹Departamento de Engenharia Elétrica – Laboratório de Engenharia de Redes
(UnB/ENE/LabRedes) – Universidade de Brasília
Av.L3 Norte – FT – ENE – LabRedes – Sala B1-01 – Asa Norte – CEP:70910-900 –
Brasília-DF – Brasil

²Departamento de Informática e Estatística (INE) – Universidade Federal de Santa
Catarina (UFSC)
Caixa Postal 476 – Trindade – Florianópolis – SC – 88040-900 – Brasil
adrianapatricia@brt14.com, mario@inf.ufsc.br

Abstract. *In the last years, the effective use of the technology of voice on IP networks comes if becoming a reality. This technological approach is one of the great targets of investments of the provider (and users) of the facility of telecommunications. Due to importance of this configuration in countless companies, we make in this article one analyzes of voice applications on a net with protocol IP. Our results indicate the importance of one politics of quality of service for the success of the execution of an application of voice in a network IP. The use of protocols that supply a quality of service has as objective to guarantee the band necessary for the transmission of the voice packages and to minimize the delays suffered for the packages in the net, being become them most constant possible and providing the priority required for the voice packages.*

Resumo. *Nos últimos anos, o uso efetivo da tecnologia de voz sobre as redes IP vem se tornando uma realidade. Esta abordagem tecnológica é um dos grandes alvos de investimentos dos prestadores (e usuários) das facilidades de telecomunicações. Devido à importância desta configuração em inúmeras empresas, fazemos neste artigo uma análise de aplicações de voz sobre uma rede com o protocolo IP. Nossos resultados indicam a importância de uma política de qualidade de serviço para o sucesso da execução de uma aplicação de voz em uma rede IP. O uso de protocolos que forneçam uma qualidade de serviço tem como objetivo garantir a banda necessária para a transmissão dos pacotes de voz e minimizar os atrasos sofridos pelos pacotes na rede, tornando-os o mais constante possível e provendo a prioridade requerida para os pacotes de voz.*

1. Introdução

Historicamente, as aplicações de voz e a transmissão de dados utilizavam sistemas de transmissão distintos e, também, empregavam diferentes tecnologias. Nos últimos anos tem sido verificado uma tendência em diversas empresas, a interligação de suas redes de voz com redes IP. Um dos fatores que levam a esta unificação é o grande volume de dados que são transmitidos nas organizações, quando comparamos com as aplicações de voz. Todavia, é importante lembrar que a transmissão da voz solicita requisitos especiais a uma rede que não são verificados de uma forma nativa nas redes com o protocolo IP.

As tecnologias de transmissão de voz em redes que utilizam o protocolo IP vêm sendo aperfeiçoadas através do desenvolvimento de novos protocolos. Estes protocolos têm características diversas tais como a solicitação de reserva (um exemplo é o protocolo RSVP [3]), tipo de prioridades, metodologias de classificação e o enfileiramento de pacotes. Estas novas características adicionadas a uma rede IP convencional permitem a adequação da rede para uma perfeita transmissão de voz.

Neste artigo, apresentamos resultados empíricos obtidos em uma série de experimentos que efetuamos para o auxílio na identificação das variáveis que mais influenciam na qualidade do funcionamento de uma rede IP suportando uma transmissão de voz. Desta forma, observamos o comportamento da rede IP e traçando comparações de forma a averiguar melhorias, ou não, nos parâmetros utilizados. É importante ressaltar que as aplicações de voz exigem uma qualidade mínima em termos de parâmetros temporais (exemplo o atraso e o jitter) e capacidade efetiva de transmissão (largura de banda).

Para uma melhor compreensão do nosso trabalho de pesquisa, este artigo foi organizado em quatro seções. Na segunda seção apresentamos o ambiente onde realizamos nossos testes para uma melhor análise de aplicações de voz sobre IP. Nossos estudos de casos são apresentados na terceira seção. Finalizamos este artigo na quarta seção, onde apresentamos nossas conclusões.

2 – Ambiente

Nosso ambiente de teste é apresentado na figura 1. A implantação do serviço de transmissão de voz ocorreu em um ambiente real de uma empresa que disponibilizou o ambiente e equipamentos utilizados neste projeto. Nesta implementação, executamos os testes para avaliação do tráfego de voz sobre redes IP. No ambiente implementado existem três cenários experimentais onde apresentamos os estudos de casos analisando o funcionamento da transmissão de voz em redes IP e os problemas encontrados em cada um deles.

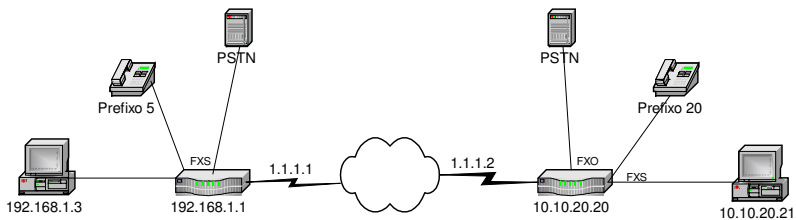


Figura 1 – Ligação dos componentes de hardware no ambiente de projeto.

Imaginamos três estudos de casos que pudessem ser representativos para nossa proposta de pesquisa. Desta forma, optamos por cenários onde existissem (a) somente tráfego de voz, (b) o tráfego de voz com rajadas de dados aleatórias e finalmente (c) o tráfego de voz com tráfego de dados contínuo e uniforme.

3 – Resultados Experimentais

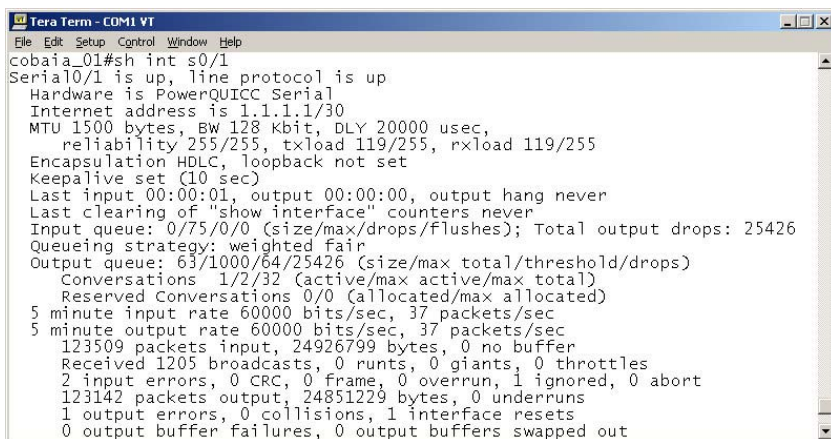
Nesta seção vamos apresentar nossos três estudos de casos implementados no ambiente experimental da empresa.

3.1 - Tráfego somente de Voz

Neste primeiro estudo de caso, efetuamos chamadas de voz entre os dois pontos, sem tráfego de dados, verificamos a qualidade de voz numa situação em que há apenas tráfego de voz em redes que utilizam o protocolo IP. Neste experimento os pudemos fazer ligações de uma extremidade para extremidade, conectados via rede IP.

A taxa de transmissão utilizada no teste foi de 64 kbps. No transporte de voz é solicitada uma disponibilidade constante de largura de banda [6], caso não exista a largura de banda necessária, todos os outros fatores de qualidade da voz estarão comprometidos. Neste primeiro experimento não tivemos problema já que estávamos transmitindo apenas a voz na rede de dados e utilizamos toda a banda requerida para a operação.

Depois de estabelecida uma nova conexão e adicionando uma música na origem, percebemos um grande atraso na transmissão. Estes atrasos foram da ordem de 1 à 1,5s e são considerados grandes para aplicações tipo VoIP. Em adição, observamos que a voz se apresentou de uma forma falha (*picotada*). Através da figura 2, pudemos observar que a transmissão de nosso teste saturou a banda chegando a 60000 bits/segundo.



```

Tera Term - COM1 VT
File Edit Setup Control Window Help
cobaia_01#sh int s0/1
Serial0/1 is up, line protocol is up
Hardware is PowerQUICC Serial
Internet address is 1.1.1.1/30
MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
  reliability 255/255, txload 119/255, rxload 119/255
Encapsulation HDLC, loopback not set
Keepalive set (10 sec)
Last input 00:00:01, output 00:00:00, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 25426
Queueing strategy: weighted fair
Output queue: 63/1000/64/25426 (size/max total/threshold/drops)
Conversations 1/2/32 (active/max active/max total)
Reserved Conversations 0/0 (allocated/max allocated)
5 minute input rate 60000 bits/sec, 37 packets/sec
5 minute output rate 60000 bits/sec, 37 packets/sec
123509 packets input, 24926799 bytes, 0 no buffer
Received 1205 broadcasts, 0 runts, 0 giants, 0 throttles
2 input errors, 0 CRC, 0 frame, 0 overrun, 1 ignored, 0 abort
123142 packets output, 24851229 bytes, 0 underruns
1 output errors, 0 collisions, 1 interface resets
0 output buffer failures, 0 output buffers swapped out

```

Figura 2 – Transmissão com saturação de banda

Uma observação natural do experimento é que a maior utilização disponível do uso da largura de banda é o primeiro recurso a ser explorado, já que a transmissão de dados em tempo real depende de uma largura de banda maior. Ao aumentar a banda para 128Kbps, a qualidade da voz melhorou significativamente, mas o consumo da banda utilizada foi muito grande, a banda utilizada passou a ser 81000 bits/segundo.

Um dos importantes aspectos que devem ser considerados no projeto de soluções que envolvam a transmissão de voz em redes IP é a escolha do algoritmo de compactação de áudio. Esta escolha influi diretamente sobre uma serie de fatores, principalmente sobre a largura de banda necessária para as conexões de voz e os atrasos referentes à mesma e a qualidade da voz digitalizada observada. Com o Codec G729[2] a banda utilizada em nosso experimento foi reduzida significativamente a uma taxa de 25000bps. Este codec é recomendado quando uma largura de banda de áudio mais baixa é necessária através da Internet.

3.2 – Tráfego de Voz com Rajadas de Dados Aleatórias

Neste segundo estudo de caso, o ambiente utilizado foi o mesmo do caso anterior, considerando a figura 1, porém o tráfego de voz foi transmitido com rajadas de dados aleatórias, usamos o codec G.729 e uma banda de 64K para fazer os todos os testes de transmissão de voz junto com a rede de dados.

Neste teste configuramos a compressão de cabeçalho RTP [1] que é usado para economizar largura de banda em redes IP. Contatamos que enquanto o tráfego era apenas de voz, a conversação era normal (nítida) e de tempo real. Entretanto, no momento que surgia uma rajada de dados já não tínhamos a mesma qualidade, a voz tinha um retardo e ficava *picotando*, como o tráfego era de rajada, após terminar a transmissão de dados, a voz era normalizada.

Nesta configuração, a fila que estávamos usando era a fila do tipo FIFO[5], e esta não possui nenhum tipo de classificação de pacote. Em seguida, habilitamos a fila do tipo WFQ (Weighted Fair Queuing) [4] que funciona bem se existir um pequeno número de fluxos. Com as rajadas de tráfego de dados bem pequenas, a voz era nítida, mas à medida que aumentávamos a o tráfego de dados a qualidade caía gradativamente. Para resolvermos este problema foi aplicado o enfileiramento LLQ (Low Latency Queuing) [4] que possui sistema de política para enfileiramento de diferentes tipos de tráfego. Com isso o tráfego de voz teve sua prioridade estabelecida sem influência do tráfego de dados, assim reservamos uma banda a de 25K somente para voz. Usando a compressão RTP, com os mesmos dados do teste anterior a taxa de transmissão de bits por segundo teve uma queda de 25000 para 10000bps/seg.

Ao fazermos duas ligações simultâneas, a banda ocupada passa ser o dobro. Este recurso é benéfico se estiver rodando VoIP em links lentos. Com as duas conexões de voz estabelecida, foi possível acessar o micro da rede remota, acessando uma aplicação Web e uma conexão com o serviço de Terminal Server da Microsoft ao mesmo tempo em que falávamos e não foi percebida nenhuma interferência durante a conversação, constatando-se que realmente a banda que fora reservada para voz, não era utilizada por dados.

3.3 – Tráfego de voz com Tráfego de Dados Contínuo e Uniforme

Para o nosso terceiro experimento foi incluído mais dois roteadores, devido à necessidade de usar o protocolo RSVP[3] e para tanto não tinha sentido marcar o caminho em apenas dois pontos, conforme se pode observar pela figura 3.

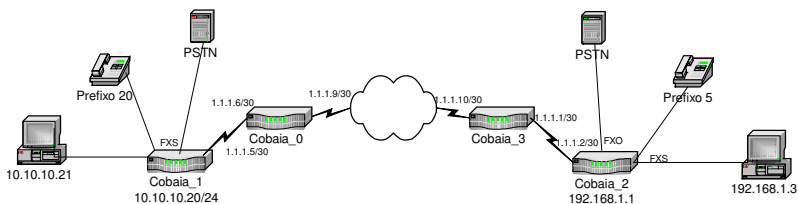


Figura 3 – Estudo de Caso 3

Em um primeiro momento as configurações utilizadas nos *gateways* foram às mesmas do segundo experimento, apesar das configurações terem sido as melhores para o estudo anterior. Estas não foram suficientes para este caso, devido aos grandes atrasos que foram gerados, pois os roteadores inseridos no meio não tinham nenhuma configuração de prioridade e reserva. Com o RSVP[3] habilitado em todos roteadores da

rede, foi possível marcar todo o caminho. Para isto todos o roteadores na rede através da qual uma chamada é roteada precisa suportar RSVP.

Ao configurarmos a priorização de pacotes, reserva de recursos e a largura de banda, notamos que a qualidade de voz melhor significativamente e conseguimos com sucesso que os pacotes fossem entregues em seus destinos sem atraso.

4 – Conclusões

As redes IP já representam uma plataforma importante para as aplicações de voz. Como a qualidade de voz é crítica em uma rede VoIP, precisamos considerar a qualidade de voz em ambas fases de projeto e implementação do desenvolvimento do VoIP.

Nos experimentos realizados neste artigo, nosso objetivo foi testar algumas das diversas formas de se transmitir voz em redes IP e compreender a relação entre alguns mecanismos necessários para a perfeita transmissão de voz. Desta forma, em nossos experimentos utilizamos algumas perturbações que são usualmente identificadas e são fatores que influenciam o desempenho da rede. Configuramos a rede IP para suportar tráfego de voz em tempo real e para obter uma qualidade de voz satisfatória. Podemos afirmar que nossos estudos de caso atingiram o sucesso esperado, visto que foi possível se identificar os problemas nas transmissões e suas possíveis correções.

Em adição, nossos testes indicaram que a (1) qualidade do áudio e os atrasos são diretamente influenciados pelos *buffers* de *jitter*, (2) a escolha da técnica de codificação do áudio é uma das decisões mais importantes a se considerar, e (3) a relevância da definição da prioridade e a escolha de uma fila adequada para o ambiente.

Bibliografia

- [1] Held, Gilbert. **Voice Over Data Networks**. McGraw-Hill, 1998
- [2] Hersent, Oliver; Guide, David; Petit, Jean Pierre. **Telefonia ip – comunicação multimídia baseada em pacotes**. Addison Wesley, 2002.
- [3] RNP – **O Protocolo RSVP e o Desempenho de Aplicações Multimídia**. www.rnp.br/news/0005/rsvp.shtml, 2002
- [4] Promon Telecom – **Voz sobre rede de dados**, Outubro de 2002.
- [5] Chowdhury, Dhiman D J. **Projetos avançados de redes IP**. Campos, 2002.
- [6] Soares, Lilian/Freire, Victor. **Redes Convergentes**. AltaBooks, 2002

Desenvolvimento de Aplicações 3G – Perspectivas –

Lucas Mello Schnorr¹, Juergen Rochol¹

¹Universidade Federal do Rio Grande do Sul - Instituto de Informática
Av. Bento Gonçalves, 9500 - Campus do Vale - Bloco IV
Bairro Agronomia - Porto Alegre - RS -Brasil
CEP 91501-970 Caixa Postal: 15064

lmschnorr@inf.ufrgs.br

Abstract. *3G is the third generation of celular systems. Its characteristic allows a greater interactivity with the user, new applications, a large bandwidth and the possibility of a new paradigm of interaction human/computer. This text shows some examples of 3G applications projected, together with the characteristic that allow these applications, a new interaction paradigm and where already exists third generation celular systems and its services in the world.*

Resumo. *3G é a terceira geração de sistemas celulares. Suas características permitem uma maior interatividade com o usuário, novas aplicações, uma grande largura de banda e a possibilidade da criação de um novo paradigma de interação humano/computador. Este texto mostra alguns exemplos de aplicações 3G, juntamente com as características que permitem essas aplicações, uma discussão sobre um novo paradigma de interação e onde já existem sistemas celulares de terceira geração e seus serviços no mundo.*

1. Introdução

O primeiro sistema celular a ser utilizado em larga escala foi lançado no final dos anos 70 e início dos anos 80. Este sistema tinha como foco principal a comunicação por voz e realizava um número pequeno de ligações ao mesmo tempo. Esse sistema era conhecido como a primeira geração dos sistemas celulares.

A segunda geração de sistemas celulares surgiu no início dos anos 80. Ela teve como principal mudança a codificação da voz em dados digitais e a introdução de pequenas mensagens textuais (*Short Message Service*). O principal padrão utilizado na segunda geração foi o GSM (*Global System for Mobile Communications*).

A evolução a partir da segunda geração em direção a terceira tem como etapa intermediária a geração conhecida como 2.5G. Esta geração permite uma comunicação através de pacotes e permite taxas de transferência de dados maiores que a segunda geração. O serviço principal que evoluiu do GSM foi o GPRS (*General Packet Radio Service*) que permite que cada celular fique sempre conectado à rede. As principais aplicações que surgiram nessa geração intermediária são as mensagens EMS (*Enhanced Messaging Service*), que permite enviar mensagens de texto com imagens anexadas.

Ainda na evolução para terceira geração (3G), um novo serviço GSM chamado EDGE (*Enhanced Data Rates for Global Evolution*) está sendo considerado como a geração 2.7G. A principal característica deste serviço é a taxa de transferência máxima de 384kbts/s, permitindo um rol de aplicações mais sofisticadas que a geração 2.5G [Sandhu, 2003].

A evolução dos sistemas celulares sempre foi marcada dentro de cada geração com uma nova característica e, a nível de usuário, com uma nova aplicação. A terceira geração dos sistemas celulares ainda está à espera de uma aplicação marcante [Garber, 2002]. Este texto tem como objetivo mostrar algumas dessas aplicações para a terceira geração de sistemas celulares e uma discussão sobre o novo paradigma de interação usuário/equipamento na área de celulares e aplicações 3G.

Este texto está organizado da seguinte forma: a seção 2 discute assuntos relacionados a mudança de paradigma no desenvolvimento de aplicações 3G; a seção 3 mostra algumas aplicações que propõe ser parte do rol de futuros serviços oferecidos pelos sistemas 3G; a seção 4 mostra onde já existem sistemas 3G no mundo e quais serviços as operadoras desses sistemas oferecem ao usuário final; a seção 5 apresenta as principais dificuldades da evolução dos sistemas celulares para a tecnologia 3G. Por fim, é apresentada uma conclusão.

2. Descoberta de novas aplicações

As características da terceira geração dos sistemas celulares, como por exemplo a grande largura de banda, oferecem suporte a altos níveis de interatividade. Este novo suporte nos celulares é o que torna desafiante o desenvolvimento dessas novas aplicações.

Segundo Heiko Sacher e Gareth Loudon [Sacher and Loudon, 2002], as aplicações interativas hoje se classificam em dois principais paradigmas de interação: o paradigma dos celulares e dos computadores comuns. Quando os programadores desenvolvem aplicações para esses sistemas, eles embarcam nessas aplicações, regras de cada paradigma. Por exemplo, quando um usuário está trabalhando com um computador; o usuário sabe que a forma que ele interage com o computador é diferente do celular.

A natureza da tecnologia da terceira geração desafia essa classificação de paradigmas. Uma perspectiva puramente técnica de 3G pode ser vista como uma convergência do paradigma de interação com celular e com computadores [Sacher and Loudon, 2002]. Assim, devem existir aplicações que explorem este novo paradigma de interação humano/computador.

No texto de Sacher e Loudon é proposta uma nova forma de se descobrir aplicações para essa nova combinação de paradigmas de interação: aplicações baseadas na cultura do consumidor. Esta técnica é apresentada ao longo desta seção.

Identificação de aplicações baseado na cultura do usuário

Quando se tem um conjunto já definido de padrões e técnicas de desenvolvimento, pode-se construir melhorias incrementais em uma aplicação. Quando estamos tratando de criar um conjunto de aplicações para uma nova tecnologia, como é 3G, deve-se procurar outros meios de iniciar o desenvolvimento. Uma alternativa de como deve ser construídas as

aplicações é aquela que se baseia na cultura interativa do usuário para identificar necessidades e possíveis soluções.

As regras de interação humanas são definidas em valores compartilhados, crenças e regras de uma cultura. A definição é a chave para se descobrir regras apropriadas para aplicações e produtos novos [Sacher and Loudon, 2002]. Exemplos que exploram esta forma de se identificar novas aplicações ou produtos foram feitos e descritos no trabalho de Suchman [Suchman, 1995].

3. Aplicações 3G

3G irá permitir maiores taxas de transferências variando de 144kbit/s até 2Mbit/s dependendo da taxa de mobilidade de usuário. As principais características da terceira geração são um aumento de suporte a serviços multimedia e capacidade de permitir taxas de transferências fixas e variáveis, largura de banda sob demanda e taxas de transferência assimétricas nos links de comunicação [Sandhu, 2003].

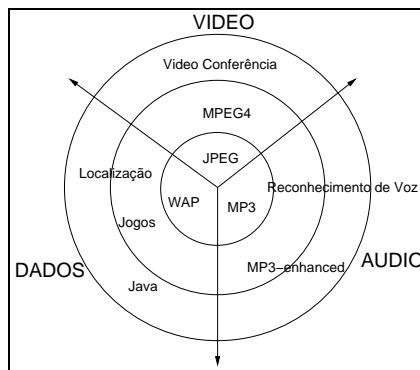


Figura 1: Evolução das aplicações 3G nos quesitos de dados, áudio e vídeo [Philips, 2002]

Dada as características de 3G, pesquisadores e desenvolvedores devem descobrir o que os usuários querem fazer a qualquer hora e em qualquer lugar, e determinar como estas tarefas podem ser disponíveis e fáceis de utilizar [Tarasewich and Warkentin, 2000].

A figura 3 mostra a evolução das aplicações 3G em três áreas: dados, áudio e vídeo. Atualmente, com os serviços disponibilizados pela tecnologia 2.5G tem-se serviços de envio de imagens e músicas e alguns recursos de reconhecimento de voz. Com a evolução para a tecnologia 3G, espera-se ter aplicações poderosas relacionadas a jogos, vídeo-conferência, *m-commerce* e serviços de localização. As próximas sub-seções mostram algumas aplicações que podem vir a ser utilizadas em serviços 3G.

Propaganda móvel

De acordo com Varshney e Vetter [Varshney and Vetter, 2002], propaganda móvel é uma classe de aplicação de comércio móvel (*m-commerce*). Esta forma de propaganda pode

ser usada utilizando informação coletada por serviços móveis e informação sobre a localização dos usuários móveis. Dessa forma pode-se fazer propaganda específica da região onde o usuário esta e como ele pode à loja que está fazendo a propaganda.

Com as características de 3G, as mensagens podem conter vídeos, imagens e sons de alta qualidade, permitindo uma maior eficácia na propaganda de um produto ou loja. As mensagens podem ser personalizadas com informações coletadas na forma que o usuário realiza as compras e os lugares.

Devido a grande capacidade de divulgação em massa que o meio sem fio permite, questões relacionadas a privacidade de informação e a autorização por parte do usuário receber mensagens devem ser levadas em conta na implementação de um sistema de propaganda através dos sistemas 3G.

Gerenciamento de estoque móvel

Gerenciamento de estoque móvel é uma classe de aplicações que envolve localização de produtos e serviços. O rastreamento dos produtos pode ajudar as empresas a determinar o tempo de entrega ao consumidor, melhorando este serviço e obtendo um limiar competitivo com os outros negócios de um setor.

Um exemplo desta classe de aplicação é o transporte da matéria-prima de um produto para o local de produção. Nessa aplicação, os componentes iniciais, enquanto se movem ao local de produção, podem ser controlados em termos de velocidade de deslocamento e quantidade, de forma que não haja estoque, reduzindo assim os custos de produção. Situações como a adaptação da linha de produção a chegada de novas matérias-prima podem ser resolvidas através do controle sobre o transporte até a fábrica.

Aplicações como essa devem associar as características dos sistemas 3G com sistemas de posicionamento global (GPS). A utilização de gráficos de alta qualidade, verificação de localização e sons podem ajudar ao usuário da aplicação a interagir com o sistema. Dessa forma, a aplicação pode aliar a sua utilidade com uma boa exploração dos recursos 3G.

Leilão móvel

A Internet hoje está repleta de sites de leilão, nos quais qualquer pessoa é capaz de vender um produto leiloando-o ou utilizando um valor fixo. Aplicações desenvolvidas para 3G podem interagir com estes sites de forma que possam, enquanto se movem e em qualquer lugar, leiloar produtos ou comprar produtos.

Um exemplo claro disso são os leilões nos quais o produto a ser leiloado está geograficamente distante de uma cidade e das pessoas interessadas. O chefe do leilão neste caso, munido de uma aplicação 3G para tal, pode leiloar o produto a pessoas que estão longe.

Características como autenticação de identidade, alta disponibilidade do serviço 3G e garantias de entrega devem ser salientadas nas aplicações desenvolvidas para leilão móvel junto com 3G.

Jogos

Jogos de computadores são um tipo de aplicação altamente utilizada pela indústria do entretenimento. Os sistemas celulares de hoje permitem a utilização de jogos mas de

forma mono-usuário. O grande avanço de 3G, neste caso, é permitir a utilização de grande largura de banda para se construir jogos multi-usuário com alta interatividade.

Um problema no desenvolvimento dessas aplicações é que os aparelhos 3G devem ter uma alta resolução na qualidade das imagens e um grande poder de processamento computacional, para prover ao jogo uma grande variedade de recursos.

Devido a mobilidade dos sistemas celulares e a localização geográfica através de GPS, seria possível criar jogos que têm uma relação mais próxima a da realidade. Nesses jogos, os jogadores poderiam andar por um ambiente geográfico real e interagir com outros usuários distantes através do sistema celular 3G com vídeocone, por exemplo. Jogos assim poderia ser utilizados como simulação da realidade em resolver casos policiais de investigação.

Localização

Através da utilização de técnicas para se localizar um aparelho celular baseado no seu sinal (GPS), pode-se encontrar pessoas geograficamente e marcar encontros via telefone celular. Esse recurso 3G pode ser explorado também para se encontrar a melhor rota para se chegar a algum lugar geográfico.

O sistema pode ser utilizado também através do desenvolvimento de aplicações que encontrem rotas que estão congestionadas e sugerir então rotas alternativas para se chegar a algum lugar o mais rápido possível.

Pode-se desenvolver aplicações que procurem produtos ou serviços que estejam o mais próximo possível do usuário que faz a requisição. Isso torna o serviço ou o encontro do produto mais rápido e barato, visto que pode-se fazer uma consulta prévia do preço antes da compra.

Vídeocone

Devido a grande largura de banda e aparelhos capazes de obter imagens contínuas em forma de vídeo, é possível através visualizar a pessoa com quem se conversa ao telefone. Essa classe de aplicação necessita uma grande largura de banda diretamente relacionada com a qualidade do vídeo que está se transmitindo e recebendo.

4. Realidade

A primeira rede celular 3G foi lançada no Japão pela companhia NTT DoCoMo. Ela tem como abrangência um raio de 30 quilômetros nos arredores de Tóquio. Este sistema celular utiliza uma tecnologia de rede chamada FOMA - *Freedom of Mobile Multimedia Access*. Outra rede 3G foi entregue em novembro de 2002 na Hungria sob o nome de Vodafone (VPRAM Rt).

Os aparelhos celulares também estão se adaptando a nova tecnologia de sistemas celulares. Modelos de aparelhos da Siemens tem captura de imagens gráficas e utilizam codificadores como MPEG4 para vídeos enviados como mensagens Multimedia. Nokia lançou um modelo de celular que suporta redes W-CDMA e GSM, podendo ser utilizado em ambas as redes.

5. Ameaças à tecnologia 3G

Uma variedade de fatores decidirão se 3G terá sucesso como a próxima geração dos sistemas móveis. Alguns desses fatores são os custos altos, o processamento e tráfego de dados onde normalmente se trafega voz, o conteúdo disponibilizado pelos serviços e a inexistência de uma aplicação marcante de forma prática até agora [Garber, 2002].

Os altos custos se referem principalmente ao pagamento dos direitos de uso do espectro da frequência padronizado para a tecnologia 3G. As licenças, em alguns países como a Coreia do Sul, atingem preços exorbitantes de bilhões de dólares. O processamento de grandes quantidades de dados se refere a uma característica da padronização 3G, que é prover alta largura de banda a todos os usuários. As aplicações entram então com uma grande importância por ter a missão de fornecer aos usuários informações atrativas combinadas com uma aplicação marcante.

6. Conclusão

3G é a nova tecnologia celular que permitirá o desenvolvimento de aplicações que possam explorar uma grande largura de banda e por consequência uma maior interatividade. Os investimentos das empresas na compra dos direitos de utilização das frequências 3G têm sido grandes.

Mesmo assim, é importante salientar que 3G necessita de uma aplicação marcante, como foi as mensagens de texto na tecnologia 2G e o envio de imagens na tecnologia 2.5G. Discute-se muito qual será esta aplicação, sendo que algumas delas foram apresentadas neste texto. As características principais dessas aplicações são interatividade, alta qualidade e alta disponibilidade, juntamente com autenticação de usuários.

Referências

- Garber, L. (2002). Will 3g really be the next big wireless technology? *Computer*, 35(1):26–32.
- Philips (2002). 3g vision. <http://semiconductors.philips.com/markets/communications/3g/vision>.
- Sacher, H. and Loudon, G. (2002). Uncovering the new wireless interaction paradigm. *Interactions of the ACM*, 9(1):pp 17–23.
- Sandhu, K. (2003). 3g, 3w, what? where? when? *3rd Annual Multimedia Systems*. Southampton University, UK.
- Suchman (1995). Making work visible. *Communications of the ACM*, 38(9):p 56.
- Tarasewich, P. and Warkentin, M. (2000). Issues in wireless e-commerce. *ACM SIGecom Exchanges*, 1(1):pp 21–25.
- Varshney, U. and Vetter, R. (2002). Mobile commerce: framework, applications and networking support. *Mobile Networks and Applications*, 7(3):185–198.

Sistema de Hardware e Software para Videomonitoração Através de Telefones Celulares

**Jorge Guedes, Rafael Rehm, Fernando Thiesen, Francisco Souza, Luciano
Azevedo**

Pontifícia Universidade Católica do Rio Grande do Sul
Laboratório Metropoa – Redes Metropolitanas de Alta Velocidade de Porto Alegre
Departamento de Engenharia Elétrica – Av. Ipiranga 6681. Prédio 30, Bloco 5, Sala 151
Porto Alegre – RS – 90619-900 – Brasil – Fone: 51 3320-3500 ramal: 4056 subramal: 223
<http://camera.metropoa.tche.br>

{guedes, rehm, thiesen, franciscoss, azevedo}@pucls.metropoa.tche.br

Abstract

In this article is presented research realized to develop an integrated monitoring system through camera directly connected to the server, where is possible to establish the access to the images through cellphones. This server developed in our lab consists in integrating an extremely low-cost kernel linux server software, which integrates the functionality of a web server with the image capture of a safety camera. Thus, it consists a system of multi videomonitoration via cellphones GSM/GPRS and web, which makes possible the visualization of images generated by cameras installed in home and building environment.

Resumo.

Neste artigo apresentamos os estudos realizados para o desenvolvimento de um sistema integrado de videomonitoração através de câmeras ligadas diretamente ao servidor, onde o acesso às imagens poderá ser feito por telefones celulares. Este servidor desenvolvido em nosso laboratório consiste na integração de software dotado de kernel Linux, de baixíssimo custo, o qual integra a funcionalidade de servidor WEB com a captura de imagens de câmeras de segurança. Assim constitui-se um sistema de múltipla videomonitoração via redes celulares GSM/GPRS e web, que possibilita a visualização de imagens de câmeras instaladas em ambientes residenciais e prediais.

Introdução

Este artigo descreve um sistema em hardware e software, que tem como gerenciador a plataforma Linux [LINUX], o qual está em desenvolvimento no Laboratório Metropoa da PUCLRS. O projeto tem como objetivo criar uma ferramenta, capaz de possibilitar a monitoração de diversos ambientes através de um telefone celular WAP, que suporte a tecnologia GSM/GPRS.

O WAP não se limita à páginas estáticas, ele oferece a possibilidade de integrar banco de dados, conteúdo dinâmico e videomonitoramento. Foi projetado para trabalhar com vários tipos de redes sem fio, tais como TDMA, CDMA, GSM/GPRS, entre outras.[DORNAN] As vantagens de utilizar WAP para a videomonitoração são inúmeras. Possui praticamente todas as vantagens da Internet, mas com um importante diferencial: a mobilidade. É possível monitorar a sua casa ou escritório utilizando os novos telefones de última geração, não importando o lugar, desde que se tenha cobertura da rede celular. Desta forma, as informações de imagens de câmeras poderão estar disponíveis na tela de telefones celulares.

2. O acesso através da tecnologia WAP.

O WAP é a sigla para *Wireless Application Protocol* (Protocolo de Aplicações Sem Fio) [SOUZA] [WAP], um padrão criado para especificar a forma como os dispositivos sem fio acessam a internet. Se um telefone ou outro dispositivo de comunicação é tido como WAP, significa que este possui um software conhecido como microbrowser, o qual tem a capacidade de interpretar tudo que é especificado como sendo WML (*Wireless Markup Language*). As possibilidades desses navegadores estarão em relação direta com as capacidades do dispositivo. Cada navegador é distinto e pode interpretar o WML de forma distinta.

Já o WML é uma linguagem de programação baseada no XML (*Extended Markup Language*). A especificação oficial do WML foi desenvolvida e é mantida pelo WAP Fórum [NOKIA], um consórcio industrial fundado pela Nokia, Phone.com, Motorola e Ericsson, que atualmente possui mais de 450 membros representando fabricantes, operadores e empresas provedoras de conteúdo de todas as partes do mundo. Concebido para telas de pequena dimensão e navegação sem teclado, a finalidade desta nova tecnologia é oferecer serviços e conteúdos de Internet através de conexões sem fio.

Existem diversos tipos de tecnologia WAP, a primeira versão de publicação do Wap 1.0 foi em meados de 1997 e desde então ela tem evoluído muito chegando até a sua última versão, a 2.0, suportando figuras em jpeg e a linguagem Java conhecida como J2Me. Para o sistema especificado, deveremos utilizar celulares WAP com a tecnologia 2.0 para acessar o videomonitoramento.

3. Definição do Sistema de Hardware e Software

O projeto foi desenvolvido em arquitetura modular (arquitetura de tarefas independentes). Para iniciar a definição do sistema proposto é necessário citar as câmeras de segurança, pois desempenham um papel fundamental no sistema. É necessário a utilização de câmeras analógicas de alta definição, pois permitem o controle da imagem sob quaisquer condições de luminosidade, isto é possível por possuírem um conjunto de lentes chamada auto-íris que adaptam-se automaticamente ao nível de luz existente no ambiente. Com estas lentes, e considerando que as imagens capturadas são coloridas, pode-se conseguir imagens de alta resolução na tela do celular.

Para as aquisições de imagens é utilizada uma placa de captura, controlada pelo módulo de captura de vídeo através de uma interface padrão PCI. Este módulo inclui os softwares desenvolvidos para controlar os dispositivos necessários para obtenção de

imagens a serem transmitidas via *Web*. Além de capturar e controlar o envio das imagens, o sistema também é responsável por adicionar nas imagens a data e hora do local onde a captura é efetuada.

O módulo *webserver* é responsável por permitir a integração da aquisição de imagens e o sistema servidor de páginas [APACHE] na mesma plataforma, disponibilizando o acesso via WAP. Com este aspecto modular do sistema, podemos desenvolver outros serviços, fazendo cada um funcionar de forma autônoma.

Outro aspecto importante para a funcionalidade do sistema é tornar a transmissão de dados rápida, obtivemos tal resultado comprimindo as imagens para o formato jpeg. O formato jpeg é um padrão de compreensão com perda, que não contém toda a informação original mas apenas uma aproximação adequada. O resultado final desse processo é bastante satisfatório.

A arquitetura modular do sistema é escalável, foi projetada de tal forma que novas funcionalidades possam ser continuamente agregadas, bastando para tanto reorientar o projeto, no sentido de manter a estrutura baseada nos módulos implementados. Além da modularidade implícita, foi implementado uma arquitetura confiável dentro dos padrões de disponibilidade elevada. No que se refere à confiabilidade considera-se o sistema operacional LINUX, e para disponibilidade considera-se a redundância na comunicação por meios alternativos, como a utilização de outras tecnologias de acesso complementares à telefonia móvel celular, por exemplo um web browser instalado em um PC.

A telefonia celular pode ser utilizada como meio de acesso e monitoração em áreas sem disponibilidade de redes locais do tipo LAN, ou ainda em situações emergenciais onde não se tenha disponível um computador. Os canais de dados GPRS da telefonia celular podem ser utilizados para recebimento de sinais de vídeo simultâneo ao envio de comandos e controles, através de um portal responsável por acessar todas as funções do sistema, na área de cobertura da operadora celular. Desta forma, agrega-se ao protótipo velocidade na comunicação, característica de sistemas de alta confiabilidade.

3.1. Descrição da Arquitetura de Videomonitoração

A figura a seguir ilustra como é feito o acesso as imagens geradas pelo servidor.

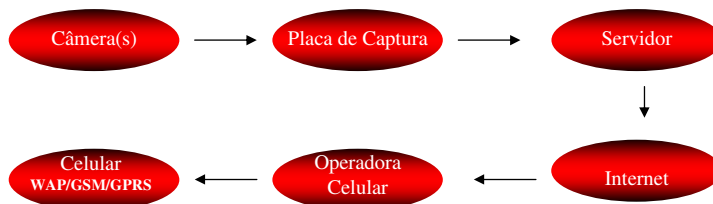


Figura 1. Arquitetura de Videomonitoração

4. A Técnica de Videomonitoramento

A técnica em implementação utiliza as conexões com a Internet, embora tenha muitas limitações. Explica-se da seguinte forma: ao se estabelecer a ligação através de um dispositivo WAP, abre-se uma conexão sem fio através do canal CSD para celulares com tecnologia TDMA, esta conexão é do tipo PPP (*Point-to-Point Protocol*), até um provedor de serviço. Depois de comprovar a identificação e senha do usuário, o dispositivo WAP estará apto a acessar endereços na Internet através de seu microbrowser.

Para acesso através de celulares com tecnologia GSM/GPRS [WIRTH] a diferença está na maneira de se realizar a conexão, onde teremos o tráfego de dados por pacotes e não por conexão pontual. O usuário será tarifado através da quantidade de dados que “baixar” no terminal GSM, além disso, será dotado ao dispositivo WAP um endereço IP, completando dessa forma os requisitos para acessar as páginas WAP na Internet através de um *Gateway* WAP.

Via de regra as opções de conexão, como o *gateway* que se deve conectar, o *login* e a senha, já estão configuradas nos telefones com suporte à WAP e a GSM. O usuário apenas inclui o URL do endereço WAP ao qual deseja acessar, como propomos, por exemplo, a URL do serviço de videomonitoração[REHM][REHM1], com as páginas de acesso tendo funções escritas em Java que possibilitarão visualizar as imagens no celular.

As figuras 2 e 3 descrevem os algoritmos propostos para a técnica de videomonitoramento.

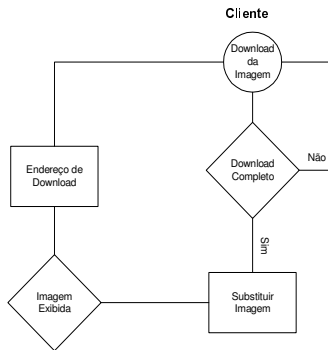


Figura 3: Algoritmo de Exibição de Vídeo

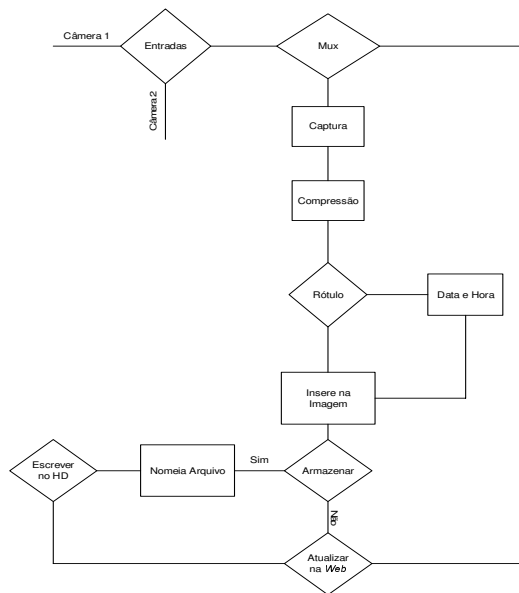


Figura 2: Algoritmo de Captura e Armazenamento de Vídeo

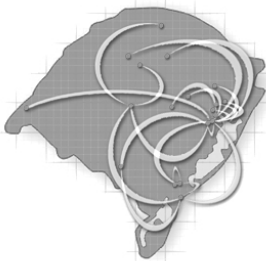
5. Conclusão

Este software foi escolhido por ser comprovadamente estável, e com algumas características especiais como o multiprocessamento e multitarefa. Estes aspectos multidisciplinares permitem a integração de aquisição de imagens e o sistema *webserver* na mesma plataforma. Além disso, o Linux é um sistema operacional *Free Software*, muito difundido na comunidade acadêmica e usado por desenvolvedores de software.

Durante o desenvolvimento da pesquisa a equipe encontrou questões à respeito da aplicação no celular. Nos questionamos em relação à qualidade do acesso através de um telefone celular, que tem algumas restrições, tais como o tamanho da tela, a memória do dispositivo e a navegação com um teclado reduzido. A resposta para esta questão foi respondida com o sistema desenvolvido em nosso laboratório. Esse sistema gera imagens em formato jpeg, ou seja, os arquivos gerados tem em média 1,5KB de tamanho. Considerando que os celulares WAP atuais tem a navegação em torno dos 7KBps, o tamanho da imagem não representa problema. Conseguiu-se o movimento do vídeo através de uma sequência de imagens (frames) enviadas pelo servidor em implementação, as quais são recebidas pelo aparelho celular e processadas em um aplicativo java para celular.

6. Referências

- [APACHE] APACHE HTTP Server FAQ. *Apache Software Foundation*.
Disponível em: <<http://httpd.apache.org/docs/misc/FAQ.html>>.
Acesso em: Março de 2001.
- [DORNAN] DORNAN, Andy, L. *Wireless Communication*. Primeira Edição.
Campus, 2001.
- [LINUX] LINUX. Home Page Online. Disponível em:
<<http://www.linux.org>>. Acesso em: Julho de 2001.
- [NOKIA] NOKIA. Disponível em: <<http://www.wapforum.org>>. Acesso
em Fevereiro de 2003.
- [REHM] REHM, R. J.; QUADRA, A.; GUEDES, J., Telemonserver: Um
Sistema de Videomonitoração Via Web. **Anais COBRAPI 2002,
2002 - Curitiba – PR.**
- [REHM1] REHM, R. J.; GUEDES, J., **Câmera Robô: Telemonitoração e
Controle de Câmeras de Vídeo Via Internet**. Anais COBRAPI
2002, 2002 - Curitiba – PR.
- [SOUSA] SOUSA, Maxuel Barbosa de, L. *Wireless – Sistemas de Rede
Sem Fio*. Primeira Edição. Brasport, 2002.
- [WIRTH] LIMA, Almir Wirth Lima Jr., L. **Telecomunicações Modernas**.
Segunda Edição. Book Express, 2001.
- [WAP] Rinaldo Demétrio, A tecnologia wap : aprenda a criar páginas
para celulares com a linguagem WML
Primeira Edição, Editora Érica, 2000.



Sessão Técnica 3

Segurança

Uma solução de Autenticação Fim a Fim para o LDP (Label Distribution Protocol)

Morvan D. Müller, Carlos B. Westphall, Carla Westphall

Laboratório de Redes e Gerência (LRG) - Universidade Federal de Santa Catarina (UFSC) - Tel:
+55.48.3317559, Caixa Postal 476, CEP 88040-970, Florianópolis, SC, Brasil

{morvan,westphal,carla}@lrg.ufsc.br

Abstract. *This works propose a solution for the LDP (Label Distribution Protocol) protocol from the MPLS architecture. The objective is authenticate in a trust way, on an end to end basis, the establishment of an LSP (Label Switching Path) between the Ingress LSR (Label Switching Router) and its Egress. It is intended to supply the LDP protocol deficiency, that doesn't have one end to end authentication mechanism defined for non-adjacent LSRs. (Key-words. LDP, security, MPLS).*

Resumo. *Este trabalho propõe uma solução de autenticação para o protocolo LDP (Label Distribution Protocol) da arquitetura MPLS. O objetivo é autenticar de forma confiável, em um escopo fim a fim, o estabelecimento de um LSP (Label Switching Path) entre um LSR (Label Switching Router) de Ingresso e o seu respectivo LSR de Egresso. Pretende-se suprir a deficiência do protocolo LDP de não possuir um mecanismo de autenticação fim a fim definido para LSRs não-adjacentes. (Palavras-chave. LDP, segurança, MPLS).*

1. Introdução

O MPLS (*Multiprotocol Label Switching*), RFC 3031 [ROSEN, 2001], é uma técnica de comutação de pacotes baseada em etiquetas (*labels*). O protocolo LDP (*Label Distribution Protocol*) é responsável pela distribuição dessas etiquetas e pelo estabelecimento dos caminhos lógicos, LSPs (*Label Switched Paths*) no MPLS. Uma lacuna na segurança do LDP pode comprometer todo o ambiente MPLS, pois a distribuição das etiquetas realizada pelo LDP é o que determina quem pode participar ou não do domínio MPLS. Existe uma autenticação definida para o LDP, RFC 3036 [ANDERSSON, 2001], baseada em TCP/MD5 [HEFFERNAN, 1998], porém a mesma é restrita a LSRs adjacentes pois depende de uma conexão TCP estabelecida entre os LSRs envolvidos. No caso de LSPs entre LSRs não-adjacentes, especialmente durante o estabelecimento do primeiro LSP, não existe uma conexão TCP fim a fim entre estes LSRs.

1.1. Trabalhos Correlatos

[DE CLERCQ, 2001], descreve uma proposta de autenticação fim a fim para o protocolo LDP, sugerida em forma de *draft* (atualmente expirada) para o IETF. Através de uma análise desta proposta conclui-se que a mesma apresenta um erro arquitetural, fato reconhecido pelos seus autores, por considerar erroneamente que ao enviar uma mensagem LDP solicitando um LSP para uma determinada FEC, o LSR de origem

(ingresso) sabe qual é o LSR de destino (egresso) que vai processar a requisição. Na maioria dos casos isso não é uma verdade dentro da forma padrão de operação do protocolo LDP.

[BUDA, 2001] aborda a segurança do MPLS e levanta a problemática da autenticação fim a fim no estabelecimento de LSPs pelo LDP. [WU, 2000] descreve uma solução que depende da confiabilidade dos LSPs criados entre LSRs não-adjacentes, pelo LDP.

2. A Solução de Autenticação Fim a Fim para o LDP

A solução deste trabalho faz uso de um mecanismo de autenticação baseado em criptografia assimétrica (chave pública e privada), anexado as mensagens LDP, o que possibilita ao LSR receptor verificar e autenticar o emissor das mensagens. Provê integridade às informações através de um mecanismo de resumo de mensagens (*hash*) e adicionalmente protege contra ataques de repetição através da inserção de um *nonce* as mensagens LDP. A solução não prove confidencialidade aos dados e foi planejada para ambientes onde LSPs atravessam múltiplos domínios externos, não confiáveis entre si, que por esse motivo necessitam de uma forma para autenticar as extremidades do LSP durante o seu estabelecimento. Como requisito a solução exige que o LDP esteja operando no Modo de Controle Ordenado e quanto aos modos de distribuição do LDP, "Sob Demanda" e "Não Solicitado", ambos são compatíveis com a solução proposta, a qual pode adicionalmente ser aplicada ao protocolo CR-LDP (*Constrained-Based Routing Protocol*) [JAMOUSSE, 2002] pelo fato do mesmo ser baseado no LDP.

2.1. TLVs e Tipos Definidos ao LDP pela Solução de Autenticação

Foram definidos dois novos TLVs (*Type-Length-Value*) ao LDP para prover a autenticação: "TLV de Nonce" e "TLV de Hash", e um novo "Código de Status" com o valor "Authentication Failed" para o TLV de Status do LDP, usado nas mensagens LDP Notification para anunciar que uma mensagem Label Mapping ou Label Request falhou na autenticação. As mensagens LDP envolvidas no processo de autenticação são: LABEL REQUEST, LABEL MAPPING e LDP NOTIFICATION. Os TLVs da autenticação são inseridos (no envio de mensagens LDP) e processados (no recebimento de mensagens LDP) em mensagens LABEL MAPPING, LABEL REQUEST e LDP NOTIFICATION, somente se o LSR se encontrar na condição de EGRESSO ou INGRESSO para a(s) FEC(s) em questão na mensagem LDP. Nos LSRs INTERMEDIÁRIOS os TLVs da autenticação são apenas repassados ao próximo LSR do caminho (*next hop*), mantendo a mesma ordem da mensagem original.

1.2.1 TLV de Hash

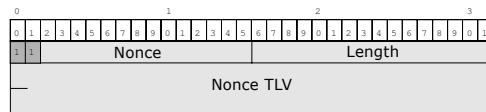


Figura 1. TLV de Hash

Este TLV transporta um resumo de mensagem (*hash*) cifrado. **U-bit e F-bit:** (1 bit cada) Atribuído em "1" indica ao LDP que ignore este TLV se o mesmo não for reconhecido e

o repasse para o próximo LSR do caminho. **Hash:** (14 bits) Este campo define o tipo do TLV, "TLV de Hash". **Length:** (2 bytes) indica o tamanho total em bytes dos seguintes campos: **LSR Identifier:** (6 bytes) identifica o LSR que originou a mensagem LDP, composto pelo LSR-ID (Identificador do LSR) e pelo espaço de etiquetas em uso pelo LSR. **Hash Digest:** (20 bytes) contém um valor Hash gerado a partir de uma mensagem LDP cifrado com a chave privada do LSR remetente. Na definição do tamanho deste campo foram considerados os algoritmos de *hash* (sha-1/160 bits) e de criptografia assimétrica "Curvas Elípticas", discutido na seção 0.

2.2.1 TLV de Nonce

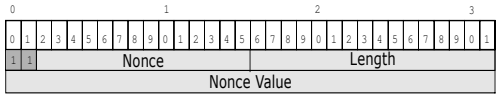


Figura 2. TLV de Nonce

Foi definido um novo TLV para transportar um valor de *nonce*. **U-bit e F-bit** seguem a mesma descrição do item 1.2.1. **Nonce:** (14 bits) Este campo define o tipo do TLV, "TLV de Nonce". **Length:** (2 bytes) indica o tamanho em bytes do campo "Nonce Value". **Nonce Value:** (8 bytes) armazena um valor nonce, de natureza incremental usado para detectar ataques de repetição.

2.2. O Modelo de Autenticação Proposto

Considerando que o LERA deseja estabelecer um LSP para uma FEC 10.1.0.0/8, prefixo IP o qual conhece via informações do seu roteamento IP. A Figura 3 ilustra o cenário onde o LERB (EGRESSO) autentica positivamente a mensagem de requisição LDP (*Label Request*) enviada pelo LERA (INGRESSO) e retorna uma mensagem LDP (*Label Mapping*) autenticada ao LERA, utilizando a solução de autenticação fim a fim que será descrita passo a passo nesta seção.

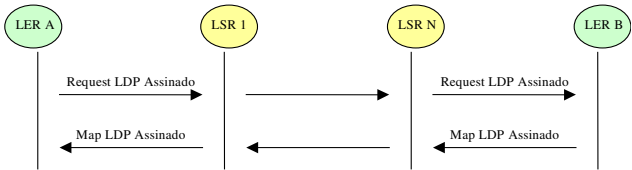


Figura 3. Diagrama da autenticação fim a fim.

Considere que o LDP está operando no modo de distribuição “Sob Demanda” e modo de controle “Ordenado” nos LSRs do ambiente e que os relógios do LERA e LERB estão sincronizados. Para SOLICITAR o LSP aplicando a solução de autenticação fim a fim, o LERA executa os seguintes passos:

- a) codifica uma mensagem LDP LABEL REQUEST solicitando uma etiqueta para a FEC 10.1.0.0/8, cuja rota de destino (*next hop*) é conhecida via seu roteamento IP;
- b) gera um valor nonce, codifica os campos do TLV de Nonce (Figura 2) e anexa o mesmo ao final da mensagem LDP.

c) codifica o TLV de HASH (Figura 1) baseado no conteúdo da mensagem LDP. No campo "LSR Identifier", o LERA insere o seu LSR-ID e o espaço de etiquetas (*labels*) que está usando. Codifica o campo "Hash Digest" que depende do tipo da mensagem LDP. Para mensagens LABEL REQUEST e LABEL MAPPING a entrada de dados é formada por um string de bytes conforme a Figura 4 e para para mensagens LDP NOTIFICATION uma string de bytes conforme a Figura 5:

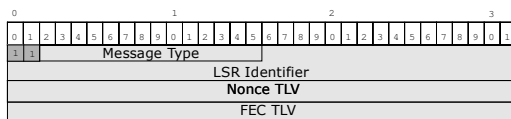


Figura 4. Entradas para mensagens LDP LABEL REQUEST e LABEL MAPPING

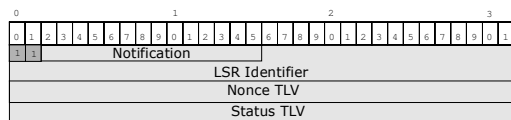


Figura 5. Entrada para mensagens LDP NOTIFICATION

Como o LERA está enviando uma mensagem LABEL REQUEST, forma um string conforme a Figura 4 e sobre esta aplica uma função hash (esta proposta considera o algoritmo "sha-1/160 bits"). Sobre o resultado da função *hash* aplica um algoritmo de criptografia assimétrica (esta proposta considera o algoritmo de "Curvas Elípticas") usando a sua chave privada (LERA) para a cifragem. Os resultados destas operações formam o campo "Hash Digest". Então o LERA anexa o TLV de Hash ao final da mensagem LDP e a envia ao próximo LSR do caminho LSR1 (*next hop*), descoberto via informações do seu roteamento IP;

Os LSRs INTERMEDIÁRIOS do caminho (LSR1 e LSRN) não podem atender a requisição solicitada (pois não possuem uma etiqueta para a FEC e não estão na condição de Ingresso/Egresso para a FEC), assim repassam os campos da autenticação e ao mesmo tempo mantêm um status de requisição pendente em relação a FEC a cada hop até a requisição alcançar o LER B, que ao receber a requisição LDP verifica que é o EGRESSO para a FEC então processa a autenticação.

Para PROCESSAR os Tlvs da autenticação fim a fim no recebimento da mensagem, o LERB executa os seguintes passos:

- identifica que o LERA é o emissor pelo campo "LSR Identifier" do TLV de Hash;
- verifica em sua configuração local se este LSR (LERA) está autorizado a estabelecer LSPs (**controle de autorização**), e em caso positivo seleciona a chave pública do LERA. Cada LSR que implementa a autenticação fim a fim possui em sua configuração local uma lista de controle de acesso a qual contém o "LSR-ID" e a chave pública correspondente dos LSRs autorizados a estabelecer LSPs com este LSR. Estas entradas são informadas manualmente via configuração local dos LSRs.
- decifra/valida o campo "Hash Digest", do TLV de Hash recebido, usando a chave pública do remetente (LERA). Caso obter sucesso significa que o remetente é AUTÊNTICO (**autenticação da origem**);

d) nos mesmos moldes do LERA (Figura 4), o LERB gera um Hash sobre a mensagem recebida e compara com o valor do campo "Hash Digest" do TLV de Hash recebido, assim pode verificar se mensagem recebida está ÍNTEGRA (**controle de integridade**);
e) gera um nonce local e compara este valor com o nonce recebido do LERA no campo "Nonce Value" do TLV de Nonce, aplicando algum mecanismo de verificação de nonce, [ABADI, 1996] descreve vários mecanismos de nonce, esta proposta não define um mecanismo de nonce específico. Se a verificação falhar a mensagem deve ser descartada. (**controle contra ataques de repetição**).

Se a autenticação ocorrer com sucesso, o LERB gera uma mensagem LDP LABEL MAPPING de resposta, gera/atribui uma etiqueta MPLS e executa os mesmos passos que o LERA: gera um *hash* da mensagem, cifra com a sua chave privada (LERB) e envia a resposta. A resposta é encaminhada através dos LSRs intermediários, geram um par de etiquetas (entrada/saída) correspondentes a cada hop até alcançar o LERA. Este por sua vez detecta que é o LSR de INGRESSO para FEC (10.1.0.0/8) e procede a autenticação do LERB. Para isso verifica se o LERB está autorizado a estabelecer LSPs, verifica sua autenticidade e a integridade da mensagem recebida. Baseado no resultado da autenticação executa ou não o estabelecimento do LSP com o LERB

Se a autenticação falhar, uma mensagem de notificação (LDP NOTIFICATION) com o código de status "*Authentication Failed*" será enviada em resposta para reportar a falha de autenticação.

Observe que ao solicitar o LSP o LERA conhece apenas a FEC (10.1.0.0/8) e o endereço IP do LSR1 (*next hop* para esta FEC). Ele não sabe quem será o LSR de egresso no domínio MPLS para esta FEC. Observe também que as mensagens Label Request e Label Mapping que criam o LSP, são trocadas via roteamento IP e apenas após o estabelecimento do LSP que os pacotes subsequentes serão roteados via MPLS, através das etiquetas geradas e autenticadas pelo LDP.

Os mecanismos adotados para prover a solução de autenticação foram salientados em negrito no texto acima: controle de autorização, autenticação da origem, controle de integridade e controle contra ataques de repetição.

Quanto a distribuição de chaves, cada LSR envolvido na autenticação precisa gerar e conhecer seu próprio par de chaves (pública e privada) e ambas as entidades LSR das extremidades do LSP (Ingresso e Egresso) devem conhecer a chave pública do LSR da extremidade oposta e inseri-la em sua lista de controle de autorização dessa forma autorizando que este LSR possa estabelecer LSPs. Sugere-se duas alternativas para distribuir as chaves públicas no ambiente: a) distribuição manual: informar as chaves públicas dos LSRs autorizados manualmente na configuração local dos LSRs. b) distribuição usando Certificação Digital: a solução está descrita em [MÜLLER, 2002].

2.3. Discussão sobre o Método de Autenticação Adotado

O método de autenticação adotado foi baseado em criptografia de chave pública por dois motivos: a) ao solicitar um LSP para uma FEC o LSR requisitante não sabe quem será o Ingresso (em caso de Label Mapping) ou Egresso (em no caso de Label Request), ou seja, não conhece o destinatário final.. A criptografia de chave pública resolveu essa problemática pois de posse da chave pública do LSRs autorizados, mantida na lista de controle de autorização em cada LSR, o LSR receptor pode verificar a assinatura do

LSR remetente. b) como o MPLS objetiva fast switching e alto desempenho, procurou-se evitar uma troca de mensagens adicional/inicial apenas para negociação de chaves de sessão e algoritmos para a autenticação, o que faria dobrar o tempo de estabelecimento do LSP. A solução adotada utiliza campos de controle (“TLV de Hash” e “TLV de Nonce”) “de carona” nas mensagens Label Request ou Label Mapping que o LDP usa para criar o LSP, assim não exige mensagens adicionais.

2.4. Algoritmos de Função Hash e Criptografia Assimétrica Recomendados

Para função *Hash*, sugere-se o algoritmo SHA-1 (*Secure Hash Algorithm*) [STALLINGS, 1999], com *digest* de 160 bits o qual gera como saída uma string com 20 bytes de tamanho. Para as funções de criptografia de chave pública sugere-se o algoritmo “Curvas Elípticas” [STALLINGS, 1999], que é rápido, trabalha com blocos pequenos e não acrescenta *overhead* a criptografia, ou seja, o hash cifrado se mantém com 20 bytes. O objetivo é gerar o mínimo *overhead* possível ao LDP.

2.5. Implementação do Protótipo em Linux

Foi realizada a implementação de um protótipo da solução descrita neste trabalho, utilizando um projeto de código fonte aberto que implementa o LDP e MPLS na plataforma linux. O projeto utilizado foi o “*MPLS for Linux*” (<http://sourceforge.net/projects/mpls-linux/>) [LEU, 2000] o qual está vinculado ao grupo “source forge” (<http://sourceforge.net>). O código está dividido em dois módulos principais MPLS-LINUX, que implementa o MPLS no kernel do linux, e LDP-PORTABLE, que implementa as funcionalidades do LDP. Alteramos o módulo LDP-PORTABLE, inserido o código necessário para implementar as funcionalidades da autenticação fim a fim apresentada neste trabalho. A linguagem utilizada foi o “C ANSI” (compilador gcc) e manteve-se a interface original do LDP-PORTABLE. As versões das ferramentas utilizadas na implementação do protótipo foram: Linux RedHat 7.2, linux kernel 2.4.19, LDP-PORTABLE versão 0.200 (<http://prdownloads.sourceforge.net/mpls-linux/ldp-portable-0.200.tar.gz?download>), MPLS-LINUX versão 1.170 (<http://prdownloads.sourceforge.net/mpls-linux/mpls-linux-1.170.tar.gz?download>) e ZEBRA versão 0.96 (<http://www.zebra.org>). Maiores detalhes a respeito da implementação estão descritos em [MÜLLER, 2002].

3. Conclusão

A solução apresentada traz incrementos importantes com relação à segurança do LDP. Uma forma de autenticação fim a fim, para viabilizar a autenticação mútua entre os LSRs de Ingresso e Egresso durante o estabelecimento de um novo LSP é de fundamental importância para a segurança LDP, principalmente em ambientes MPLS multi-domínio onde os domínios não são confiáveis entre si. Um exemplo clássico de ambiente multi-domínio MPLS é o provimento de VPNs baseadas em BGP/MPLS onde vários provedores VPN fornecem o serviço VPN a um cliente baseados em acordos (SLA's) que possuem entre si [ROSEN, 1999]. A solução apresentada possui um escopo de aplicação genérico e abrangente dentro do LDP, ou seja, se aplica a todas as situações de estabelecimento de LSPs, inclusive entre LSRs adjacentes e pode adicionalmente ser aplicada ao protocolo CR-LDP. Como perspectivas futuras, sugere-se avaliar os prós e contras de prover confidencialidade às informações transportadas pelo protocolo LDP.

4. Referências Bibliográficas

- ABADI, M.; NEEDHAM, R. "Prudent Engineering. Practice for Cryptographic Protocols" (1996). IEEE Transactions on Software Engineering, v. 22, n. 1, p. 6-15. (Disponível por <http://www.cs.virginia.edu/~survive/DOCS/prudent.ps>. Acesso em 10 set. 2002.)
- ANDERSSON, L.; Doolan, P., Feldman, N., et al. (2001) "LDP Specification". RFC 3036, Janeiro. (Disponível por <http://www.ietf.org/rfc/rfc3036.txt>. Acesso em 20 nov. 2001).
- BUDA, G.; CHOI, D.; et. al. (2001) "Security Standards for the Global Information Grid". IFIP/IEEE International Symposium on Integrated Network Management, Seattle, Maio.
- DE CLERCQ, J.; PARIDAENS, O.; TJOENSET Y., SCHRIJVER, P. (2001) "End to End Authentication for LDP". Draft-schrijvp-mpls-ldp-end-to-end-auth-03.txt. jeremy.de_clercq@alcatel.be, fevereiro. (Contato com o autor em 10 jan. 2002. Cópia disponível por <http://www.lrg.ufsc.br/~morvan/draft-schrijvp-mpls-ldp-end-to-end-auth-03.txt>. A draft no IETF expirou cf. <http://www.ietf.org/internet-drafts/draft-schrijvp-mpls-ldp-end-to-end-auth-04.txt>).
- HEFFERNAN, A. (1998) "Protection of BGP Sessions via the TCP MD5 Signature Option". RFC 2385, Agosto. (Disponível por <http://www.ietf.org/rfc/rfc2385.txt>. Acesso em 25 jan 2002).
- JAMOSSI, B, et al. (2002) "Constraint-Based LSP Setup using LDP". RFC 3212, Janeiro. (Disponível por <http://www.ietf.org/rfc/rfc3212.txt>. Acesso 12 fev. 2002).
- LEU, J; et. al. (2000) "Project: MPLS for Linux". Grupo Source Forge, Novembro. (Disponível por <http://sourceforge.net/projects/mpls-linux>. Acesso em 06 fev. 2002). (Trabalho em progresso).
- MÜLLER, M. (2002) "Uma Solução de Autenticação Fim a Fim para o LDP (Label Distribution Protocol)". Dissertação de Mestrado, Universidade Federal de Santa Catarina (UFSC), Centro Tecnológico (CTC), Florianópolis-SC, Brasil, Dezembro. (Disponível por <http://www.lrg.ufsc.br/~morvan/dissert-ldpauth.pdf>).
- NIST - *National Institute for Standards and Technology*. (2000) "Descriptions of SHA-256, SHA-384, and SHA-512". Outubro. (Disponível por <http://csrc.nist.gov/cryptval/shs/sha256-384-512.pdf>. Acesso em 20 mai. 2002).
- ROSEN, E.; REKHTER, Y. (1999) "BGP/MPLS VPNs". RFC 2547, Março. (Disponível por <http://www.ietf.org/rfc/rfc2547.txt>. Acesso em 13 set. 2001).
- ROSEN, E; VISWANATHAN, A.; CALLON, R. (2001) "Multiprotocol Label Switching Architecture". RFC 3031, Janeiro. (Disponível por <http://www.ietf.org/rfc/rfc3031.txt>. Acesso em 26 jul. 2001).
- STALLINGS, William. (1999) "Cryptography and Network Security: Principles and Practice". New Jersey, editora Prentice-Hall, Inc., 2ª ed.

Análise dos Processos de Segurança em Sistemas Móveis de 3ª Geração

Fabício Jorge Lopes Ribeiro¹, Jaime Cesar Ribeiro Lopes¹
Aloysio de Castro P. Pedroza¹

¹ Grupo de Teleinformática e Automação (GTA)

Universidade Federal do Rio de Janeiro

Programa de Engenharia Elétrica – Poli-COPPE/UFRJ

Caixa Postal. 68504 – CEP 21945-970 – Rio de Janeiro – RJ - Brasil

{fabricio, jaime, aloysio}@gta.ufrj.br

Resumo. Este artigo apresenta um estudo dos processos de segurança em Sistemas Móveis de Terceira Geração (3G), através do emprego de uma técnica de descrição formal utilizando a linguagem LOTOS. Estes protocolos são parte integrante de uma camada de serviços da arquitetura de um sistema móvel 3G, que provê mecanismos de Segurança. Os protocolos em estudo são empregados em atividades de consulta, que vão desde requisitos de confiabilidade que o sistema pode especificar até acordos de chaves de autenticação para usuários. A arquitetura utilizada para a implementação destes protocolos tem suas premissas definidas no fórum de discussão de 3G.

Abstract. This paper presents a study of security processes in 3rd Generation Mobile Systems using a formal description technique based on the LOTOS language. These protocols are part of a service layer of the architecture of a 3G mobile system, which provides security mechanisms. The protocols in study are used in information querying, such as reliability system provisioning capabilities and users authentication keys agreement. The architecture used in this implementation is well defined in the 3G-discussion forum.

1. Introdução

A necessidade de confiabilidade e segurança é preocupação constante em todos os ambientes, mas, na telefonia pública, o atendimento aos seus requisitos torna-se um grande desafio. Por outro lado, com o desenvolvimento da telefonia móvel celular, que propiciou a universalização nas comunicações de voz, a comunicação móvel de dados recebeu igualmente impulso no sentido de atingir sua ubiquidade.

Se a segunda geração (2G) trouxe a telefonia móvel para o mercado em geral, espera-se que a arquitetura de terceira geração (3G) estenda-se-á além da telefonia e abranja o fornecimento de comunicação de dados em alta velocidade e permita recursos multimídia. Essa arquitetura de terceira geração vem sendo desenvolvida pelo grupo de trabalho 3GPP (Third Generation Partnership Project).

A arquitetura de terceira geração deverá ser baseada em uma rede IP, já que este protocolo tornou-se o protocolo universal para comunicações em rede. O uso de pacotes IP na estrutura de transporte e sinalização vem se tornando naturalmente o caminho para a convergência entre as redes fixas e móveis. Esta convergência acontecerá pela

padronização de uma arquitetura totalmente baseada no protocolo IP, que incluirá o sistema celular, as redes fixas e as redes locais sem fio. Sendo assim, muitos dos requisitos de confiabilidade e segurança atualmente existentes, ou em definição, deverão também seguir os aspectos já adotados nas redes convencionais e deverão balizar o desenvolvimento dos seus análogos para redes móveis.

O desenvolvimento de um modelo padrão para protocolos de confiabilidade e segurança de terceira geração deverá basear-se, necessariamente, nos protocolos que compõe a arquitetura de segurança IP (IPSec) [Kent e Atkinson 1998] para garantir às redes sem fio a interoperabilidade aos serviços já utilizados nas redes atuais. A arquitetura necessária ao desenvolvimento de tais garantias deve ser organizada em camadas, cada uma delas provendo parte da segurança requerida. Tais camadas apresentarão um protocolo de sinalização e funções diversas de monitoramento no domínio da operadora, e cada uma delas terá seus requisitos definidos com base em padrões de confiabilidade desejados. Por outro lado, uma restrição importante é que, pelas limitações dos sistemas sem fio no que tange à vulnerabilidade dos canais, a sinalização deve apresentar alta complexidade e gerar pequeno volume de tráfego adicional.

Nesse sentido, mostraremos uma arquitetura de estudo das características e aspectos de integração deste processos de segurança. A especificação e a verificação de protocolos envolvidos nos processos de garantia de confiabilidade e segurança deve ser orientada por técnicas de descrição formal, empregando mecanismos e linguagens apropriados. As técnicas de descrição formal, por serem métodos de definição do comportamento de um sistema com o uso de uma sintaxe formal e uma semântica, permitem uma implementação de protocolos sem ambigüidades, precisa e completa. Além disso, provêm uma base bem definida para a verificação e validação desses protocolos, entendidas como a avaliação de conformidade dos mesmos com relação ao comportamento esperado [Fernandez, Garavel et al. 1996], [Bolognesi e Brinksma 1987].

O restante deste artigo está organizado nas seguintes seções: na seção 2, mostramos a importância da aplicação da técnica de descrição formal na verificação de segurança em protocolos de comunicação. Na seção 3, apresentamos os fundamentos da arquitetura de segurança de terceira geração que empregamos em nosso trabalho. A seção 4 apresenta a conclusão e temas para trabalhos futuros.

2. Comunicações Seguras dos Sistemas Móveis de 3ª Geração

Segurança, autenticação e controle de acesso são características vitais que devem ser encontradas nas comunicações em rede. Por outro lado, com a ênfase recentemente dada a novas aplicações baseadas em multimídia, levar em conta requisitos de confiabilidade tornou-se necessário à correta especificação de novos sistemas de comunicação. Estas necessidades são maiores em sistemas de comunicações móveis sem fio devido às restrições inerentes de largura de banda. Este meio geralmente é considerado não confiável, pois as mensagens estão sujeitas a perdas e interceptação durante a comunicação e restrições do meio.

Uma análise do comportamento dos protocolos que promovem as garantias de confiabilidade e segurança deve estar de acordo com os requisitos determinados pelo sistema. Um processo de verificação formal para protocolos se adequa a este esforço de

se atestar a confiabilidade e segurança de um sistema sem fio de comunicação. A especificação de um protocolo com o conceito de entidades confiáveis e não confiáveis torna-se viável devido à flexibilidade dos tipos de dados abstratos, que permitem a descrição de grande parte das operações seguras baseada no processo de modelagem do esquema clássico de segurança [Germeau e Leduc 1997].

O processo de validação e a formalização das propriedades de segurança definem uma ordem de estados que acarreta em uma comunicação segura, sendo esta ordem avaliada através das propriedades que são capazes de se expressar como eventos de segurança. No entanto, este processo pode acarretar modelos infinitos, sendo assim, necessário efetuar alguma simplificação.

O modo de transformação de método que acarreta estados infinitos em um sistema de estados finitos depende da limitação de números arbitrários das entidades envolvidas. A estrutura da especificação é composta por vários processos que interagem entre si através das portas de comunicação existentes neste protocolo. Cada entidade envolvida no protocolo é modelada pelo processo que descreve o seu exato comportamento.

Há um grande número de mecanismos de segurança, mas poucos deles são usados nos protocolos atuais. A análise dos tipos de mecanismos demonstra a confiabilidade e segurança de um protocolo [Leduc 2001]. O nível de abstração provido pelas técnicas de descrição formal e sua relativa simplicidade na definição do comportamento confiável e seguro de um protocolo são de grande valia na verificação de aspectos de segurança.

Muitas propriedades seguras podem ser verificadas [Pecheur, Zanetti et al. 1998]. Estas propriedades são estados que no protocolo podem acontecer sem prejuízo da segurança no sistema. A autenticação, o controle de acesso, a integridade e a aceitação são propriedades de segurança mas também são propriedades de confiabilidade. Cada um desses serviços de segurança necessita de um estado particular que pode acontecer ou não. A especificação formal permite, de um modo abstrato, a obtenção de todos os detalhes dos mecanismos de segurança. Assim, podemos focar somente nos serviços realmente seguros. A verificação de que todos os eventos no protocolo são seguros atesta a confiabilidade e a segurança deste protocolo.

3. Modelo UMTS para Domínios Seguros de Rede

Uma fraqueza identificada no sistema de 2a geração é a falta de segurança no núcleo da rede. Isto não foi tratado como um grande problema pois os sistemas de 2a geração eram compostas por sistemas proprietários e controladas por um número reduzido de instituições. Agora, com a introdução no backbone GPRS do IP [Rautpalo 2000], não somente usado para o tráfego de sinalização mas também para o tráfego de usuários, isto se traduz em novas ameaças e riscos para o sistema.

Os serviços seguros tem necessidade de confiabilidade, integridade, autenticação. Isto será assegurado com procedimentos padronizados e baseados em técnicas de criptografia, que possibilitam a implementação dos domínios seguros [Kent e Atkinson 1998].

Estes domínios são gerenciados por uma única autoridade que define a política de segurança que será implementada. O controle dos níveis de segurança é determinado

por esta política e implementado pelos dispositivos de borda (Security Gateways - SEGs). Os SEGs são responsáveis pela integridade e a autenticação dos dados de origem.

O domínio de rede UMTS deve ser dividido logicamente e fisicamente em domínios seguros. Este controle dos domínios seguros deve corresponder ao núcleo da rede e a sua separação deve ser realizada pelos roteadores de borda.

3.1. Roteadores de Borda Seguros

Os SEGs são entidades na borda dos domínios seguros que serão usados pelos protocolos baseados em IP [Rautpalo 2000], controlando as comunicações entre domínios diferentes (interface Za) e entre SEGs e entidades de rede internas no domínio (interface Zb).

Todo tráfego IP dos domínios seguros de rede deve passar por estes roteadores de borda antes de entrar ou sair dos domínios seguros, sendo o número destes dispositivos dependente do equilíbrio entre a necessidade de acessibilidade externa e o balanceamento de carga, para evitar um único ponto de falha. Eles são responsáveis por executar a política de segurança nas comunicações entre as redes (vide figura 1).

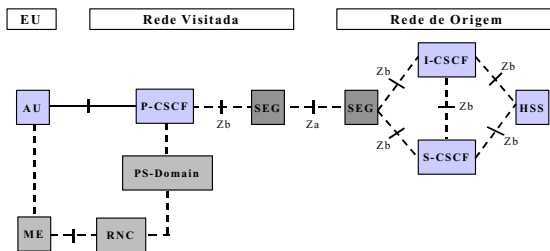


Figura 1: Arquitetura de Segurança entre Domínios Seguros

No modelo de segurança proposto para os sistemas de 3ª geração, estes dispositivos de borda devem ter capacidade de oferecer armazenamento seguro das chaves de autenticação, o que pode ser assegurado com a utilização de parte dos processos existentes no protocolo IKE (Internet Key Exchange – IKE) [Harkins e Carrel 1998].

Na arquitetura UMTS, o estabelecimento das SAs (Security Associations), poderá ser realizado pelo protocolo de troca de chave da Internet [Harkins e Carrel 1998]. Este protocolo tem como objetivo principal negociar, estabelecer e manter associações seguras.

Em uma típica comunicação segura entre dois SEGs, o gerenciamento e a distribuição das chaves também poderão ser baseados no protocolo ISAKMP (Internet Security Association and Key Management Protocol) [Maughan, Schertler et al. 1998], que é fundamental para o estabelecimento das duas associações IPsec.

O protocolo ISAKMP é um protocolo de transação, definido para prestar um serviço necessário ao atendimento a requisitos de troca de chaves criptográficas. Durante esta sessão de estabelecimento, os SEGs trocam informações para

estabelecimento de associações seguras. Há dois modos de operação (principal e agressivo) para estabelecimento destas associações. As características de segurança variam com a simples mudança no modo de operação.

Em muitos trabalhos de análise de protocolos, onde a criptografia é a base da garantia de segurança, verificamos que o ponto fraco se encontra justamente durante o processo de estabelecimento e negociação das características de segurança a serem adotadas entre as partes envolvidas. É neste ponto onde procuramos a vulnerabilidade nos processos seguros.

O número e a complexidade das mensagens compostas por cada estabelecimento, influenciam diretamente na ação de intrusos durante a interceptação, modificação e retransmissão de mensagens. O grande problema que encontramos é que, quanto maior a robustez de um sistema, mais difícil se dá a sua implementação devido a sua alta complexidade. Achar este equilíbrio é cada vez mais uma necessidade, levando em consideração a rápida evolução dos sistemas como o de 3a geração.

Os resultados obtidos com a verificação do comportamento do protocolo, em relação ao modelo do serviço, sugerem que a análise do número de estados, transições e rótulos do protocolo pode ser utilizada para aferir a sua maior eficácia em estabelecer comunicações seguras. A grande diferença encontrada na expansão dos estados e rótulos é que determina o grau de vulnerabilidade dos modos em relação a interceptações das mensagens.

4. Conclusão e Trabalhos Futuros

Neste trabalho apresentamos um processo de validação e a formalização de segurança, empregada na estrutura de uma camada de serviços, pertencente à arquitetura de sistemas móveis de terceira geração (3G). A validação tem como base a verificação da complexidade e robustez dos protocolos de segurança.

A análise da quantidade dos estados e a confirmação das propriedades observacionais permitem concluir que, pela modelagem formal de um protocolo, podemos aferir as suas propriedades de segurança e, através dos estados do protocolo, verificar eventuais falhas no procedimentos executados. As equivalências devem ser definidas para garantir que o protocolo modelado apresente, em termos observacionais, o mesmo comportamento seguro que se espera do serviço modelado.

Concluimos que a vulnerabilidade a ataques em algumas fases específicas do processo tem relação com a complexidade das mensagens e as variações de estados. Isto significa que, quanto maior a complexidade, maior a dificuldade de interceptação e replicação das mensagens por um intruso.

O resultado obtido coloca a descrição formal como ferramenta de grande contribuição para a validação de confiabilidade e segurança dos protocolos. Pode-se afirmar que trabalhos neste sentido serão fundamentais para o estabelecimento desta nova filosofia de comunicação no sistema móvel de terceira geração. Ao nosso ver, estaremos contribuindo com uma metodologia de trabalho e com a validação de protocolos de confiabilidade e segurança que ainda não foram testados no sistema móvel de terceira geração e assim verificando as garantias do atendimento aos requisitos de segurança que sejam necessários.

Referências

- Bolognesi, T. e Brinksma, E. (1987) “Introduction to the ISO specification language LOTOS”, *Computer Networks and ISDN Systems*, 14: 25–59.
- Fernandez, J. C., Garavel, H., Kerbrat, A., Mateescu, R., Mounier, L. e Sighireanu, M. (1996), in: Alur, R. e Henzinger, T. (Eds.), “CAESAR/ALDEBARAN Development Package: a protocol validation and verification toolbox”, *Proceedings of the Eighth Conference on Computer-Aided Verification*, LNCS, Springer Verlag, Berlin.
- Germeau, F. e Leduc, G. (1997) “Model-based Design and Verification of Security Protocols using LOTOS”.
- Harkins, D. e Carrel, D. (1998) “The Internet Key Exchange (IKE)”, IETF RFC 2409.
- Kent, S. e Atkinson, R. (1998) “Security Architecture for the Internet Protocol”, IETF RFC 2401.
- Leduc, G. (2001) “Verification of two versions of the Challenge Handshake Authentication Protocol”.
- Maughan, D., Schertler, M., Schneider, M. e Turner, J. (1998) “Internet Security Association and Key Management Protocol (ISAKMP)”, RFC 2408.
- Pecheur, C., Zanetti, D., Koerner, E., Leduc, G., Léonard, L. e Bonaventure, O. (1998) “Model-based Design and Verification of Security Protocols using LOTOS”.
- Rautpalo, J. (2000) “GPRS Security - Security Remote Connections over GPRS”.

Análise das ferramentas de IDS SNORT e PRELUDE quanto à eficácia na detecção de ataques e na proteção quanto à evasões

Julio Steffen Junior¹, Eduardo Leivas Bastos²

¹Bacharel em Ciência da Computação, ²Prof. Esp. Ciência da Computação

Centro Universitário FEEVALE
RS 239, 2755 - Cep 93352-000 - Novo Hamburgo - RS – Brasil

steffen@tca.com.br, elbastos@acm.org

Abstract. *In order to protect computer networks from attacks, many security tools have been developed. One class of these tools is usually called Intrusion Detection Systems (IDS), which are tools able to detect possible attacks, to produce specific alerts and to take corrective actions in order to prevent that the attack really takes place. This work has as main goal to present a study about IDSs and to perform some experiments with two different IDS tools. The experiments are oriented to evaluate the behavior of these IDSs tools when they are exposed to different attacks generated by means of some tools available in the Internet.*

Resumo. *A necessidade de proteger a estrutura de rede contra ataques gerou um conjunto de ferramentas que visam proporcionar esta proteção. Uma ferramenta que faz parte deste conjunto é conhecida como Sistemas de Detecção de Intrusão (Intrusion Detection Systems - IDS), o IDS é uma ferramenta que auxilia na detecção de ataques alertando e realizando ações que possam impedir que um ataque seja concretizado. Este trabalho teve como objetivo apresentar um estudo da ferramenta de IDS e realizar um teste prático com duas ferramentas de IDS diferentes visando verificar o comportamento destas ferramentas frente a ataques realizados com ferramentas disponíveis em sites na Internet.*

1. Introdução

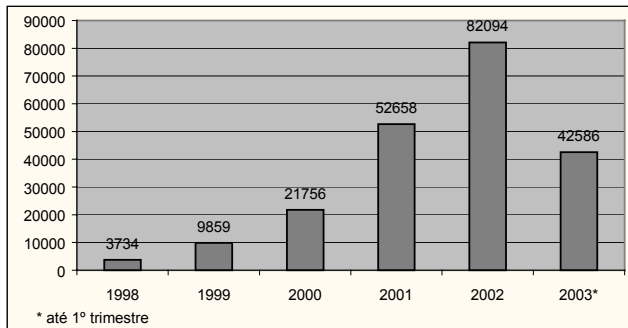
A segurança é uma das maiores preocupações enfrentadas atualmente pelos administradores de redes. Manter a empresa longe de ataques é um desafio cada vez maior para evitar o roubo de informações e a paralisação de sistemas. Somente um *firewall* já não garante que a rede esteja 100% segura, a monitoração contra ataques e intrusões, deste modo, tornou-se ponto chave na estrutura de segurança de uma rede de computadores, auxiliando o administrador da rede a prevenir ataques e a agir quando um ataque é iniciado ou detectado.

Segundo estatísticas da CERT (Computer Emergency Response Team) os ataques a redes crescem a cada ano como mostra a Tabela 2.1. Como um dos fatores para este crescimento é possível citar a sofisticação das ferramentas de ataques existentes nos dias de hoje e como consequência desta sofisticação, houve um aumento no número de pessoas que podem vir a cometer um ataque. [ALLEN, 2000].

Sistemas de Detecção de Intrusão (*Intrusion Detection Systems - IDS*) são ferramentas cuja finalidade é executar uma monitoração da rede e tentar impedir que o ataque cumpra com seus objetivos.

Tabela 2.1—Aumento do número de ataques

Fonte: CERT/CC Statistics 1988-2003



O IDS pode ser visto como mais uma ferramenta para reforçar a política de segurança da informação de uma empresa. A escolha de qual ferramenta utilizar é uma decisão difícil de ser tomada, basear-se apenas no custo é minimizar outros aspectos relevantes da questão, tais como: funcionalidades oferecidas, facilidade de configuração e gerenciamento, eficácia na detecção, entre outros.

Esse artigo aborda as principais características de um sistema de intrusão e tem como objetivo descrever uma comparação realizada entre duas ferramentas de IDS de código-aberto (*open source*) no que diz respeito à eficácia na detecção de diversos tipos de ataques.

A seção 2 aborda, de maneira clara e objetiva, como o IDS funciona, bem como os tipos, métodos e problemas encontrados em uma ferramenta de IDS. A seção 3 apresenta as duas ferramentas utilizadas nos testes. A seção 4 comenta os testes realizados e aborda os resultados obtidos.

2. Sistemas de Detecção de Intrusão – IDS

Detecção de intrusão é um processo de coleta de informações que procura identificar sinais de que um ataque está iniciando ou ocorrendo. “(...) a detecção de intrusão da rede permite identificar e reagir a ameaças contra o seu ambiente (...)” [NORTHCUTT, 2002, p. 156].

Um IDS é composto basicamente por dois dispositivos principais, o Console de Comando e o Sensor. O console de comando tem como função permitir o controle do IDS, monitorar o estado do sensor e processar os alertas enviados pelo sensor [PROCTOR, 2001]. “O sensor é o dispositivo responsável pela coleta de informação para análise de descoberta de uma invasão” [CROTHERS, 2003, p. 275].

O IDS pode trabalhar basicamente de duas maneiras, uma é analisando o tráfego da rede (Baseado em Rede), onde o sensor analisa todos os pacotes que circulam pelo segmento de rede independente de qual era o destino do pacote. A outra forma é analisando uma determinada máquina (Baseado em *Host*) a procura de códigos maliciosos para identificar sinais de que um ataque está sendo iniciado.

Quanto ao método que o IDS detecta os ataques, ele pode ser classificado como Baseado em Assinatura ou Baseado em Anomalia. O IDS baseado em assinatura trabalha procurando regras pré-estabelecidas no tráfego da rede. Quando é encontrado algum código na rede que esteja descrito em alguma regra, é gerado um alerta ou evento que permita uma ação defensiva [NORTHCUT, 2002]. Já o IDS baseado em anomalia, possui uma base de dados do comportamento da rede, a partir desta base é que o sistema verifica o que é ou não permitido e quando encontra algo fora do padrão gera o alerta.

Em Sistemas de Detecção de Intrusão é comum a incidência de falsos positivos e falsos negativos. O falso positivo ocorre quando um sensor classifica uma atividade normal na rede como sendo um ataque [NORTHCUTT, 2002], enquanto que o falso negativo, ocorre quando um sensor não gera nenhum alerta em uma condição real de ataque [PROCTOR, 2001], sendo sua ocorrência mais perigosa do que a do falso positivo.

Outro problema que merece uma atenção especial, são as técnicas de Evasão. Essas técnicas consistem basicamente em métodos que procuram enganar o IDS de forma a fazer com que um ataque real passe despercebido. Existem inúmeras técnicas de evasão, e a necessidade de formas de evita-las tornou-se uma constante entre os desenvolvedores das ferramentas de IDS, porque o poder de detecção da ferramenta fica comprometido se ela não for capaz de reconhecer essas técnicas.

3. Ferramentas Utilizadas

Para a realização dos testes foram selecionadas duas ferramentas de IDS, a seleção foi baseada em critérios pré-estabelecidos (é importante definir requisitos e critérios condizentes com a estrutura da empresa onde o IDS será instalado), um dos critérios definidos era que a ferramenta deveria possuir seu modelo de licença de *software* baseada na *GNU General Public License (GPL)*. As duas ferramentas selecionadas foram o Snort e o Prelude.

O Snort é um sistema de detecção de intrusão baseado em rede amplamente utilizado, possui uma arquitetura simples baseada em *plugins*, onde executa basicamente as funções de captura de pacotes na rede, análise dos pacotes e geração de alertas. É um sistema leve, capaz de trabalhar em grandes redes e detectar uma grande variedade de ataques em tempo real, sendo o seu sistema de detecção baseado em assinaturas [CAMPELLO, 2002].

O Prelude é uma ferramenta de IDS híbrida, pode trabalhar como um IDS baseado em rede ou como um IDS baseado em *host* ou ainda das duas formas ao mesmo tempo. Como o Prelude é composto por módulos, é possível instalar somente o módulo desejado e condizente com a necessidade [TRICAUD, 2002]. O Prelude, assim como o Snort, também possui seu sistema de detecção baseado em assinatura.

4. Análise Prática

O objetivo desta análise foi verificar o funcionamento e o comportamento do IDS em um ambiente de rede simulado. Com esta análise foi possível estudar melhor o funcionamento do IDS frente a diferentes tipos de ataques que foram realizados.

Com o intuito de realizar um estudo comparativo fiel entre as duas ferramentas, ambas foram submetidas aos mesmos ataques sob uma configuração *default* da ferramenta. Após os testes, uma base de dados com uma análise comparativa entre as ferramentas foi gerada com a finalidade de demonstrar a eficiência na detecção dos vários ataques. Porém, este comparativo não tem o intuito de definir qual ferramenta é melhor, pois seria necessário uma estrutura de testes com recursos mais sofisticados, maiores investimentos e um período de testes superior ao executado para a obtenção de resultados mais precisos.

Durante os testes, alguns itens foram observados em relação às ferramentas: qual o comportamento da ferramenta utilizando-se a configuração *default*, a quantidade de falsos positivos e falsos negativos gerados, a capacidade de detecção de técnicas de evasão e a capacidade de detecção sob diferentes níveis de utilização da rede.

Os testes foram executados em um laboratório exclusivamente montado para a ocasião. Todas as máquinas envolvidas nos testes foram preparadas e configuradas exclusivamente para os testes de modo que não influenciassem nos resultados. Uma metodologia foi utilizada com o objetivo de padronizar os testes e definir quais os testes que seriam realizados, os tipos de ataques, as ferramentas usadas no teste e como seriam realizados os testes.

Os testes foram divididos em três categorias [NSS GROUP, 2002]:

- Reconhecimento de ataques: foi verificado a capacidade da ferramenta em detectar determinados tipos de ataques (*Buffer overflows* e *exploits*, *Denial of service*, ataques de HTTP, SMTP e FTP e ferramentas de *scanner*);
- Performance: foi analisado a capacidade da ferramenta de IDS em detectar os ataques com diferentes taxas de utilização da rede;
- Técnicas de evasão: foi verificado a capacidade da ferramenta em detectar as técnicas de evasão.

Cada ferramenta de IDS foi testada separadamente para que uma não influenciasse no resultado da outra. O primeiro teste realizado teve a taxa de utilização da rede em 0%, neste teste foi criada a *baseline* dos ataques realizados versus os ataques detectados. Os testes seguintes realizados foram os com taxa de utilização de 25% e 75%, o teste de evasão e o teste de falso positivo. Os testes de falso negativo ocorreram junto com os testes de reconhecimento de ataque e performance, porque sempre que um ataque é gerado e não é detectado pelo IDS ocorre um falso negativo.

Os ataques foram realizados individualmente, um após o outro, para que se tivesse certeza de que o ataque detectado, quando detectado, correspondia ao ataque gerado. Após o alerta de ataque ser gerado no console de gerenciamento ele foi devidamente documentado para análise posterior e comparações entre os resultados obtidos pela mesma ferramenta, bem como comparações entre os resultados das duas ferramentas de IDS e para que ao final dos testes pudesse ser montada uma planilha demonstrativa com os resultados obtidos. Antes de um novo ataque ser iniciado o alerta anterior era apagado.

Na tabela 4.1, é possível verificar que nenhuma das duas ferramentas obteve 100% de aproveitamento com taxa de utilização da rede em 0%. O esperado era que com taxa de 0% todos os ataques fossem detectados, pois não havia nenhum fator que pudesse contribuir para que as ferramentas falhassem ao detectar qualquer um dos

ataques. Outro ponto importante é que os números de detecção entre as duas ferramentas se alteraram sensivelmente quando o uso da taxa de rede se tornou presente nos testes.

Tabela 4.1 – Reconhecimento de ataque e Performance

Ferramenta SNORT									Ferramenta PRELUDE										
Tráfego	0%			25%			75%			Tráfego	0%			25%			75%		
Ataque	G	D	%	G	D	%	G	D	%	Ataque	G	D	%	G	D	%	G	D	%
DOS	10	8	80	10	8	80	10	8	80	DOS	10	8	80	10	8	80	10	8	80
http	10	6	60	10	1	10	10	2	20	HTTP	10	10	100	10	10	100	10	8	80
SMTP	10	10	100	10	0	0	10	1	10	SMTP	10	10	100	10	10	100	10	10	100
FTP	10	5	50	10	2	20	10	0	0	FTP	10	5	50	10	5	50	10	5	50
BO/Exploit	10	8	80	10	2	20	10	2	20	BO/Exploit	10	6	60	10	6	60	10	6	60
Portscan	10	8	80	10	4	40	10	8	80	Portscan	10	6	60	10	6	60	10	6	60
Scan Vuln.	10	10	100	10	9	90	10	10	100	Scan Vuln.	10	10	100	10	10	100	10	10	100

G= Ataques Gerados

D= Ataques detectados

BO= Buffer Overflow

De todas as ferramentas de ataques utilizadas existiram quatro, de tipos de ataques diferentes, que não foram detectadas por nenhuma das duas ferramentas de IDS. Este fato demonstra que existem centenas de ferramentas de ataques disponíveis e que a constante atualização e criação de novas regras é necessária e que é difícil estar 100% protegido e fica evidente a ocorrência dos falsos negativos. Outro ponto a ser observado é que a ferramenta Snort mostrou uma deficiência maior na detecção dos ataques quando submetida a taxas de utilização da rede mais elevadas, quando comparada com os resultados da ferramenta Prelude. Seria necessário um estudo mais aprofundado para determinar qual o real motivo que influenciou a queda tão significativa do poder de detecção da ferramenta Snort, se foi o tipo de tráfego, o método usado pela ferramenta para analisar o tráfego ou a necessidade de otimização em suas configurações após a instalação.

Na tabela 4.2 é possível observar os resultados dos testes de evasão usando a ferramenta *Fragroute* (permite criar regras para modificar os pacotes enviados). Neste teste a ferramenta Prelude mostrou uma maior deficiência em relação à ferramenta Snort, conseguindo detectar apenas o ataque de DoS (*Denial of Service*).

Tabela 4.2 – Teste de evasão usando *Fragroute*

Tráfego	Ferramenta SNORT									Ferramenta PRELUDE								
	0%			25%			75%			0%			25%			75%		
	G	D	%	G	D	%	G	D	%	G	D	%	G	D	%	G	D	%
Ataque																		
Slice (DoS)	3	3	100	3	3	100	3	3	100	3	3	100	3	3	100	3	3	100
WUFTP (Exploit)	3	3	100	3	3	100	3	3	100	3	0	0	3	0	0	3	0	0
Simplestealth (Portscan)	3	3	100	3	1	33	3	3	100	3	0	0	3	0	0	3	0	0

G= Ataques Gerados

D= Ataques detectados

Na tabela 4.3 é encontrado os resultados dos testes de evasão usando a ferramenta *Nikto* (*scanner* que procura por vulnerabilidades em servidores *WWW*). Como é possível verificar, o número de técnicas existente é bem numeroso e ambas as ferramentas não foram capazes de detectar 100% das técnicas de evasão, apresentando novamente, variações em sua capacidade de detecção quando submetidas ao teste com taxa de utilização da rede, o que originou um alto índice de falsos negativos. Por esse motivo é que as técnicas de evasão são consideradas perigosas, pois os números de ataques não detectados são bem elevados.

Tabela 4.3 – Teste de evasão usando Nikto

Ferramenta SNORT - Teste de Evasão usando Nikto									
Tráfego	0%			25%			75%		
Técnicas	Checadas	Alertas Gerados	%	Checadas	Alertas Gerados	%	Checadas	Alertas Gerados	%
URL encoding	907	195	21,5	907	13	1,43	907	20	2,21
././ directory insertion	1420	572	40,3	1420	15	1,06	1420	77	5,42
Long URL	1420	1185	83,5	1420	13	0,92	1420	169	11,9
Fake parameter	1420	1262	88,9	1420	28	1,97	1420	166	11,7
Ferramenta PRELUDE - Teste de Evasão usando Nikto									
Tráfego	0%			25%			75%		
Técnicas	Checadas	Alertas Gerados	%	Checadas	Alertas Gerados	%	Checadas	Alertas Gerados	%
URL encoding	907	552	60,9	907	326	35,9	907	462	50,9
././ directory insertion	1420	726	51,1	1420	424	29,9	1420	523	36,8
Long URL	1420	1854	131	1420	643	45,3	1420	650	45,8
Fake parameter	1420	2266	160	1420	795	56	1420	1445	102

Como já comentado no início do artigo, a segurança de uma estrutura de rede depende de um conjunto de ferramentas e como pode ser observado nos resultados contidos nas tabelas, o IDS sozinho não pode garantir a total segurança de uma rede, pois apresenta alguns problemas que o impede de ser 100% confiável. Mas, é claro que mesmo com os problemas de falsos positivos, falsos negativos e evasão, sua presença na estrutura de segurança de uma rede é importante, pois fornece informações e ações que sem o IDS seriam difíceis de serem detectadas e tomadas. O que é necessário, é que sejam realizadas mais pesquisas com o intuito de procurar solucionar os problemas existentes em uma ferramenta de IDS, afim de torná-las mais representativas no escopo de proteção das redes de computadores.

Referências bibliográficas

ALLEN, Julia et al. **State of the Practice of Intrusion Detection Technologies**, 2000. Disponível em: <<http://www.cert.org/archive/pdf/99tr028.pdf>>. Acesso em: 18 set. 2002.

Campello, Rafael Saldanha; WEBER, Raul Fernando. **Sistemas de detecção de intrusão**. Disponível em: <<http://www.inf.ufrgs.br/~gseg/producao/minicurso-ids-sbrc-2001.pdf>>. Acesso em: 10 nov. 2002.

CERT - Computer Emergency Response Team. **CERT/CC Statistics 1988-2003, 2003**. Disponível em: <<http://www.cert.org/stats/>>. Acesso em: 25 mai. 2003.

Crothers, Tim. **Implementing Intrusion Detection Systems: A Hands-on Guide for Securing the Network**. Indianapolis: Wiley Publishing, 2003. 297p.

Northcutt, Stephen; ZELTSER, Lenny et al. **Desvendando segurança em redes**. Rio de Janeiro: Editora Campus, 2002. 650p.

NSS GROUP. **Intrusion detection systems, Group test (edition 3)** 2002. Disponível em: <<http://www.nss.co.uk/ids/edition3/index.htm>>. Acesso em 11 nov 2002.

Proctor, Paul E. **The practical intrusion detection handbook**. New Jersey: Prentice-Hall PTR, 2001. 359p.

Tricaud, Sebastien. **Prelude’s FAQ, v0.5**. Disponível em: http://www.prelude-ids.org/article.php3?id_article=8. Acesso em: 25 abr 2003.

Compreendendo Ataques *Denial of Services*

Leandro Márcio Bertholdo, Andrey Vedana Andreoli e Liane Tarouco

Computer Emergency Response Team of RS – CERT-RS
Centro de Processamento de Dados da Universidade Federal do Rio Grande do Sul
Rua Ramiro Barcelos, 2574 – Porto Alegre – RS – Brasil
{berthold, andrey, liane}@penta.ufrgs.br

Resumo.

Esse trabalho aborda o tópico de ataques de negação de serviço (DoS), inclusive distribuídos (DDoS). Nele são analisados alguns dos ataques clássicos e outros que ainda são utilizados, concentrando-se no que os autores consideram como os mais nocivos desta classe: os ataques distribuídos. São citadas classificações, características e contra-medidas que são utilizadas e/ou pesquisadas em relação à detecção e bloqueio dessa classe de ataques.

1. Introdução

Os ataques conhecidos como denial-of-service (DoS) são caracterizados por uma tentativa explícita do atacante de impedir que um usuário legítimo utilize determinado serviço [CERT 97]. Algumas estratégias utilizadas nesses ataques são:

- Inundar uma rede visando impedir que usuários legítimos façam uso dela.
- Impedir ou romper a conexão entre duas máquinas visando impedir o acesso a um serviço.
- Impedir o acesso de um determinado serviço ou site.
- Impedir ou negar um serviço a um sistema ou pessoa específicos.

Conceitualmente, nem todos os ataques contra serviços são necessariamente ataques de negação de serviço. Em outros casos, alguns tipos de ataques podem incluir um componente de negação de serviço como parte de um ataque maior, como o caso Mitnick [NOR 99].

Os ataques distribuídos possuem conceitos semelhantes aos de sistemas distribuídos, ou seja, são ataques que podem ser efetuados a partir de diversos computadores simultaneamente. Neste tipo de ataque é realizada uma sobrecarga ou inundação de pacotes contra um determinado serviço, host ou rede, gerando muitas vezes uma quantidade de dados global maior que a rede ou host pode suportar, tornando a rede ou serviços instáveis e conseqüentemente prejudicando o seu desempenho [NEU 99] [CERT 00].

Segundo estatísticas do CERT/CC [CERT 01], vários ataques DoS e DDoS são registrados diariamente, e envolvem principalmente novos vermes e ferramentas para DDoS. Alguns desses vermes incluem um comando e estrutura de controle que permite ao intruso dinamicamente modificar o comportamento do verme após ele infectar a

vítima. Em alguns casos esse controle é inclusive realizado sem que o atacante necessite saber quem são os sistemas que foram infectados – foram registrados casos onde o host infectado monitora um canal IRC aguardando a ordem para atacar. Características como essa tornam ainda mais difícil desenvolver uma solução global para o problema. Ferramentas como essa possuem somente um objetivo: terrorismo.

2. Características

Os ataques de denial-of-service surgiram explorando falhas de implementação em serviços e sistemas operacionais, como Ping-of-death¹ [CERT 96] e, em alguns casos, até mesmo a forma de funcionamento dos protocolos, como no caso do SYN Flooding² [CERT 96b] e do UPD packet storm³ [CERT 96c]. Em um segundo momento, surgiram os amplificadores de ataque, tais como os ataques smurf⁴ [CERT 98] utilizando técnicas de IP Spoofing [CERT 95].

Com o passar dos anos todas as técnicas de ataques conhecidas (spoofing, flooding, amplificadores de ataques, etc.) acabaram por ser incorporadas nos ataques DoS, e esses, por sua vez, realizados de forma distribuída. Para se chegar a esse nível de contaminação, o conceito utilizado foi o de vermes e viroses, através do qual uma vulnerabilidade explorada na máquina do usuário injeta um código malicioso que aguarda as ordens do atacante sobre o alvo a ser atingido e os ataques a serem realizados. Em muitos casos antigas ferramentas como Trinoo, TFN e Mstream [CERT99][CERT 99b] em novas formas utilizadas, acrescidas do requinte de trocar as informações de ataque de forma cifrada.

3. Ferramentas utilizadas para realizar ataques DoS e DDoS

Basicamente o ataque DDoS caracteriza-se por, primeiramente, explorar vulnerabilidades já conhecidas em sistemas operacionais e serviços e, através delas, obter acesso privilegiado a qualquer máquina na Internet. Geralmente esse acesso indevido é obtido através de scripts automatizados que varrem toda a Internet a procura de hosts vulneráveis. No passado, quando as ferramentas para DDoS eram instaladas manualmente, a preferência era por hosts bem conectados e sistemas operacionais que permitissem a instalação de sniffers e rootkits. Hoje em dia, com o advento de conexões domésticas de banda larga (ADSL, Cable, ISDN) essa preferência não existe mais.

¹ O ping-of-death era caracterizado por gerar um buffer overflow quando o host atacado recebia um pacote ICMP de tamanho superior a 65535 bytes, causando uma reinicialização ou desativação do sistema operacional do host atacado.

² Nesse caso o host era atacado com inúmeras tentativas de estabelecer uma conexão para um serviço (pacotes SYN) e não aceitava novas conexões até obter uma resposta das que estavam nesse estado. Essa resposta nunca era enviada e as conexões somente recebiam um timeout após alguns minutos – mas o atacante enviava constantemente outros pedidos de conexão.

³ Uma das formas desse ataque faz uso da porta udp/echo, que faz com que o host solicitado envie indefinidamente e na maior velocidade possível pacotes para a estação solicitante.

⁴ O ataque smurf se aproveita da existência de um endereço de broadcast direto, que permite que uma estação remota solicite uma resposta de todas as estações em uma determinada rede. Como o atacante forja o endereço da estação conectada e emite um fluxo constante de requisições, um único pacote enviado pelo atacante pode facilmente ser multiplicado por 100 vezes ou mais.

Depois de realizado esse primeiro passo, a invasão do sistema, e considerando-se as versões originais do TFN e Trinoo, uma lista de endereços IPs das máquinas exploradas formam a rede de ataque. Neste ponto, cada uma das máquinas listadas já possui instalado o software necessário para efetuar o ataque propriamente dito, ou seja, as ferramentas para DoS.

No passo seguinte é criada uma hierarquia de ataque, composta pelos atacantes, estações mestres e estações zumbis. As máquinas consideradas mestres eram responsáveis por receberem os comandos de ataque, reenviando-os às estações zumbi. Este grupo de zumbis era quem efetivamente concretizava o ataque.

Uma vez que o software estivesse instalado nos futuros zumbis, eles passavam a anunciar ao mestre a sua presença. Assim, para efetuar o ataque bastava que o mestre fornecesse o IP a ser atacado e o tempo que duraria o ataque. A partir desse ponto o zumbi entrava em atividade. Uma consequência comum desse ataque era a saturação do link ou a paralisação dos serviços oferecidos pela vítima através de técnicas como SYN Flooding, Smurf ou simplesmente pacotes ICMP destinados à vítima. Veja abaixo a ilustração de uma rede de ataque:

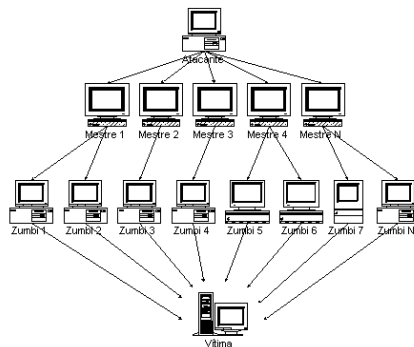


Figura 1 - Exemplo de rede de ataque para DDos

4. Evoluções dos Ataques DoS

Nos últimos anos, vermes com um forte potencial nocivo do ponto de vista de negação de serviço foram identificados. Vermes cujo problema gerado foi a sobrecarga na infraestrutura da rede, tanto no que tange a utilização de cpu de roteadores e switches quando na utilização da banda disponível, sem citar-se o grande número de hosts contaminados e um tempo muito curto [NEU00][CER03][CER03b].

Dentre esses podemos citar o Codered-v2 que em menos de 24h contaminou aproximadamente 350 mil hosts e o W32.Slammer que em 25 de janeiro de 2003 contaminou 75 mil hosts em aproximadamente 30min [CAI 02]. Em ambos os casos vários nodos da rede caíram devido à alta utilização de CPU, e o restante sofreu devido à massiva utilização da rede. Para se ter uma idéia do impacto e do número de redes envolvidas pela ação do W32.Slammer, a figura 2 abaixo mostra a oscilação das tabelas de roteamento BGP de alguns dos maiores provedores mundiais.

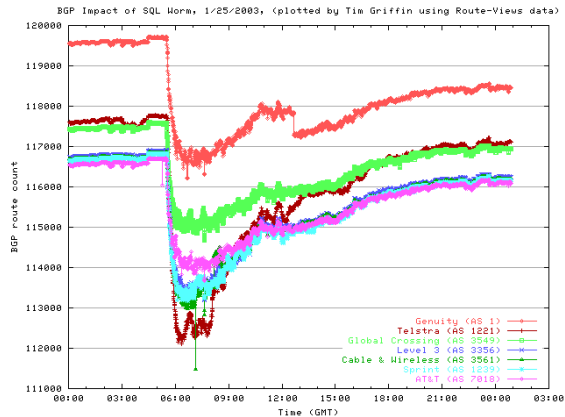
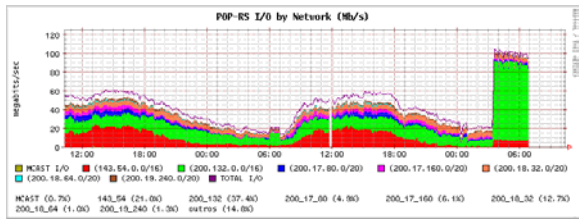


Figura 2: Impacto no protocolo BGP durante a ação do Slammer

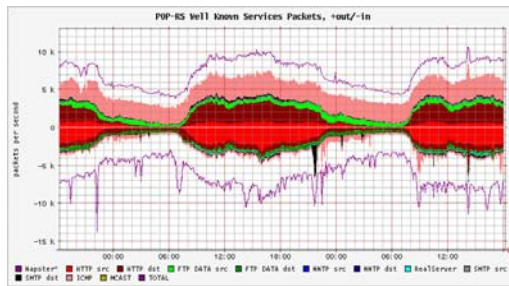
5. Como detectar um ataque DoS

Algumas anomalias podem sinalizar a ocorrência deste tipo de ataque, tais como :

- Excesso de tráfego: A banda utilizada excede o máximo, ultrapassando o número de acessos esperados ou a assimetria deste.



- A existência de pacotes UDP e ICMP de tamanho acima do normal ou em excesso: Geralmente as sessões UDP utilizam pacotes pequenos de dados dificilmente maiores que 10 bytes (payload). As mensagens ICMP não excedem a faixa entre 64 e 128 bytes. Pacotes cujo tamanho seja superior a esses números são considerados suspeitos de conter mensagens de controle, destinadas a cada um dos agentes que está participando do ataque. Apesar do conteúdo dos pacotes estar cifrado, o endereço do destino é verdadeiro, desta forma pode-se localizar um dos agentes que estão realizando o ataque baseado no seu fluxo de mensagens.



- Pacotes TCP e UDP que não fazem parte de uma conexão: Alguns tipos de DDOS utilizam aleatoriamente vários protocolos (incluindo protocolos orientados a conexão) para enviar dados sobre canais não orientados a conexão. Isto pode ser detectado utilizando-se firewalls que mantenham o estado das conexões (statefull-firewalls). Outro ponto importante é que estes pacotes costumam destinar-se a portas acima de 1024.

6. Contra-medidas

Até o momento, não existe uma solução definitiva contra os ataques de denial-of-service e ataques distribuídos. Algumas pesquisas estão sendo realizadas propondo soluções para peças do problema, como:

- Identificar a origem dos pacotes forjados [BEL 00].
- Inibir os amplificadores de ataques [CERT 98].
- Overlay networks [KER 02].
- Active Networks [STE 02].

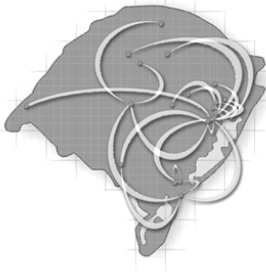
7. Conclusões

Cabe a cada administrador controlar os recursos de sua rede, observando de forma contínua os comportamentos considerados suspeitos, enumerados anteriormente. Como foi visto, boa parte das tentativas de DOS e DDOS baseiam-se na exploração de vulnerabilidades de onde origina o ataque e pela falta de monitoração e aplicação de ações rápidas por parte de quem sofre o ataque. Uma vez que ambos os lados façam o seu “dever de casa”, muitos ataques poderão ser evitados e/ou rapidamente minimizados.

7. Referências

- [BEL 00] S. Bellovin, “ICMP traceback Messages” Internet Draft, <http://www.research.att.com/~smb/papers/draft-bellovin-itrace-00.txt> Acesso em Agosto/2003.
- [CAI 02] Analysis of the Sapphire Worm <http://www.caida.org/analysis/security/sapphire/> Acesso em Agosto/2003.

- [CERT 95] CERT® Advisory CA-1995-01 IP Spoofing Attacks and Hijacked Terminal Connections. <http://www.cert.org/advisories/CA-1995-01.html> Acesso em Agosto/2003.
- [CERT 96] CERT Advisory CA-1996-26 Denial-of-Service Attack via ping <http://www.cert.org/advisories/CA-1996-26.html> Acesso em Agosto/2003.
- [CERT 96b] CERT Advisory CA-1996-21 TCP SYN Flooding and IP Spoofing Attacks. <http://www.cert.org/advisories/CA-1996-21.html> Acesso em Agosto/2003.
- [CERT 96c] CERT Advisory CA-1996-01 UDP Port Denial-of-Service Attack. <http://www.cert.org/advisories/CA-1996-01.html> Acesso em Agosto/2003.
- [CERT 97] Denial of Service Attacks. CERT Coordination Center . . Outubro/1997 - Initial Release. http://www.cert.org/tech_tips/denial_of_service.html Acesso em Agosto/2003.
- [CERT 98] CERT Advisory CA-1998-01 Smurf IP Denial-of-Service Attacks. <http://www.cert.org/advisories/CA-1998-01.html> Acesso em Agosto/2003.
- [CERT 99] Results of the Distributed-Systems Intruder Tools Workshop. Pittsburgh, Pennsylvania USA, November 2-4, 1999.
- [CERT 99b] CERT® Advisory CA-1999-17 Denial-of-Service Tools. <http://www.cert.org/advisories/CA-1999-17.html> Acesso em Agosto/2003.
- [CERT 01] CERT Advisory CA-2001-20 Continuing Threats to Home Users. <http://www.cert.org/advisories/CA-2001-20.html> Acesso em Agosto/2003.
- [CERT 03] Advisory CA-2003-08 Increased Activity Targeting Windows Shares. <http://www.cert.org/advisories/CA-2003-08.html>. Acesso em Agosto/2003.
- [CERT 03b] Vulnerability Note VU#484891 - Microsoft SQL Server 2000 contains stack buffer overflow in SQL Server Resolution Service. <http://www.kb.cert.org/vuls/id/484891> Acesso em Agosto/2003.
- [KAS 02] Kashiwa, Dai; Chen, Eric; Fuji, Hitoshi. Active Shaping: A countermeasure against DDos Attacks. Universal Multiservice Networks, 2002. ECUMN 2002. 2nd European Conference on. IEEE.
- [KER 02] Keromytis, Angelos; et al; SOS: Secure Overlay Services. ACM SIGCOMM, 19-23 Agosto 2002.
- [NEU 00] Peter G. Neumann. Inside Denial-of-Service Attacks. Communications of the ACM; April 2000/Vol. 43, No. 4.
- [NOR 99] NORTH CUTT, S. 1999. Network Intrusion Detection: An Analyst's Handbook. New Riders.
- [STE 02] Sterne Dan; et al; Active Network Based DDos Defense. DARPA Active Networks Conference and Exposition 2002. IEEE Computer Society.



Sessão Técnica 4

Rede Peer-to-Peer, Qualidade de Serviço e Roteamento em Redes IP

Localização de Conteúdo em Redes Peer-to-Peer

André Detsch

¹ PIPCA - Programa de Pós-Graduação em Computação Aplicada
Centro de Ciências Exatas e Tecnológicas - UNISINOS

Resumo. *Com a rápida popularização de aplicações como Napster e Gnutella, redes peer-to-peer têm sido o foco de atenção de muitos projetos de pesquisa. Um dos desafios mais relevantes nesta área é a forma com que é realizada a busca por conteúdo dentro de uma rede destas dimensões e com esta dinamicidade. Neste artigo, são apresentadas as principais técnicas criadas com esta finalidade, bem como uma visão geral do processo e das métricas de avaliação de sistemas peer-to-peer.*

1. Introdução

A Internet, como inicialmente concebida no final dos anos 60, era um sistema peer-to-peer (P2P), ou seja, os nós que faziam parte da rede tinham funções similares entre si. Nos anos 90, com o avanço da Internet, esta forma de comunicação passou a dar lugar a uma estrutura principalmente cliente-servidor, onde os usuários finais passaram a agir quase que exclusivamente como consumidores de informação ([1]). Esta realidade passou a mudar quando aplicações P2P como Napster ([2]) e Gnutella ([3]) passaram a possibilitar que usuários comuns, mesmo desprovidos de uma conexão estável ou de um endereço IP válido, contribuíssem para a disseminação de dados através da Internet. De certa forma, algumas idéias da Internet original passaram a ganhar força, mas em um ambiente já bastante adaptado ao modelo vigente. Dentro dessa nova gama de aplicações, surgem diversas questões de pesquisa que precisam serem investigadas.

Um dos principais desafios para a franca utilização de recursos P2P é a localização de conteúdo dentro da rede. Enquanto no modelo cliente-servidor a informação pode ser encontrada diretamente em um servidor projetado para ficar no ar 24 horas por dia, em uma rede P2P os dados estão distribuídos em milhares de nós que, muitas vezes, se conectam à rede por curtos períodos de tempo.

Este artigo faz uma análise do estado da arte na área de P2P, descrevendo brevemente as principais tecnologias e apontando desafios pendentes. Assim, a Seção 2. apresenta os principais sistemas desenvolvidos até o momento, enquanto a Seção 3. explica as técnicas utilizadas na avaliação de sistemas deste tipo. A Seção 4. finaliza o artigo, tecendo observações gerais e apontando alguns horizontes na área.

2. Mecanismos de Busca em Redes P2P

Conforme citado acima, a localização dos dados é um dos principais campos de pesquisa em redes P2P. O principal precursor dos sistemas puramente P2P estudados até hoje é o Napster ([2]). Apesar de ele ser considerado por muitos um sistema P2P, já que os dados ficam distribuídos nos diversos nós que compõem a rede, o seu mecanismo de busca não segue esta política, estando baseado em um servidor que centraliza os índices. Ou seja, apesar dos dados estarem distribuídos através dos integrantes da rede P2P, a localização pode ser realizada fazendo-se uma consulta direta ao servidor central.

Embora tenha como grande vantagem sua simplicidade, esta abordagem é inadequada pela sua falta de escalabilidade e pela presença de um ponto central de falha. Protocolos posteriores,

como o Gnutella ([3]), surgiram como uma alternativa a este modelo centralizado de busca. Nestes protocolos não existe uma entidade central responsável por indexar os dados da rede: todos os nós, a princípio, acumulam as mesmas funções de servir arquivos, efetuar buscas e auxiliar no roteamento das mensagens envolvidas no processo. As redes P2P descentralizadas podem ser divididas em duas categorias principais: aquelas que não estabelecem uma estruturação forte dos dados contidos nos nós componentes, e aquelas que buscam realizar esta estruturação para aumentar a eficiência do processo de busca, conforme descrito a seguir.

Redes P2P não estruturadas

Neste tipo de rede, cujo principal representante é o Gnutella, quando um novo dado é inserido, o nó envolvido se resume a armazenar o arquivo e indexá-lo localmente. Para localizar um dado, utiliza-se “inundação”: em termos gerais, uma mensagem de busca é repassada por cada nó a todos os seus vizinhos. Conceitualmente, é realizado um *Breadth-First Search* (BFS) na rede em busca dos arquivos desejados ([4]). O número de níveis pela qual a mensagem passa nesta busca é limitada por um atributo TTL (*time to live*), com semântica análoga à utilizada no IP, mas em nível de aplicação.

Este tipo de rede é comumente utilizada na vida real em função da sua simplicidade e não necessidade de armazenamento de informações complexas nos nós ([5]). Quando um nó se desconecta, a rede deixa de contar com os dados que o nó compartilhava, mas nenhuma outra atualização é necessária para manter a consistência da rede. Desta forma, mudanças frequentes na topologia da rede têm pouco impacto no desempenho. Além disso, estes sistemas são flexíveis: podem operar sobre qualquer estrutura P2P construída, o que permite a utilização de redes ótimas segundo algum critério arbitrário, como a aproximação da configuração da rede em nível de aplicação (a rede P2P) com a rede real sobre a qual ela está baseada.

A principal desvantagem de redes P2P sem estruturação de dados é o volume de tráfego gerado durante o processo de busca. Como forma de aperfeiçoar o sistema, em especial quanto ao tráfego gerado, a literatura recente registra uma série de propostas baseadas nas características da Internet e de seu uso.

A exploração da heterogeneidade da rede é uma forma de melhorar o funcionamento do sistema como um todo ([6]). Nós com maior capacidade de processamento e banda disponível devem desempenhar papéis mais importantes dentro da rede, como p.ex. na rede FastTrack ([7]). Nesta abordagem, um conjunto de *SuperNodes* (com recursos de processamento e banda abundantes) forma o núcleo da rede, enquanto os demais nós se conectam a um destes *SuperNodes*. Cada *SuperNode* indexa, localmente, os arquivos compartilhados por todos os nós que se conectam a ele. Desta forma, o roteamento de buscas é realizado apenas dentro do núcleo da rede, reduzindo bastante o consumo da banda naqueles nós com menos recursos. Sua desvantagem é o tráfego gerado durante a transmissão dos índices para o *SuperNode*. Esta idéia foi adaptada para o Gnutella, com o nome de *Ultrapeers* ([8]).

Substituir a inundação por formas menos agressivas de distribuição de mensagens de busca (*queries*) também pode trazer bons resultados, como p.ex. *Iterative Deepening* e *Directed BFS* ([4]). Com *Iterative Deepening*, os nós realizam a inundação de maneira gradual, variando o valor de TTL. Em virtude da forma de distribuição dos dados da rede, itens populares tendem a ser encontrados sem a necessidade de um TTL alto. Caso o dado não seja encontrado, o valor de TTL é incrementado e a busca recomeça. *Directed BFS* utiliza heurísticas para repassar as buscas apenas para os vizinhos com maior chance (baseado no desempenho anterior) de levar à localização do dado. Outra técnica eficaz é o *Random Walk* ([6]), onde a consulta assume o papel de um “*walker*” que caminha de nó em nó (sem uso de *broadcast*) em busca do dado. Para aumentar a eficiência do mecanismo, são disparados vários “*walkers*” paralelos a cada busca realizada.

Conforme citado acima, redes não estruturadas podem trabalhar sobre qualquer estrutura P2P

construída. Vários projetos de pesquisa exploraram esta capacidade para melhorar o desempenho do sistema. Em [9], é explorado o interesse comum por conteúdo: os nós tendem a se aproximar e estabelecer conexões diretas com aqueles que mais freqüentemente possuem o conteúdo procurado. Em [10] é discutido um algoritmo para a minimização do diâmetro da rede. Basicamente, algoritmo trabalha na etapa de inserção de novos nós na rede: ao invés do nó se conectar a algum membro aleatório, é escolhido um membro que respeite certas propriedades, de forma que o diâmetro não seja aumentado desnecessariamente. Existem também técnicas que tentam aproximar a topologia em nível de aplicação (formada pelas conexões entre vizinhos) à topologia real (formada pelos enlaces entre roteadores e hosts), como em [11].

O *Yappers* ([5]) utiliza alguns conceitos de Tabela Hash Distribuída (técnica apresentada a seguir) de forma a criar “categorias” de nós e dados. Nós de uma classe X são responsáveis por indexar dados da classe X, nós da classe Y são responsáveis por indexar dados da classe Y, e assim por diante, para um número determinado de classes. Com esta categorização, é possível realizar o roteamento das buscas por um dado de uma determinada classe apenas entre os nós daquela classe, direcionando o processo de pesquisa de forma a melhorar a eficiência.

Redes P2P estruturadas

Uma das principais fontes de ineficiência na busca em redes não estruturadas é a falta de informações sobre a distribuição dos dados na rede. Para que haja um controle maior neste sentido e, conseqüentemente, seja possível realizar buscas mais eficientes na rede, vários grupos de pesquisa elaboraram projetos de redes estruturadas. A estruturação dos dados propriamente dita se dá através da utilização de Tabela Hash Distribuída (*Distributed Hash Table - DHT*)([12]).

DHT tem como base a utilização de uma chave que identifica o dado sendo compartilhado. No caso de compartilhamento de arquivos, esta chave pode corresponder à codificação do próprio nome do arquivo. Aos nós também é atribuída uma chave neste mesmo espaço de chaves (ou seja, com o mesmo número de dígitos). Cada nó da rede fica “responsável” por um intervalo de chaves de dados, ou seja, armazena a informação de qual máquina possui o dado para cada uma das chaves do seu intervalo. Em todos os algoritmos, existe uma função de proximidade entre a chave sendo procurada e a chave dos nós. A partir desta função, é possível encaminhar (rotear), de acordo com as informações armazenadas no nó, a busca para o vizinho “mais próximo” do dado sendo procurado. A função de proximidade, as informações armazenadas em cada nó e a forma de roteamento resultante são características de cada algoritmo.

De forma geral, a principal vantagem deste tipo de rede é a escalabilidade e o menor tráfego gerado a cada busca realizada. Outra é que mesmo itens raros podem ser facilmente encontrados. Em compensação, a complexidade do protocolo aumenta, e é preciso constantemente manter a consistência geral da estrutura mantida para roteamento das buscas: a cada inserção ou remoção de nós/dados, deve-se atualizar as informações mantidas por determinados nós.

Em [13], foi apresentado o primeiro algoritmo que poderia ser utilizado de forma escalável com DHT. Entretanto, um ambiente estático era assumido, fazendo com que este mecanismo não pudesse ser diretamente aplicado em redes P2P. Em 2001, apareceram trabalhos importantes sobre a utilização de DHT em ambientes dinâmicos, centrados em redes P2P ([12], [14], [15], [16]). A seguir, dois destes trabalhos são brevemente explicados.

O Chord ([15]) utiliza um espaço de chaves organizado em um anel unidimensional. A proximidade entre chaves é dada pela proximidade do identificador do nó em relação ao dado dentro deste anel. Em cada nó, são mantidos dois conjuntos de vizinhos. A *successor list* é mantida para garantir a eficácia do roteamento, e armazena a identificação de k hosts que seguem o nó no espaço de chaves. A eficiência do roteamento é alcançada através da *finger list*, de $O(\log n)$ nós espaçados exponencialmente no espaço de chaves. O número de *hops* por busca é da ordem de $O(\log n)$. O

CAN ([16]) distribui as chaves em um espaço d -dimensional, cabendo a cada nó uma subregião. O processo de busca se dá através do encaminhamento da consulta para o nó mais próximo do nó objetivo. O número de vizinhos mantidos pelos nós é $O(d)$, e o número de *hops* envolvidos na busca é $O(dn^{1/2})$.

Assim como no caso de redes não estruturadas, a aproximação da topologia P2P (aplicação) com a topologia real pode trazer benefícios consideráveis. Entretanto, esta adaptação é mais complexa, posto que cada nó é responsável por uma região dos dados e deve, necessariamente, ter como vizinhos nós responsáveis pelas regiões “próximas” (segundo o critério específico do protocolo). O *Brocade* ([17]) atua com este objetivo, visando minimizar o tráfego de mensagens entre *Stub Domains*. Para tanto, é utilizado uma camada adicional de roteamento, estabelecida entre alguns “*supernodes*”. Sempre que um *supernode* intercepta uma mensagem de busca, ele identifica o domínio (de rede) onde o dado poderá ser localizado, e realiza o encaminhamento direto para um *supernode* daquele domínio. Com isso, pode-se reduzir consideravelmente o custo de rede e a latência envolvida no processo de busca.

Uma limitação na utilização direta de DHT é que cada dado é indexado por apenas uma chave; por exemplo, para que seja localizado um dado, deve-se conhecer o nome exato do arquivo buscado. Uma forma simples de se adaptar DHT para possibilitar a consulta por partes do nome do arquivo seria indexar cada “*n-grama*” (*substrings* distintas, de tamanho n , presentes em uma *string*) do nome do arquivo. Por exemplo, para a *string* “GNU Linux”, os n -gramas, para um n igual a três, são: “GNU”, “NU”, “U L”, “L”, “Lin”, “inu” e “nux”. Ao se efetuar uma busca, a *string* seria dividida também nos seus n -gramas e seria realizada uma busca para cada um deles. A partir da consolidação dos resultados obtidos, seria possível identificar os arquivos de interesse. Ainda não existem estudos aprofundados para a avaliação do comportamento desta técnica ([18]).

Ainda sobre DHT, existe um grande foco de estudo na possibilidade de utilização de semântica de banco de dados relacional sobre a rede P2P. Isto permitiria buscas bem mais avançadas, abrindo um vasto leque de novas aplicações para estes sistemas. Tais estudos ([18]) ainda estão em fase inicial, e futuras descobertas poderão levar a grandes avanços na área de sistemas P2P.

A *Freenet* ([19]) pode ser definida como uma rede estruturada, embora fracamente. Na *Freenet*, também existe o uso de chaves *hash* para a localização de dados, porém os dados são replicados e posicionados de acordo com o volume de buscas, sem que algum nó seja responsável por algum intervalo de dados.

3. Avaliação de Sistemas P2P

São várias as métricas utilizadas para avaliar a qualidade dos sistemas P2P. A seguir, as principais métricas são brevemente descritas.

Volume de tráfego gerado pela indexação/busca de ítems. Em um sistema estável, a minimização do custo de se efetuar uma busca é muito mais importante do que a minimização do custo de indexação. Entretanto, em sistemas dinâmicos como redes P2P, o custo e a complexidade da indexação de novo ítems deve ser cuidadosamente considerada.

Latência de busca. O tempo entre início da pesquisa e recebimento dos resultados.

Sobrecarga de nós. Os mecanismos devem visar distribuir a carga entre os nós da rede, idealmente considerando as capacidades, mas sempre evitando a super-utilização de algum nó específico. Este problema pode ocorrer, por exemplo, em uma rede estruturada quando um nó fica responsável por uma região muito popular de dados.

Quantidade de informações armazenadas em cada nó. Muitas técnicas armazenam uma quantidade maior de informações (como índices dos nós vizinhos) para reduzir o tráfego gerado

pelas buscas, conforme visto na seção anterior. O número de vizinhos que o nó mantém também se encaixa nesta categoria. De forma geral, é desejável minimizar a quantidade de informações que cada nó deve armazenar.

Eficácia na localização de ítems raros e eficiência na busca de ítems populares. A distribuição dos dados em redes P2P tipicamente obedece à lei de Zipf ([20]). Esta característica pode ser utilizada para tornar mais eficiente a busca por ítems populares, bastante replicados. Por outro lado, é desejável que, uma vez presente na rede, qualquer dado possa ser localizado.

Complexidade de inserção/remoção de nós. Em especial no caso de redes estruturadas, o impacto resultante da entrada ou saída (abrupta ou não) de um nó de rede pode ser importante e até inviabilizar o uso em ambientes muito dinâmicos (como é tipicamente o caso de sistemas P2P).

Robustez. De forma análoga ao ítem anterior, o impacto de um mau funcionamento de um nó (seja acidental ou intencional) não deve inviabilizar o uso do sistema.

O processo de avaliação de sistemas P2P apresenta dificuldades, incluindo sua escala e padrões reais de uso. O problema de escalabilidade, bastante comum em simulações de rede, se agrava para o caso de redes P2P, onde o sistema em desenvolvimento tem o objetivo de atender a vários milhares de nós simultaneamente. Em virtude disto, avaliações analíticas são importantes, e estão presentes em vários estudos desta área. Exemplos de modelos analíticos que visam modelar os diferentes sistemas P2P são [21] e [22].

Apesar de simuladores de rede de uso geral serem utilizados para alguns experimentos, a simulação de P2P em larga escala ainda se restringe a simuladores específicos ([16], [15]), desenvolvidos em função do próprio projeto de pesquisa. Em [16], autores argumentam que a escala atingida ultrapassa 260.000 nós.

Quando se trata de simulação ou análise de redes P2P, um dos fatores vitais para a validade das conclusões é a utilização de padrões de uso reais, ou seja, gerar eventos (buscas, inserção e remoção de nós, etc.) e criar um ambiente (distribuição e tamanho dos arquivos) que sejam condizentes com a realidade. Neste sentido, existem vários estudos que objetivam realizar esta modelagem de maneira fiel (p.ex., [23] e [20]). Em [23], obteve-se um modelo que representa uma hora de funcionamento da rede Gnutella, no momento da pesquisa, com 400.000 nós. É bastante comum a utilização de alguns rastros de tráfego Web ([24]), que gerariam padrões de utilização compatíveis com redes P2P.

4. Conclusões

O estudo de redes P2P, como explorados atualmente, é uma área nova e em franca expansão. Diferente do que ocorre com a maioria das tecnologias, o estudo de redes P2P começou após a sua utilização em massa na Internet. Com isso, surgiu uma grande lacuna em termos de pesquisa, sendo a mesma preenchida gradativamente. Este volume de pesquisa ajuda a compreender as capacidades de sistemas P2P, mas ainda é difícil prever que tipo de aplicações resultarão deste esforço a médio prazo. De qualquer forma, a introdução deste tipo de tecnologia já caracteriza uma nova era na Internet, onde todos podem participar na construção do conteúdo disponível na rede.

De forma geral, podemos identificar vários pontos de estudo a serem tratados. A interação entre as otimizações propostas para os protocolos muitas vezes ainda é desconhecida. Também não se sabe até que ponto é possível aproveitar os benefícios das redes não estruturadas e os benefícios das redes estruturadas em um só sistema. A utilização de DHT promete ser cada vez mais difundida, mas a sua complexidade e relativa falta de maturidade nas implementações ainda limita o seu uso em grande escala. Face à revisão crítica e abrangente realizada sobre a literatura, cujos resultados são objeto deste artigo, percebe-se que muito será realizado nos próximos anos e que há grande potencial para desenvolvimento de novas idéias nesta área.

Bibliografia

- [1] Andy Oram. *Peer-to-peer: Harnessing the Power of Disruptive Technologies*. O'Reilly, March 2001.
- [2] Napster, August 2003. <http://www.napster.com>.
- [3] Gnutella, August 2003. <http://gnutella.wego.com>.
- [4] Beverly Yang and Hector Garcia-Molina. Efficient Search in Peer-to-Peer Networks. Technical report, Computer Science Department, Stanford University, 2002.
- [5] Prasanna Ganesan, Qixiang Sun, and Hector Garcia-Molina. YAPPPERS: a peer-to-peer lookup service over arbitrary topology. In *INFOCOM 2003*, March 2003.
- [6] Qin Lv, Sylvia Ratnasamy, and Scott Shenker. Can Heterogeneity make Gnutella Scalable? In *IPTPS 2002*, March 2002.
- [7] Fasttrack, August 2003. <http://www.fasttrack.nu>.
- [8] Anurag Singla and Christopher Rohrs. Ultrapeters: Another Step Towards Gnutella Scalability, December 2001. <http://www.bearguru.com/kb/articles/Ultrapeters.html>.
- [9] Kunwadee Sripanidkulchai, Bruce Maggs, and Hui Zhang. Efficient content location using interest-based locality in peer-to-peer systems. In *INFOCOM 2003*, March 2003.
- [10] G. Pandurangan, P. Raghavan, and E. Upfal. Building Low-Diameter Peer-to-Peer Networks. *IEEE Journal on Selected Areas in Communications (JSAC)*, 21(6), August 2003.
- [11] T.S. Eugene Ng, Yang-hua Chu, Sanjay Rao, Kunwadee Sripanidkulchai, and HuiZhang. Measurement-based optimization techniques for bandwidth-demanding peer-to-peer systems. In *INFOCOM 2003*, March 2003.
- [12] Peter Druschel and Antony Rowstron. Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems. In *18th IFTP/ACM International Conference on Distributed Systems Platforms (Middleware 2001)W*, November 2001.
- [13] C. Plaxtron, R. Rajaraman, and A. Richa. Accessing nearby copies of replicated objects in a distributed environment. In *ACM SPAA 1997*, June 1997.
- [14] B. Y. Zhao, J. Kubiatowicz, and A. Joseph. Tapestry: An infrastructure for fault-tolerant wide-area location and routing. Technical report, University of California at Berkeley, 2001.
- [15] Ion Stoica, Robert Morris, David Karger, Frans Kaashoek, and Hari Balakrishnan. Chord: A scalable Peer-To-Peer lookup service for internet applications. In *SIGCOM 2001*, 2001.
- [16] Sylvia Ratnasamy, Paul Francis, Mark Handley, Richard Karp, and Scott Shenker. A Scalable Content-Addressable Network. In *SIGCOM 2001*, 2001.
- [17] Ben Y. Zhao, Yitao Duan, Ling Huang, Anthony D. Joseph, and John D. Kubiarowicz. Brocade: Landmark Routing on Overlay Networks. In *IPTPS 2002*, March 2002.
- [18] Matthew Harren, Joseph M. Hellerstein, Ryan Huebsch, B. T. Loo, S. Shenker, and I. Stoica. Complex Queries in DHT-based Peer-to-Peer Networks. In *IPTPS 2002*, March 2002.
- [19] Freenet, August 2003. <http://freenet.sourceforge.net>.
- [20] Juliano Santos, Leonardo Rocha, Diêgo Nogueira, Paulo Araújo, Virgílio Almeida, and Wagner Meira Júnior. Caracterização de carga de redes Peer-to-Peer. In *SBRC 2002*, May 2002.
- [21] Hung-Chang Hsiao and Ghung-Ta King. Modeling and Evaluating Peer-to-Peer Storage Architecture. In *IPDPS 2002*, 2002.
- [22] Zihui Ge, Daniel Figueiredo, Sharad Jaiswal, James F. Kurose, and Don Towsley. Modeling peer-peer file sharing systems. In *INFOCOM 2003*, March 2003.
- [23] M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system design. *IEEE ICJ*, 2002.
- [24] W3C Web Characterization Repository, August 2003. <http://repository.cs.vt.edu>.

A Tecnologia Peer-to-Peer como Ferramenta para Comunicação em Redes Ad Hoc Móveis

Felipe Jung Vilanova, Juergen Rochol,
Maria Janilce Bosquiroli Almeida, Lisandro Zambenedetti Granville

Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre, RS – Brasil

{fjvilanova, juergen, janilce, granville}@inf.ufrgs.br

Abstract. *This paper introduces a discussion about the employment of peer-to-peer technology as a tool for communication in mobile ad hoc networks. It describes both technologies and their characteristics, and draw a comparison between them, focusing on the similarities, but introducing some differences too. Some tools that can be used for the integration of that technologies are introduced as well.*

Resumo. *Este artigo apresenta uma discussão sobre a utilização de tecnologias peer-to-peer como ferramenta de comunicação em redes ad hoc móveis. O artigo descreve as duas tecnologias e suas características, e estabelece uma comparação entre elas, enfatizando suas semelhanças, mas também apresentando algumas diferenças. Também são apresentadas algumas ferramentas que podem ser utilizadas para a integração dessas duas tecnologias.*

1. Introdução

Avanços no desenvolvimento das tecnologias de redes sem fio e de computação móvel tornaram muito comum o uso de dispositivos móveis. As tecnologias de redes sem fio permitem que as pessoas se comuniquem sem a necessidade de uma infra-estrutura de rede pré-existente, e a computação móvel permite que as pessoas com dispositivos móveis acessem recursos e serviços enquanto se deslocam.

Outra tecnologia, que também está se tornando bastante popular, são as redes peer-to-peer (P2P), que apresentam a mesma natureza *ad hoc* das redes móveis sem fio. Por isso, é interessante o estudo da aplicação destas redes para o paradigma da computação móvel.

Redes P2P permitem que os dispositivos móveis utilizem os benefícios da descentralização total e compartilhem recursos eficientemente. Segundo Liu (2002), este tipo de rede é uma boa solução para sistemas de computação móvel trabalhando em um ambiente de rede *ad hoc* sem fio onde são necessários a descoberta de recursos e o roteamento dinâmico de informações através da rede.

Traçar um paralelo entre essas tecnologias pode ser útil na identificação de características comuns a estas redes e talvez até possibilite o intercâmbio de soluções que possam vir a ser aplicáveis em ambos os casos.

Este artigo tem por objetivo discutir a utilização das redes P2P para estabelecer comunicação em redes *ad hoc* móveis (MANETs – *Mobile Ad Hoc Networks*), e apresentar alguns exemplos de aplicações P2P com este propósito. O artigo está

organizado da seguinte maneira. Na seção 2 é apresentado o conceito e as características das MANETs. Na seção 3 são apresentados os conceitos e características das redes P2P. A seção 4 discute a relação entre essas duas tecnologias, e também apresenta as semelhanças e diferenças entre as mesmas. Na seção 5 são apresentadas algumas ferramentas P2P existentes, e que podem ser utilizadas em MANETs.

2. Redes Ad Hoc Móveis (MANET)

Uma rede *ad hoc* móvel (MANET) consiste em um conjunto de dispositivos, equipados com transmissores sem fio, que podem se deslocar arbitrariamente. Corson e Macker (1999) definem esse tipo de rede como um sistema descentralizado, autônomo e sem estrutura alguma pré-existente, onde todos os dispositivos funcionam como roteadores, e cooperam dinamicamente para estabelecer comunicação.

2.1. Características das MANETs

Uma das principais características de uma MANET é a natureza dinâmica da sua topologia. Como os dispositivos são livres para se deslocarem, a topologia pode mudar aleatoriamente, em intervalos de tempos indeterminados. A vantagem desse sistema é a eliminação da necessidade de uma infra-estrutura pré-existente, e de pontos de falha isolados. Porém, a ausência de infra-estrutura torna essas redes mais vulneráveis, menos seguras e confiáveis, e mais difíceis de gerenciar.

Todos os nodos carregam as mesmas responsabilidades e como essas redes não possuem uma estrutura física determinada, os nodos se organizam espontaneamente e têm que lidar frequentemente com mudanças de localização e topologia, além de operações frequentes de conexão e desconexão.

O roteamento é o principal problema dessas redes. Como não existe um roteador padrão na rede, cada nodo deve agir como roteador e distribuir as mensagens pela rede. Em redes com muitos nodos, um percentual significativo de banda será ocupada com a disseminação das mensagens de controle. Também deve ser levado em conta o limite da área de transmissão de cada nodo. Se uma conexão que utiliza um nodo intermediário falha durante uma transmissão, as mensagens que estão sendo transmitidas não podem chegar ao destino, e nem o transmissor dessas mensagens será informado que elas não chegaram.

Outra característica é a limitação de banda e a baixa confiabilidade. Os nodos se comunicam por canais sem fio, que comparados aos cabos, têm pouca largura de banda e são facilmente influenciados por fatores do meio ambiente como ruído e barreiras físicas. Os nodos também sofrem com as freqüentes mudanças de propagação das ondas eletromagnéticas e com as limitações dos dispositivos móveis, que têm baixo alcance, menos recursos e menor capacidade de processamento [Liu 2002],[Corson e Macker 1999].

3. Peer-to-Peer

O conceito de redes P2P, popularizado por sistemas de troca de arquivos, não é novo, mas atualmente, fatores como o aumento da quantidade de aplicações que usam a Internet e a necessidade de largura de banda, poder computacional e de armazenamento, impulsionaram o crescimento dessa tecnologia.

Redes P2P são sistemas distribuídos sem qualquer controle centralizado ou organização hierárquica, em que os nodos executam funções equivalentes. Essas redes podem ser caracterizadas em uma classe de sistemas e aplicações que utilizam recursos distribuídos para executar funções críticas de um modo descentralizado. Esses recursos e serviços incluem a troca de informação, de ciclos de processamento e de espaço de armazenamento. Em uma rede P2P, computadores que tradicionalmente têm sido usados somente como clientes comunicam-se diretamente entre si, podendo atuar como clientes e servidores, assumindo o papel que for mais eficiente para a rede. Alguns dos benefícios das redes P2P incluem: aumento da escalabilidade, pela eliminação da dependência de pontos centralizados; eliminação da necessidade de uma infra-estrutura cara, devido à comunicação direta entre nodos; e a possibilidade de agregar recursos.

3.1. Redes P2P puras e híbridas

Uma das idéias da descentralização é que os usuários mantenham e controlem os dados e recursos, mas isto dificulta a implementação de modelos P2P, pois não há um servidor central com uma noção geral de todos os dispositivos da rede e os dados que eles disponibilizam. Essas informações têm que ser conhecidas por todos os nodos, e quando um nodo entra na rede, deve executar mecanismos para descobrir os outros dispositivos, e guardar informações sobre seus endereços. Isso cria problemas de escalabilidade e de gerenciamento.

Por essa razão, muitas redes P2P são implementadas de uma maneira híbrida, como por exemplo o Napster (www.napster.com), onde há um diretório central para os arquivos, mas os nodos fazem o *download* desse arquivos diretamente dos dispositivos a que eles pertencem [Milojicic, Kalogeraki e Lokose 2002].

3.2. Característica das redes P2P

As redes puras são auto-organizáveis e não necessitam de uma entidade central para o gerenciamento. Uma consequência direta dessa descentralização é o alto grau de escalabilidade dessas redes.

De acordo com Milojicic, Kalogeraki e Lokose (2002), o anonimato é uma das características mais desejáveis. Ele permite que as pessoas não se preocupem com as consequências legais e que não seja possível a censura das informações digitais.

A principal vantagem dessas redes é que elas distribuem as responsabilidades de fornecer serviços entre todos os seus nodos. Isso elimina uma alta carga de serviços em um dispositivo, e pontos isolados de falha, e fornece uma solução escalável e de baixo custo para a prestação de serviços. Mas a desvantagem é que redes P2P são difíceis de controlar devido ao alto grau de distribuição e anonimato [Liu 2002].

4. Utilização da tecnologia P2P em MANETs

Tanto as MANETs como redes P2P têm como característica básica a ausência de uma infra-estrutura fixa, e o fato de os próprios nodos dessas redes serem responsáveis pelo seu gerenciamento. Devido a essa natureza comum, torna-se atrativo o emprego de redes P2P para estabelecer e gerenciar a comunicação entre os dispositivos de uma MANET. Para Lindemann, Klemm e Waldhorst (2003), o modelo de uma rede P2P encaixa-se perfeitamente em uma ambiente *ad hoc* sem fio onde é necessário descobrir recursos, fazer o roteamento dinâmico, disseminar as informações através da rede, e compartilhar os recursos.

Existem muitas similaridades entre as duas tecnologias, principalmente no que diz respeito ao problema básico: como estabelecer a comunicação entre os dispositivos, mas também existem algumas diferenças, devido aos diferentes níveis aos quais estão relacionadas e às diferentes motivações pelas quais foram criadas.

4.1. Similaridades entre MANETs e P2P

A base de ambos é o conceito de auto-organização. Na maioria dos casos, exceto em redes P2P híbridas, não há entidades centrais para gerenciar e coordenar a rede nem uma estrutura de rede pré-definida. A rede é estabelecida assim que os participantes decidem estabelecer conexões entre si.

A troca freqüente da topologia, resultante das permanentes mudanças das conexões é outra semelhança dessas tecnologias. Mesmo que os nodos permaneçam os mesmos, as conexões entre eles são permanentemente alteradas. Em redes sem fio isto é causado pela mobilidade dos nodos, que podem sair da área de transmissão do nodo ao qual estava conectado e estabelecer novas conexões com outros nodos. Em redes P2P, uma situação que pode ser comparada à mobilidade geográfica dos dispositivos sem fio é a freqüente variação dos grupos de nodos que estabelecem conexão em determinados momentos. Isso dificulta um diagnóstico da situação corrente da rede, é quase impossível manter um conjunto de informações atualizadas sobre toda a rede.

Em ambas as redes não há uma hierarquia padrão. Hierarquias só podem ser introduzidas virtualmente, com a utilização de protocolos, ou utilizando dispositivos para gerenciar os outros nodos. Mas como a topologia dessas redes muda com freqüência, as tabelas de roteamento de uma arquitetura hierárquica logo ficariam desatualizadas, e manter essas informações atualizadas seria difícil e oneroso.

Uma outra característica, ou problema, comum a essas redes é como um novo participante pode conectar-se a uma rede P2P ou a uma MANET já existente. Como geralmente não há uma entidade de gerenciamento central, é necessário que o novo nodo encontre os membros ativos nessas redes.

Devido às características de auto-organização e independência de cada nodo, é difícil para essas redes oferecer funcionalidades de gerenciamento, como autorização, autenticação, e, até mesmo, qualidade de serviço [Schollmeier, Gruber e Finkenzeller 2002].

4.2. Diferenças entre MANETs e P2P

Em uma MANET, a descoberta de rota e o estabelecimento das conexões precisam da ajuda de nodos intermediários: as conexões são indiretas, enquanto em redes P2P as conexões são diretas. O enlace direto de uma conexão P2P é muito mais fácil de manter que um enlace indireto, com múltiplos intermediários, de uma MANET. Devido à mobilidade dos nodos intermediários, o número de re-estabelecimentos de rota aumenta.

A tecnologia P2P baseia-se na infra-estrutura de roteamento IP das redes a cabo para implementar o roteamento entre os nodos, e essa infra-estrutura não existe nas MANETs. Hu, Das e Pucha (2002) apontam isso como o maior desafio para a aplicação de redes P2P sobre as MANETs.

Outra diferença é a estrutura da rede. Uma rede P2P é uma camada virtual criada para conectar os dispositivos. Ela é separada e completamente diferente da camada física. Os usuários de uma rede P2P podem estar distribuídos através de todo o mundo, tornando impossível a determinação da localização física de cada nodo, e da própria rede. Já a estrutura física de uma MANET pode ser diretamente mapeada para a

estrutura lógica. Apesar de a posição física de um único dispositivo de uma MANET não poder ser determinada, devido a sua mobilidade, é possível saber a localização aproximada de toda a rede [Schollmeier, Gruber e Finkenzeller 2002].

5. Algumas redes P2P utilizadas em MANETs

As redes P2P encontradas que podem ser aplicadas em MANETs são :

Bluetooth (<http://www.bluetooth.com>): um conjunto de protocolos P2P para estabelecer conexão entre dispositivos móveis. Uma rede bluetooth (conhecida como Piconet) permite a interconexão de até oito dispositivos em um raio de 10 metros. Há um dispositivo central que armazena o número de identificação de cada nodo.

DPSR (Dynamic P2P Source Routing Protocol) : proposto por Hu, Das e Pucha (2002), é um protocolo para roteamento em MANETs. Integra funções dos protocolos de roteamento P2P que operam com identificadores para os nodos e funções dos protocolos de roteamento em MANETs, que operam com endereços IP. A idéia é adaptar o protocolo de roteamento P2P à camada de rede da MANET fazendo um mapeamento entre o endereço IP e o identificador de cada nodo

MeshNetworks (<http://www.meshnetworks.com>): ferramenta P2P para roteamento em redes móveis. Fornece mecanismos para conectar dispositivos móveis em uma rede auto-organizável e auto-gerenciável.

Optimezed Routing Independent Overlay Network, ORION [Lindemann, Klemm e Waldhorst 2003] : é um conjunto de protocolos para construção e manutenção de uma plataforma que permite o roteamento de todos os tipos de mensagens necessárias para uma rede P2P de troca de arquivos.

Passive Distributed Indexing (PDI): apresentado por Lindemann e Waldhorst (2002) um serviço de busca distribuída pra troca de arquivos em aplicações móveis baseado na tecnologia pee-to-peer. PDI define um conjunto de mensagens para a transmissão de consultas e respostas.

Proem [Kortuem 2001] é uma plataforma que fornece uma solução completa para e o desenvolvimento e emprego de redes pee-to-peer em MANETs. Tem como objetivo facilitar o desenvolvimento de aplicações para comunicação e troca de arquivos em redes móveis e garantir privacidade e integridade dos dados transmitidos.

6. Conclusão

Ambas tecnologias mostram similaridades, possuem características de descentralização e auto organização, não possuem uma infra-estrutura pré-existente, e necessitam resolver o mesmo problema: estabelecer conexão em um ambiente descentralizado, dinâmico e freqüentemente variável. Mas apesar de existirem similaridades entre essas redes, elas se baseiam em estruturas físicas completamente diferentes.

As similaridades existem, e, mesmo com a diferença entre as estruturas, o emprego de redes P2P em MANETs continua atrativo. Problemas enfrentados pelas redes móveis, como descoberta de nodos, roteamento, disseminação de dados e segurança, podem ser resolvidos por ferramentas P2P.

Referências bibliográficas

- Corson, S. e Macker, J. (1999) “Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations”, RFC 2501, Internet Engineering Task Force, Network Working Group.
- Hu, I., Das, S. e Puch, H. (2002) “Exploiting the Synergy between Peer-to-Peer and Mobile Ad Hoc Networks”, School of Electrical and Computer Engineering, Purdue University, West Lafayette.
- Klemm, A., Lindemann, C. e Waldhorst, O. (2003) “A Special-Purpose Peer-to-Peer File Sharing System for Mobile Ad Hoc Networks”.
- Kortuem, G. (2001) “Proem: A Peer-to-Peer Computing Platform for Mobile Ad-hoc Networks”, Wearable Computing Laboratory, Department of Computer Science, University of Oregon.
- Lindemann, C. e Waldhorst, O. (2002) “A Distributed Search Service for Peer-to-Peer File Sharing in Mobile Applications”, *2nd IEEE Conference on Peer-to-Peer Computing(P2P 2002)*, Linköping, Sweden, p. 71-83 .
- Liu, H. (2002) “Peer-to-Peer Applications in Ad Hoc Wireless Networks”, Dissertação de Mestrado em Ciência da Computação, Univesidade de Dublin
- Milojicic, D. S., Kalogeraki, V. e Lukose, R. (2002) “*Peer-to-peer Computing*”, HP Laboratories Palo Alto.
- Schollmeier, R., Gruber, I. e Finkenzeller, M. (2002) “Routing in Mobile Ad Hoc and Peer-to-Peer Networks. A Comparison”, International Workshop on Peer-to-Peer Computing, Pisa, Italy, May.

Arquitetura Híbridas de QoS em Redes IP

Adelmo Jerônimo Silva¹, M. A R. Dantas²

¹Departamento de Engenharia Elétrica – Laboratório de Engenharia de Redes
(UnB/ENE/LabRedes) – Universidade de Brasília
Av.L3 Norte – FT – ENE – LabRedes – Sala B1-01 – Asa Norte – CEP:70910-900 –
Brasília-DF – Brasil

²Departamento de Informática e Estatística (INE) – Universidade Federal de Santa Catarina
(UFSC)
Caixa Postal 476 – Trindade – Florianópolis – SC – 88040-900 – Brasil

adelmo.silva@g8networks.com.br, mario@inf.ufsc.br

Abstract. *In this article we tackle the enhancement of applications quality in IP networks. The IETF (Internet Engineering Task Force) has already proposed some standard which became QoS (Quality of Service) architectures. The goal of this paper is to present effects in using simultaneously two standards approaches of the IETF. Therefore, we have realized some simulations considering the DiffServ and MPLS paradigms. Our experimental results indicate that these two approaches can produce in a complementary fashion improvements for the peer-to-peer quality.*

Resumo. *Neste artigo abordamos a melhoria da qualidade das aplicações em redes IP. O IETF (Internet Engineering Task Force) já padronizou algumas propostas que se tornaram arquiteturas de QoS (Quality of Service). Nosso objetivo neste artigo é apresentar um estudo dos efeitos de se utilizar simultaneamente algumas dessas abordagens já padronizadas pelo IETF. Com este objetivo realizamos algumas simulações com redes IP habilitadas para tratamento de serviços diferenciados (DiffServ) em conjunto com o MPLS (Multiprotocol Label Switching). Nossos resultados indicam que o DiffServ e o MPLS são duas tecnologias que se complementam e em conjunto produzem melhorias interessantes na qualidade de serviço fim-a-fim.*

1. Introdução

Atualmente, as redes IP são as mais utilizadas para prover serviços. A Internet é a maior rede IP em atividade, onde estão conectados milhões de computadores em todo o mundo. Em adição, existe uma tendência de crescimento sem precedentes, tanto em número de usuários como em número de serviços oferecidos [11]. Por causa do crescimento acentuado, um dos principais problemas das redes IP vem sendo a questão do compartilhamento dos recursos [6]. Como pacotes entram na rede de forma assíncrona, dois ou mais pacotes podem entrar em um nó exatamente ao mesmo tempo e serem destinados à mesma saída. Este fato caracteriza a disputa pela capacidade de processamento do nó, uma vez que estes pacotes não serão tratados simultaneamente e alguns terão que esperar até os anteriores sejam processados. Se o tráfego exceder a capacidade de rede e processamento do nó por um tempo maior, então filas serão formadas e pacotes terão que ser bufferizados, ou até mesmo descartados.

O estágio atual de utilização da Internet, tem uma vasta gama de aplicações, todas com diferentes requisitos para seu perfeito funcionamento. Por outro lado, tem sido verificada uma evolução continua nos meios físicos de transmissão onde já é possível se atingir taxas de transmissão da ordem de Mbytes/segundo com uma latência da ordem de μ segundos. Diferentes aplicações que utilizam imagem, vídeo, voz, gráficos e textos convivem na rede, todavia cada qual deve utilizar a rede da uma melhor possível de uma maneira diferenciada.

Neste artigo, apresentamos um estudo cuja abordagem é orientada a melhoria da qualidade das aplicações em redes IP. Desta forma, realizamos algumas simulações com redes IP habilitadas para tratamento de serviços diferenciados (DiffServ) em conjunto com o MPLS (*Multiprotocol Label Switching*). Para apresentação de nosso trabalho de pesquisa o artigo foi organizado da seguinte forma: o ambiente de rede IP é discutido e alguns de seus novos esforços são introduzidos na segunda seção; na terceira seção apresentamos nossa proposta de configuração para estudo através das simulações; o modelo e os resultados experimentais são ilustrados na seção quatro; finalmente na seção cinco apresentamos nossas conclusões e algumas sugestões de trabalhos que podem considerados como continuação da presente pesquisa.

2. O Ambiente de Rede IP

Enquanto a Internet só dispunha de aplicações simples, a demanda era bem atendida pelo nível de qualidade oferecido. O crescimento acentuado da Internet nos últimos anos e o seu constante amadurecimento motivaram o surgimento de novas aplicações distribuídas, com grande necessidade de largura de banda e restrições de retardo. Assim, outras classes de serviços com diferentes prioridades e necessidade de recursos passaram a ser demandas gerando a necessidade de melhoramentos no tipo de serviço oferecido. Desta observação surgiu a idéia do paradigma conhecido como *qualidade de serviço* (QoS). Em uma rede com QoS pode-se ajustar a mesma para oferecer desempenhos específicos para diferentes classes de serviços.

Para atender a demanda por QoS na Internet, a comunidade científica está empenhada em pesquisas de desenvolvimento de novos mecanismos de suporte a aplicações avançadas. Alguns mecanismos já foram propostos e estão em fase de testes e padronização. Dentre inúmeros esforços, as iniciativas do IETF (*Internet Engineering Task Force*) são aqueles que tem apresentado um melhor conjunto de soluções para mecanismos de controle de QoS. Estas soluções são consideradas basicamente em quatro abordagens [7]: classificação do tráfego por priorização (*DiffServ*); e reserva prévia de recursos (*IntServ*); melhoramento do encaminhamento (MPLS) e melhoramento das técnicas de roteamento (QoSRR).

O objetivo do nosso trabalho é fazer um estudo sobre as abordagens DiffServ e MPLS, tentando identificar as melhores características de cada técnica. Nosso esforço visa entender se é possível uma solução única com contribuição de todas as abordagens. É importante compreender se os dois paradigmas podem contribuir para melhoria das aplicações com um serviço diferenciado fim-a-fim.

Na abordagem *DiffServ* o tráfego da rede é dividido em categorias conforme a prioridade de cada aplicação e os recursos são divididos de acordo com o critério de política de administração da rede. Os mecanismos de encaminhamento de pacotes e tratamento de filas dão tratamento preferencial a pacotes cujas aplicações são identificadas como tendo requisitos mais exigentes, ou seja, encaminham os pacotes de forma diferenciada. Por esta razão, os estes esforços são chamados de *serviços diferenciados*. O tratamento diferenciado dos pacotes se dá através do agrupamento em categorias de tráfego denominadas classes. Uma classe pode conter um único fluxo, ou a agregação de múltiplas instâncias de fluxo. Tratamento diferenciado nos pacotes pode ser empregado na criação de serviços. Um serviço está associado às necessidades das aplicações como largura de banda, atraso, jitter e taxa de perda de pacotes.

Por outro lado, a abordagem conhecida como *MPLS (Multiprotocol Label Switching)* é considerada mais como uma ferramenta de engenharia de tráfego que uma arquitetura QoS. O MPLS utiliza um *label* (etiqueta ou rótulo) que marca um caminho previamente estabelecido por onde um determinado pacote deve seguir na rede. O *label* é uma informação de tamanho fixo colocada no cabeçalho dos pacotes de dados entre as camadas 2 (enlace) e 3 (rede) do modelo OSI. Este fato permite diminuir o tempo de processamento gasto pelos roteadores na consulta de tabelas de rotas para encaminhar de datagramas.

3. Proposta de Configuração

Embora sejam verificadas algumas características comuns entre as abordagens *DiffServ* e *MPLS*, muitas de suas características são complementares. Em adição, estas propostas foram concebidas para solucionar problemas distintos, todavia é possível combiná-las em uma única arquitetura híbrida visando a melhoria de resultados à implementação de QoS fim-a-fim.

Para se atingir um grau de satisfação na QoS fim-a-fim é importante que se faça uma distinção entre as funções das abordagens envolvidas para facilitar a compreensão de como estas podem interagir. Um outro cuidado a ser verificado é a utilização da configuração propriamente dita. Para um melhor entendimento da nossa pesquisa a utilização de ambiente real de rede poderia significar algumas imprecisões nas observações, uma vez que ficaria impossível o controle de toda a rede. Assim, decidimos optar pela execução de simulações que pudessem responder nossas entradas de uma forma mais precisa e assim melhor entender as relações do ambiente híbrido. Em outras palavras, a intenção do nosso trabalho é simular uma rede IP com as arquiteturas *DiffServ* e *MPLS* em conjunto. Semelhante a outros trabalhos de pesquisa ([8] [9] [10]) acreditamos que esta decisão é acertada, pois compreender a interação entre diversos parâmetros que podem ser depois aplicados em um caso real.

4. Topologia e Resultados Experimentais

A topologia escolhida para implementação da simulação é a mostrada na figura 1. A configuração escolhida teve com base as observações de contexto teórico das abordagens

dos serviços diferenciados e MPLS. Utilizamos para simulação o pacote de software *Network Simulator* (versão 2.1b9).

Em nossa topologia existem dezesseis fontes de tráfego, sendo que cada quatro fontes (f1, f2, f3 e f4) entram em um dos quatro nós (s1, s2, s3 ou s4) existentes ainda fora do domínio MPLS/DiffServ. Esses quatro nós são ligados à dois roteadores limítrofes (LSR1 e LSR6) através de enlaces de 100Mbps e cuja latência física é de 1ms. Os pacotes atravessam a rede, sendo encaminhados pelos LSRs (*Label Switch Routers* da rede MPLS), conforme o protocolo de roteamento e chegam a dois outros roteadores (LSR2 e LSR8) limítrofes que funcionam como elementos egressos para a rede MPLS/DiffServ.

Os elementos egressos entregam os pacotes a quatro nós receptores (d1, d2, d3 ou d4) e deste ponto são encaminhado às suas respectivas aplicações receptores. Há três caminhos possíveis para cada pacote gerado. Por exemplo, um fluxo gerado no nó s1 pode seguir três caminhos para chegar a um nó destino d4 (LSR1_LSR2_LSR3_LSR5_LSR8_d4, LSR1_LSR4_LSR5_LSR8_d4, ou LSR1_LSR4_LSR6_LSR7_LSR8_d4).

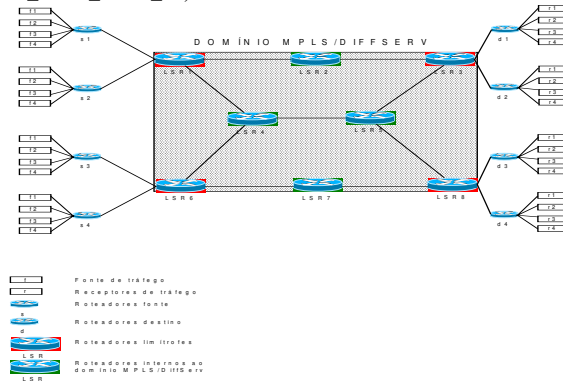


FIGURA 1: TOPOLOGIA FÍSICA DA SIMULAÇÃO

Como resultado podemos mostrar gráficos de parâmetros QoS retirados das simulações realizadas sobre a topologia da figura 1, porém com a rede implementando diferentes arquiteturas de qualidade de serviço e engenharia de tráfego tendo como tráfego fontes cbr que são similares à tráfegos multimídia e fontes ftp que, neste caso representam o tráfego TCP.

Primeiramente foi realizada uma simulação onde todos os roteadores implementam o modelo *best-effort* com tratamento de fila *drop-tail* (ou FIFO). O gráfico da figura 2 mostra que o throughput médio é bem baixo, principalmente para o tráfego TCP. Isso ocorre porque o tráfego UDP (tipicamente mal comportado) *rouba* banda do tráfego TCP (tipicamente bem comportado). Desta forma, todos os parâmetros indicam queda da qualidade de serviço da rede.

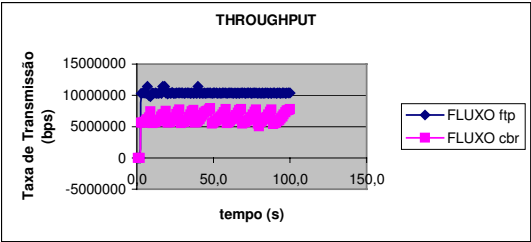


Figura 2: Variação do throughput dos fluxo cbr e ftp para rede sem QoS

A figura 3 apresenta o resultado da segunda simulação, onde foram realizados testes na rede da figura 1, porém com MPLS habilitado. Nesse caso dá pra notar uma pequena melhoria no throughput médio. Isso se deve ao fato de que a tecnologia MPLS roteia os pacotes mais rapidamente que as redes IP convencionais.

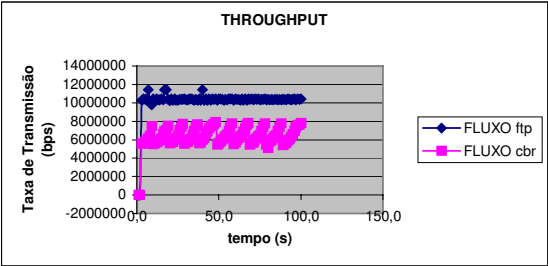


Figura 3: Variação do throughput dos fluxos cbr e ftp na rede MPLS

A figura 4 mostra os resultados da terceira simulação, os testes realizados na rede da figura 1, porém agora com mecanismos de serviços diferenciados habilitados. O gráfico mostra uma grande melhoria no tráfego cbr, uma vez que este tráfego foi classificado como serviço prioritário na rede DiffServ.

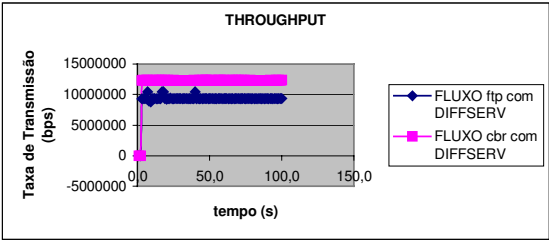


Figura 4: Variação do *throughput* dos fluxos *cbr* e *ftp* na rede *DiffServ*

Finalmente a figura 5 ilustra as medições para uma rede com MPLS e serviços diferenciados habilitados simultaneamente. Esse gráfico apresentou os melhores resultados em termos de qualidade de serviço. Isso mostra a viabilidade de usar as duas tecnologias simultaneamente numa rede IP.

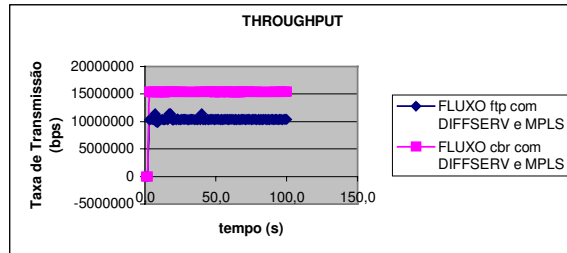


Figura 5: Variação do throughput dos fluxos cbr e ftp na rede DiffServ para rede com backbone MPLS/DiffServ

5. Conclusões

Neste artigo apresentamos simulações que nos mostraram a viabilidade e as vantagens que trazem a implementação das arquiteturas *DiffServ* e MPLS em conjunto. Neste sentido, para nossa análise de desempenho foi tomada como base as redes IP convencionais e aqueles com os mecanismos *DiffServ* e MPLS. Nossos experimentos indicam que atingimos com sucesso nossa proposta inicial de estudo de facilidades complementares para uma abordagem de qualidade de serviço fim-a-fim.

Referências

- [1] Behrouz A., *TCP/IP Protocol Suite* Book News, Inc., Portland, 1999
- [2] Davie, B., *MPLS Technology and Applications – 1st Edition*, Morgan Kaufmann Publishers, 2000.
- [3] Vegesna, S., *IP Quality of Service*, Cisco Press, 2002.
- [4] JSMNet Networking reviews, *Qualidade de Serviço em Redes TCP/IP*, http://www.jsmnet.com/Downloads/ReviewsJSMNet1_1.zip, Maio 2003
- [5] Santiago, F., *Um ambiente de Simulação de Serviços Integrados e Diferenciados*, Dissertação de Mestrado, Universidade de Brasília, 2001.
- [6] Osborne, E., *Engenharia de Tráfego com MPLS*, Cisco Press, 2003.
- [7] Santana, G. e Dantas, M., *Hybrid Quality of Service Architectures for IP Networks*, Universidade de Brasília, 2001.
- [8] Horlait, E., Rouhana, N., *Differentiated Services and Integrated Services Use of MPLS*, <http://www.computer.org/proceedings/iscc/0722/07220194abs.htm> 2003.
- [9] Law, R., Raghavan, S., *DiffServ and MPLS – Concepts and Simulation* http://saturn.acad.bg/bis/pdfs/04_doklad.pdf, Abril/2003.
- [10] Ahn, G., Chun, W., *Design and Implementation of MPLS Network Simulator Supporting LDP and CR-LDP* http://flower.ce.cnu.ac.kr/~fog1/mns/mns1.0/MNS_v1.0_arch.pdf, Abril/2003. <http://keskus.hut.fi/tutkimus/imelio/papers/qosr/spects2001.pdf>, Abril/2003.
- [11] Internet Software Consortium, *Internet Domain Survey*, July 2002, <http://www.isc.org/ds>

Roteando redes IPv6 e Multicast com MP-BGP

Andrey Vedana Andreoli, Leandro Márcio Bertholdo, Liane Tarouco

Ponto de Presença da RNP no Rio Grande do Sul – POP-RS
RSiX – Ponto de Troca de Tráfego do Rio Grande do Sul
Centro de Processamento de Dados da Universidade Federal do Rio Grande do Sul
Rua Ramiro Barcelos, 2574 – Porto Alegre – RS – Brasil
{andrey,bertholdo,liane}@penta.ufrgs.br

Resumo. *Este artigo apresenta as extensões feitas no protocolo BGP que formam o Multiprotocol BGP, tendo o intuito de fornecer suporte para o roteamento global da Internet para novas tecnologias como Multicast e IPv6. São apresentados detalhes sobre as mudanças no funcionamento e operação do BGP para o novo MP-BGP. Ao final, são dados exemplos de utilização e uma visão geral de sua utilização no Brasil.*

Palavras chave: Roteamento, BGP, Multiprotocol BGP, MP-BGP, IPv6, Multicast.

1. Informações Gerais

Com o grande crescimento da Internet desde seus primórdios, o modelo conhecido como hierárquico, constituído por uma rede centralizada que conectava redes secundárias, foi substituído por um modelo distribuído. Para tanto, adotou-se o conceito de sistemas autônomos, também chamados de ASes. Isso torna a Internet um conjunto de “células” conectadas entre si, chamadas de ASS. Essa formação garante que todos os ASs sejam alcançáveis entre si, visto que estarão conectados direta ou indiretamente. Outro fato importante é que o que ocorre internamente a um AS não é conhecido pelos demais ASes, diminuindo a complexidade da Internet global. O protocolo que iniciou a utilização desse conceito e permanece até os dias de hoje é o BGP (Border Gateway Protocol) [RFC 1771].

O funcionamento básico do BGP é baseado no estabelecimento de sessões entre dois roteadores, utilizando para isso o protocolo TCP, sob a porta 179. De forma muito freqüente o termo “peers” é atribuído a roteadores que estabelecem entre si uma sessão BGP e o termo “peering” é dado à troca de tráfego entre dois sistemas autônomos.

O protocolo BGP tem sido utilizado basicamente no roteamento IPv4, mas com o advento do IPv6, Multicast, MPLS e outras tecnologias, surgiu a demanda para que fossem feitas modificações afim de suportar essa nova gama de protocolos. Sabendo que o funcionamento do BGP tem se consolidado e que o funcionamento da Internet depende dele, surgiu uma extensão do protocolo BGP chamada de Multiprotocol BGP, tendo como principal objetivo fornecer suporte às novas tecnologias, mantendo a compatibilidade com o BGP atual.

O Multiprotocol BGP, erroneamente citado como Multicast BGP [HALABI 2000], tornou-se padrão em Fevereiro de 1998 [RFC 2858], atualmente já sendo implementado por fabricantes como Cisco, Juniper, entre outros. Como o padrão do protocolo não determina nenhuma abreviação específica para identifica-lo, encontram-se na literatura os termos MBGP e/ou MP-BGP para identificar o Multiprotocol BGP.

2. Descrição do MP-BGP

O protocolo MP-BGP apresenta-se como um complemento do protocolo BGP. Todo o seu funcionamento, políticas e atributos são mantidos. O que de fato muda é que na mesma sessão BGP podem ser trocados anúncios de múltiplos protocolos. A figura 1 apresenta o formato de uma mensagem tipo UPDATE convencional do BGP.

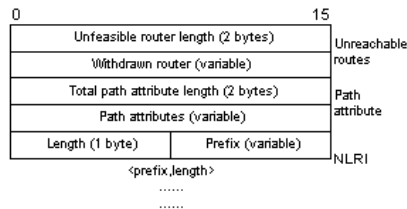


Figura 1 – Mensagem UPDATE do BGP

Conforme a figura acima, a mensagem de UPDATE é formada basicamente por 3 partes: rotas que devem ser removidas, atributos de rotas alcançáveis que estão sendo divulgadas e os prefixos que estão sendo anunciados. A mudança em si para o MP-BGP é feita nos atributos de rotas (Path attribute). Normalmente, os atributos utilizados são: AS_PATH, Origin, Next_Hop, entre outros. São apresentados então os novos atributos para suporte ao MP-BGP, listados abaixo:

- MP_REACH_NLRI: Utilizado para anúncios de redes alcançáveis a serem incluídas na tabela BGP;
- MP_UNREACH_NLRI: Utilizado para redes que devem ser retirados da tabela BGP;

Os campos pertencentes a cada um dos novos atributos são apresentados na figuras 2:

Address Family Identifier - AFI (2)
Subsequent Address Family Identifier – Sub-AFI (1)
Length of the Next-Hop Address (16 or 32)
Network Address of Netx-Hop (global and/or Link Local)
Number of SNPA's and Lenght (may be zero)
NLRI (Length and Prefix)

Address Family Identifier - AFI (2)
Subsequent Address Family Identifier – Sub-AFI (1)
WithDrawn Routes (Length and Prefix)

Figura 2 - Campos dos atributos MP_REACH_NLRI e MP_UNREACH_NLRI, respectivamente.

Como pode-se conferir na figura 2, surge o conceito de “Address Family Information” (AFI) e Subsequent Addresss Family Identifier (Sub-AFI). Estes campos são responsáveis por identificar os diversos protocolos em que o MP-BGP suporta.

Como exemplo mais prático, a figura 3 apresenta uma mensagem de UPDATE com o atributo MP_REACH_NLRI, onde são anunciados 7 prefixos de redes IPv6 Unicast:

```

❑ UPDATE Message
  Marker: 16 bytes
  Length: 139 bytes
  Type: UPDATE Message (2)
  Unfeasible routes length: 0 bytes
  Total path attribute length: 116 bytes
❑ Path attributes
  ❑ ORIGIN: IGP (4 bytes)
  ❑ AS_PATH: 65502 65504 (9 bytes)
  ❑ MP_REACH_NLRI (103 bytes)
    ❑ Flags: 0x80 (Optional, Non-transitive, Complete)
    Type code: MP_REACH_NLRI (14)
    Length: 100 bytes
    Address family: IPv6 (2)
    Subsequent address family identifier: unicast (1)
  ❑ Next hop network address (32 bytes)
    Subnetwork points of attachment: 0
  ❑ Network layer reachability information (63 bytes)
    ❑ 3ffe:2620:14:1::/64
      MP Reach NLRI prefix length: 64
      MP Reach NLRI prefix: 3ffe:2620:14:1::
    ❑ 3ffe:2620:14:2::/64
    ❑ 3ffe:2620:14:3::/64
    ❑ 3ffe:2620:14:4::/64
    ❑ 3ffe:2620:14:5::/64
    ❑ 3ffe:2620:14:100::/64
    ❑ 3ffe:2620:14:666::/64

```

Figura 3 - Mensagem de UPDATE com atributo MP_REACH_NLRI

Uma vez que a sessão MP-BGP entre dois peers seja estabelecida, são mantidas diferentes tabelas de roteamento devido aos diferentes protocolos que podem ser tratados [NANOG 2003]. As duas tabelas são: Unicast Routing Information Base (U-RIB) e Multicast Routing Information Base (M-RIB);

Algumas das tecnologias suportadas pelo MP-BGP, que são as mais utilizadas atualmente são descritas de forma mais detalhada abaixo:

a) Multicast

Conhecida também como comunicação de grupo, é uma tecnologia muito utilizada principalmente em transmissões de um único sentido destinadas a um grupo de hosts distribuídos em diferentes redes. Seu objetivo principal é otimizar o transporte de pacotes, evitando que diversas cópias de um mesmo fluxo de transmissão sejam transmitidas, economizando banda e recursos, além de garantir que o fluxo de dados seja entregue para todos que desejarem recebê-lo. É formado por um conjunto de protocolos de nível 2 e 3. Utilizado amplamente no Brasil pela Rede Nacional de Pesquisa na transmissão de eventos e cursos à distância. O MP-BGP interage com o Multicast e permite que anúncios multicast sejam feitos para seus peers [CISCO 2003a].

b) MPLS VPN

Sabendo que o funcionamento básico do protocolo MPLS é baseado em “labels” - onde cada roteador de uma rede MPLS provê um tratamento diferenciado de acordo com o label - torna-se possível estabelecer uma VPN (Virtual Private Network) [RFC 2547]. Alguns dos recursos que o MPLS fornece que podem ser úteis para uma VPN são: os níveis de garantia de QoS, latência e jitter, além de permitir o re-roteamento inteligente se necessário, que trata-se de um recurso que permite que o MPLS faça automaticamente o remanejo de um caminho, caso este não esteja mais atendendo aos recursos mínimos estabelecidos para seu funcionamento. Adicionalmente, o MPLS provê a devida segurança, visto que todos os pacotes estarão enquadrados sob a mesma classe de equivalência, tendo o mesmo tratamento e fazendo o mesmo caminho.

Como suporte, o MP-BGP permite que os anúncios de rotas sejam distribuídos entre peers na forma de <VPN-prefixo>.

c) IPv6

Com a crescente utilização e crescimento da Internet, o endereçamento IPv4 tem se tornando escasso. O que reforça essa tendência é a intenção de empregar endereçamento IP em aparelhos móveis, como celulares e computadores portáteis, aumentando ainda mais a necessidade de uma faixa de endereços maior. Diante desse problema, foi concebido o IPv6, passando dos 32 bits existentes no IPv4 para 128 bits, propondo também um conjunto de melhoras para limitações encontradas no IPv4. Com isso, o MP-BGP permite que prefixos IPv6 sejam anunciados, da mesma forma que prefixos IPv4 já são anunciados com o BGP.

d) 6PE

Também conhecida como IPv6 Provider Edge Router, o 6PE trata-se de uma funcionalidade presente em roteadores Cisco que tem por objetivo fornecer suporte a IPv6 sem a necessidade de que um backbone baseado em MPLS e IPv4 seja modificado. Para tanto, ele baseia-se na propagação de prefixos IPv6 e seus labels MPLS utilizando o BGP sobre IPv4, tendo seu atributo Next-hop identificado por um endereço IPv4.

Para a operação de uma rede com 6PE, os roteadores de distribuição que conectam sites/clientes IPv6 devem implementar o 6PE, sendo então aptos a anunciarem prefixos da família v6, juntamente com seu Next-hop IPv4, que é mapeado como “IPv4-mapped IPv6 address”, já que o Next-hop deve ser da mesma família que o prefixo anunciado nas mensagens da UPDATE do MP-BGP. O backbone em si não sofre nenhuma alteração, já que irá rotear os pacotes 6PE baseados apenas em seu label, até o próximo roteador 6PE.

Esta abordagem é muito semelhante a técnicas de tunelamento como é o caso do Generic Routing Encapsulation (GRE). Mesmo assim, a técnica de 6PE é muito utilizada em backbones europeus [6NET 2003], apresentando como principal benefício o menor overhead, além de não exigir que o backbone MPLS sofra modificações.

3. Negociação das chamadas capabilities

Outra característica do MP-BGP é a negociação de capabilities entre peers. As chamadas capabilities são features que podem ser muito úteis na operação de peerings BGP. Algumas delas já são conhecidas e utilizadas anteriormente ao MP-BGP. Tais features são negociadas ainda no início da sessão BGP, através das mensagens de OPEN. Para a negociação ser bem sucedida, ambos peers devem reconhecer e suportar a feature em questão. Abaixo são apresentadas 3 tipos que podem ser negociados:

- Multiprotocol extension (AFI e SAFI)

No início da sessão BGP, são negociados quais os protocolos que serão suportados por cada peer, determinando assim qual a AFI e SAFI que ambos estarão trocando prefixos.

- Route refresh

A motivação para esta capability é que o BGP não possui um recurso dinâmico para requisitar um novo envio dos prefixos contidos na tabela de um peer. De acordo com [RFC 2918], isso é desejável quando a política no recebimento de prefixos é alterada. O máximo que é feito até então era fazer uso do recurso de “soft-reconfiguration” que mantinha em memória todos os anúncios recebidos de determinado peer, afim de saber o que foi recebido e re-aplicar nos anúncios a nova política, sem ter que derrubar a sessão BGP. O recurso de

route refresh permite que os prefixos sejam enviados novamente por um peer, sendo disparado por uma requisição de route refresh, especificando qual a AFI e Sub-AFI desejada. Um dos benefícios desse método é a economia de memória e CPU utilizada com o recurso de soft-reconfiguration.

- Outbound route filtering

O processo de geração e processamento de prefixos entre peers BGP é uma operação que consome recursos valiosos. Em alguns casos, enquanto um dos peers - geralmente o upstream - gera os prefixos, o outro peer que recebe aplica sua prefix-list na entrada, afim de receber apenas os prefixos desejados. Em situações como esta o recurso de outbound route filtering permite que o peer que irá fazer os anúncios aplique como prefix-list de saída a mesma prefix-list que é utilizado na entrada do peer que irá receber os anúncios [CISCO 2003b]. Isso pode reduzir de forma significativa a troca de mensagens de UPDATE, além de poupar recursos valiosos de ambos.

4. Exemplos de sessão MP-BGP

Para demonstrar a configuração básica deste protocolo, foi criada uma topologia simples, onde dois roteadores estabelecem uma sessão MP-BGP, ilustrada pela figura 4:

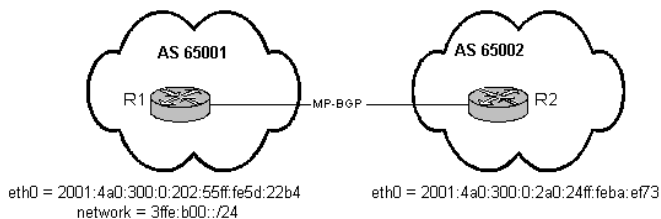


Figura 4 - Exemplo de sessão MP-BGP

Abaixo, na figura 5, é apresentada a configuração aplicada no roteador R1.

```
router bgp 60001
bgp router-id 10.0.0.1
no bgp default ipv4-unicast
neighbor 2001:4a0:300:0:2a0:24ff:feba:ef73 remote-as 60002
neighbor 2001:4a0:300:0:2a0:24ff:feba:ef73 description connection to R2
address-family ipv6 unicast
network 3ffe:b00::/24
neighbor 2001:4a0:300:0:2a0:24ff:feba:ef73 activate
neighbor 2001:4a0:300:0:2a0:24ff:feba:ef73 soft-reconfiguration inbound
exit-address-family
```

Figura 5 - Configuração MP-BGP em R1

Depois de ambos roteadores estarem configurados, pode-se verificar o status da sessão MP-BGP pela ilustração na figura 6. Verifica-se então que a sessão está estabelecida (UP) e que 1 prefixo foi recebido de R1.

```

R2> sh bgp ipv6 sum
BGP router identifier 10.0.0.2, local AS number 60002
Neighbor          V    AS  MsgRcvd MsgSent  TblVer  InQ OutQ Up/Down  State/PfxRcd
2001:4a0:300:0:202:55ff:fe5d:22b4 4 60001   24317   24696    0    0    0 14:42:25    1
Total number of neighbors 1

```

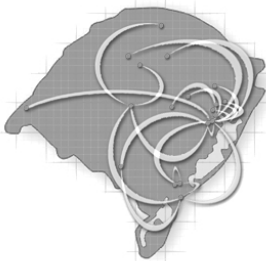
Figura 6 - Exemplo de verificação de sessão MP-BGP em R2

5. Visão geral da utilização do roteamento IPv6 e Multicast

Por serem tecnologias novas, backbones comerciais, em sua maioria, ainda não tem identificado o nicho de mercado para justificar tais inovações sobre o IPv4. Essa mudança deverá ser gradual, conforme aumentar a demanda por tais tecnologias. Já no meio acadêmico, tais tecnologias já têm sido amplamente empregadas, justificando-se pelo próprio caráter de pesquisa e inovação que tais instituições apresentam. No Brasil, assim como na Internet2, o MP-BGP tem sido utilizado pela Rede Nacional de Ensino e Pesquisa em caráter de produção, mais precisamente na conexão internacional afim de fazer peerings IPv6 e Multicast. Internamente ao sistema autônomo da RNP, é utilizado o protocolo PIM-SM para distribuição do tráfego Multicast e RIP6 para o roteamento entre os Pontos de Presença Estaduais que participam do projeto piloto de IPv6 [RNP 2003]. A escolha pelo RIP6 foi feita baseada na situação da época onde outros protocolos IGP, como era o caso do OSPF, não estavam inteiramente estáveis no suporte a essa tecnologia.

6. Referências bibliográficas

- [HALABI 2000] Sam Halabi, Danny McPerson. Internet Routing Architectures, Second Edition. Indianapolis – USA : Cisco Press, 2000
- [RFC 1771] A Border Gateway Protocol 4 (BGP-4) – RFC 1771 – The Internet Engineering Task Force – IETF
- [RFC 2858] Multiprotocol Extensions for BGP-4 – RFC 2858 - The Internet Engineering Task Force – IETF
- [RFC 2918] Route Refresh Capability for BGP-4 – RFC 2918 – The Internet Engineering Task Force – IETF
- [RFC 2547] BGP/MPLS VPNs – RFC 2547 – The Internet Engineering Task Force - IETF
- [NANOG 2003] Meeting of NANOG – on line – 2003 - www.nanog.org/mtg-0306/pdf/mcbride.ppt
- [CISCO 2003a] Cisco Enterprise : Multicast – on line – 2003 - <http://www.cisco.com/warp/public/779/largeent/learn/technologies/multicast.html>
- [6NET 2003] Initial Ipv4 do Ipv6 migration Cookbook for organizational/ISP (NREN) and backbone networks. on line – 2003 - www.6net.org/publications/deliverables/D2.2.2.pdf
- [CISCO 2003b] BGP Prefix-Based Outbound Route Filtering – on line – 2003 - http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120limit/120s/120s22/s_bpborf.htm#xtocid0
- [RNP 2003] Site da Rede Nacional de Pesquisa – on line – 2003 – <http://www.rnp.br>



Sessão Técnica 5

Comunicação sem Fio e Mobilidade

Roteamento com Agregação de Dados em Redes de Sensores

Jorgito MatiuZZi Stochero, Antonio José Gonçalves Pinto e José Ferreira de Rezende

Grupo de Teleinformática e Automação – Universidade Federal do Rio de Janeiro (UFRJ) Caixa Postal 68504 21945-970– Rio de Janeiro – RJ – Brasil

{stochero, antonio, rezende}@gta.ufrj.br

Abstract. *Sensor networks are a specialized form of mobile network where computing devices are tiny remote nodes, provided with sensing, processing, and communication capabilities. These devices collect data from the environment, interact with each other and pass it along until it reaches the users. They differ from traditional networks because sensor nodes are densely deployed, prone to failures and task specific, imposing new challenges for routing protocols. These networks are data centric, because data is more important to the network than the nodes themselves, and the protocols should take this into account during routing and addressing. In this paper we describe sensor network routing protocols and propose a technique based on bayesian inference to aggregate multisensor data while routing, reducing message traffic in order to save energy and extend the lifetime of the network.*

Resumo. *Redes de sensores são uma forma especializada de redes móveis onde os dispositivos computacionais são pequenos nós remotos, capazes de sensoriar, processar e comunicar dados. Estes dispositivos coletam dados do ambiente, interagem uns com os outros até que o dado chegue ao usuário. Tais redes diferem das tradicionais porque seus nós são densamente distribuídos, sujeitos a falhas e atuam para tarefas específicas, o que impõe desafios aos protocolos de roteamento. Estas redes são centradas em dados, uma vez que a informação é mais importante que o nó que a obtém, o que deve ser levado em conta durante o roteamento e o endereçamento. Neste artigo descrevemos protocolos de roteamento para redes de sensores e propomos uma técnica baseada em inferência bayesiana para agregar dados de múltiplos sensores durante o roteamento, reduzindo o tráfego de mensagens para economizar energia e aumentar o tempo de vida da rede.*

1. Introdução

Recentes avanços em micro-eletrônica e em comunicações sem fio possibilitaram a implementação de um novo tipo de redes de computadores móveis: redes de sensores. É previsto que, em um futuro próximo, dispositivos sensores, dotados de componentes de sensoriamento, processamento e comunicações de dados em um único circuito integrado estarão disponíveis no mercado a um baixo custo. Estes dispositivos deverão ser menores do que 1cm³, pesar menos do que 100 g, e devem consumir pouca energia, para evitar troca de baterias [1].

Uma rede de sensores é composta de centenas a milhares desses dispositivos (*nós*) dispostos em uma área geográfica de interesse. Cada nó coleta dados do ambiente, transforma a informação obtida em uma descrição do fenômeno observado, e gerencia os protocolos de comunicação. Devido ao seu pequeno tamanho, os nós são colocados próximos ao fenômeno a ser monitorado, aumentando a acurácia da observação. Suas aplicações podem ser militares (na detecção e monitoração de movimentos inimigos em uma área inóspita), médicas (na monitoração remota de parâmetros vitais de um paciente), entre outras.

Os protocolos devem possuir características de auto-configuração para descobrir quais os nós possuem informação (*fontes*), quais precisam dessa informação (*destinos* ou *sorvedouros*), e qual a melhor forma de transferir a informação de um para o outro, com mínimo consumo de energia. De acordo com o modelo rádio de primeira ordem proposto em [2], comunicações de dados são as responsáveis pelo maior peso no consumo de energia da rede, em comparação com o sensoriamento e o processamento. Dessa forma, para economizar energia é interessante aumentar o processamento dos dados para diminuir o número de transmissões, bem como utilizar transmissões por distâncias curtas (dezenas de metros) entre os nós, obtendo com isso uma menor potência no transmissor. Como em geral os sorvedouros estão bem afastados das fontes, isto significa que um pacote de dados será transmitido ao longo de vários saltos usando nós intermediários. Este relacionamento muitos-para-um (*fontes para sorvedouros*), bem como o fato de que normalmente as fontes observam o mesmo fenômeno, sugere redundância nos dados sendo transmitidos.

O objetivo deste artigo é explorar esta redundância utilizando, durante o roteamento, uma técnica de agregação para minimizar a transmissão de dados na rede e com isso aumentar o seu tempo de vida. O artigo é organizado da seguinte forma: na seção 2 são discutidos algoritmos de roteamento em redes de sensores; na seção 3 são apresentadas técnicas de agregação de dados; a seção 4 traz a proposta do trabalho, a simulação realizada e os resultados obtidos e o trabalho é finalizado na seção 5.

2. Roteamento em Redes de Sensores

A figura 1 mostra uma rede de sensores típica.

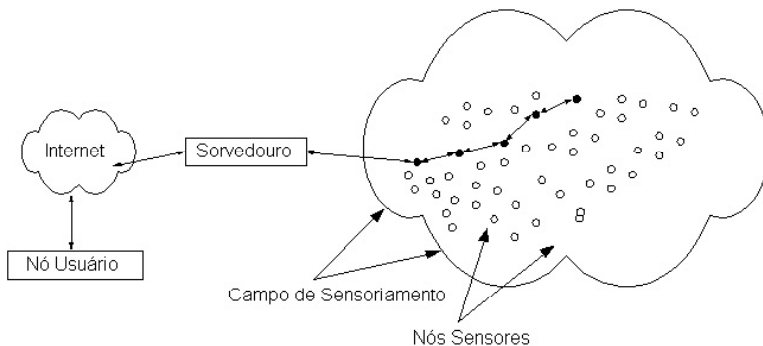


Figura 1. Rede de Sensores

O sorvedouro pode ser o consumidor final dos dados coletados, ou apenas um *gateway* para uma rede externa (Internet). Em geral, a utilização da rede de sensores depende da aplicação. O sorvedouro recebe as tarefas de sensoriamento da aplicação e as difunde até que as fontes iniciem as suas atividades. Sendo assim, o fluxo de tarefas vai do sorvedouro até as fontes, enquanto os dados fluem no sentido inverso. Dependendo do estado da rede, o mesmo nó pode atuar ora como sorvedouro, ora como fonte, caracterizando uma topologia de rede flexível e específica para cada aplicação.

Existem duas famílias de protocolos de roteamento aplicáveis para redes de sensores: hierárquicos ou planos. A figura 2 ilustra as diferenças:

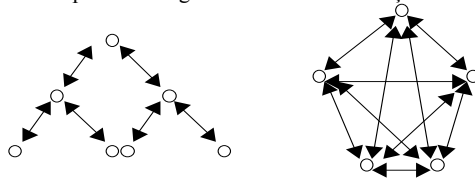


Figura 2. Protocolos de roteamento planos e hierárquicos

Na figura da esquerda é observada a topologia hierárquica, caracterizada pela existência de grupamentos de nós (*clusters*). Cada nó dentro do *cluster* comunica-se apenas com o seu pai (*cluster head*), que pode formar *clusters* de ordem mais alta. Cada *cluster head* integra a informação dos nós abaixo dele por meio de alguma técnica de agregação e reporta os resultados acima na hierarquia. O nó de mais alto nível é chamado de estação base e é responsável pela entrega da informação ao usuário. Os problemas nesta classe de protocolos estão na seleção dos *cluster heads* e na configuração da hierarquia de forma a garantir um balanceamento no gasto de energia dos nós. Exemplos de protocolos hierárquicos incluem o LEACH [2] e o PEGASIS[3].

Na figura da direita é observada a topologia plana. Cada nó é autônomo e conectado a alguns ou a todos os outros nós, dependendo do alcance do seu rádio. Não existem *cluster heads* ou estações bases. Esta abordagem causa uma pesada carga nas comunicações da rede, pois são necessárias n^2 conexões para uma rede de n nós. Para evitar isso, os protocolos devem identificar e privilegiar os melhores caminhos das fontes ao destino e limitar dessa forma o número de conexões. Exemplos de protocolos planos são SPIN [4], EAR [5], e Difusão Direcionada, ou *Directed Diffusion* [6].

O roteamento deve considerar ainda dois outros aspectos: endereçamento e o fato de redes de sensores serem centradas em dados, ou seja, o dado a ser transmitido é mais importante do que o nó que o transmite. O elevado número de nós e os escassos recursos de processamento tornam a utilização de um identificador (ID) global único para cada nó um aumento nos custos de transmissão. Sendo assim, os nós podem ser reconhecidos pelo tipo de informação que provêm (como temperatura, presença de fumaça, etc), caracterizando o que se chama de endereçamento baseado em atributos, facilitando a agregação de dados perto de onde eles são gerados. Exemplos de esquemas de endereçamento baseado em atributos podem ser achados em [7] e [8].

Um modelo para roteamento centrado em dados foi sugerido em [9], onde foi comparado com o centrado em endereços. Em [9], os nós intermediários lêem o conteúdo da mensagem e processam alguma função de agregação de pacotes, reduzindo o número de transmissões. No roteamento centrado em endereços, cada fonte envia os

dados de forma independente ao destino. A agregação pode adicionar retardo, porque os dados de fontes próximas precisam esperar por dados de fontes mais distantes para serem combinados. Algumas técnicas de agregação são discutidas a seguir.

3. Técnicas de Agregação de Dados

Na agregação de dados, uma rede de sensores é vista como uma árvore reversa, na qual o sorvedouro pede aos sensores que relatem as condições do fenômeno observado. Os dados que chegam de múltiplos sensores são agregados a cada nó com base nos atributos observados. Em [4] a agregação é explicada como uma técnica usada para solucionar o problema da implosão (*implosion*) e da sobreposição (*overlap*) em roteamento centrado nos dados.

As técnicas utilizadas para combinar ou agregar dados podem ser triviais como as funções (“máximo”, “média”, “contador”, etc.) exploradas em [10], adequadas para fusão de dados de um mesmo tipo de sensor, ou complexas como as apresentadas em [11], que realizam a fusão de dados de múltiplos sensores. A maioria das aplicações de agregação discutidas em pesquisas de redes de sensores são baseadas em funções simples. Nesse trabalho estudamos um cenário mais complexo, com o uso de técnicas de fusão de dados de diferentes tipos de sensores, descritas a seguir: inferência clássica, inferência bayesiana, os métodos Dempster-Shafer e heurísticas.

A inferência clássica procura determinar a validade das hipóteses propostas (versus hipóteses alternativas) baseada em probabilidades empíricas. Calcula a probabilidade conjunta dada uma hipótese assumida. Em geral pode validar somente duas hipóteses de cada vez (hipótese nula e sua alternativa) e não considera a informação a priori da crença da hipótese proposta.

A inferência bayesiana atualiza as probabilidades das hipóteses alternativas baseada nas observações de evidências. A fórmula de Bayes fornece uma relação entre a probabilidade a priori, a probabilidade condicional de uma observação dada uma hipótese e a probabilidade a posteriori de uma hipótese. Novas informações são utilizadas para atualizar a probabilidade a priori da hipótese em um procedimento análogo a um experimento científico. Probabilidades empíricas ou subjetivas podem ser utilizadas e várias hipóteses podem ser avaliadas simultaneamente.

O método Dempster-Shafer (DS) é uma generalização da inferência bayesiana. O método DS relaxa a restrição bayesiana quanto as hipóteses mutuamente exclusivas, adicionando um grau de incerteza.

As heurísticas utilizam várias técnicas não probabilísticas. Alguns métodos imitam a forma como humanos realizam inferência. São exemplos dessas técnicas os algoritmos de votação e técnicas de consenso, entre outras.

4. Proposta, Simulação e Resultados

Utilizamos o protocolo de roteamento *Directed Diffusion* [6], com a adição de um módulo para agregação de dados na rede utilizando inferência bayesiana. A rede é constituída por dois tipos de nós fonte, um para medir a temperatura, e outro para detectar fumaça. A aplicação necessita detectar fogo na área e a rede procura agregar as informações que colaborem com a hipótese de fogo antes de que cada mensagem (fumaça ou temperatura) chegue individualmente ao sorvedouro. Nesta seção avaliamos o desempenho dessa rede e verificamos se existe algum ganho com a proposta.

4.1 Fundamentação Teórica

A inferência bayesiana é baseada no teorema de Bayes. Supondo que H_1, H_2, \dots, H_j representam hipóteses mutuamente exclusivas e exaustivas que podem explicar um evento E (uma observação) que acabou de ocorrer, temos:

$$\text{onde, } p(H_i/E) = \frac{p(E/H_i)p(H_i)}{\sum_i p(E/H_i)p(H_i)} \quad \sum_i p(H_i) = 1$$

$p(H_i/E)$ = probabilidade a posteriori da hipótese H_i ser verdadeira dado a evidência E

$p(H_i)$ = probabilidade a priori da hipótese H_i ser verdadeira

$p(E/H_i)$ = probabilidade de observar a evidência E , dado que H_i é verdadeira

4.2 Cenário de Simulação

O cenário de simulação consiste em verificar a hipótese *fogo* levando-se em consideração as evidências *temperatura* e *fumaça*. Assumindo serem conhecidas as probabilidades a priori das três variáveis, a fórmula de Bayes é usada para calcular, a partir das evidências coletadas no local, a probabilidade de existir fogo. Para minimizar o impacto do retardo da rede, a agregação é realizada nó a nó com a informação disponível, ou seja, o nó utiliza um cache local para fazer a agregação, caso já tenha recebido uma mensagem similar. Ele não aguarda a chegada de novas mensagens. Foi utilizado o simulador ns-2 [12] e o campo de sensoriamento consistiu de uma área de 670 x 670 m, onde foram dispostos aleatoriamente 100 nós sensores. Cada nó possui um raio de cobertura de 250 m e as fontes e o sorvedouro são posicionados afastados.

4.3 Resultados

Embora a formação de uma árvore de agregação ótima seja geralmente um problema NP-completo [9], existem árvores geradas heurísticamente que possibilitam prolongar o tempo de vida da rede. As simulações iniciais apresentaram significativa redução no número de comunicações, e portanto, economia de energia, como pode ser visto nos gráficos da figura 3 para simulações de 500 e 1000 segundos.

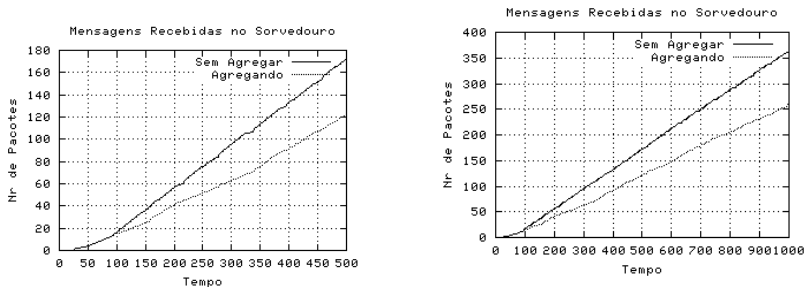


Figura 3. Topologia de 100 nós, campo de sensoriamento de 670 x 670 m

5. Conclusão e Trabalhos Futuros

Neste artigo descrevemos as características principais das redes de sensores, especialmente no que diz respeito ao roteamento de dados. Como o principal fator limitante ao tempo de vida útil de uma rede de sensores é a durabilidade das baterias dos seus nós, os protocolos de roteamento devem buscar a minimização do tráfego de mensagens e, para isso, aproveitar-se de endereçamento baseado em atributos e nas características centradas em dados da rede. Utilizamos o protocolo *Directed Diffusion* com o acréscimo de um módulo para agregação de dados utilizando inferência bayesiana e nas simulações efetuadas observamos a diminuição do tráfego de mensagens das fontes ao destino.

Como próximos trabalhos, pretendemos avaliar o desempenho de outras técnicas de agregação e compará-las com os resultados obtidos com a inferência bayesiana. Pretendemos também acrescentar um temporizador em cada nó para aguardar novas mensagens antes de agregar, aumentando o descarte de pacotes, verificando a influência no retardo sofrido pelos dados desde a fonte até o sorvedouro. Outro foco de interesse é em modificar o estabelecimento de rotas no *Directed Diffusion* para beneficiar nós intermediários com potencial para agregação e com isso aumentar os ganhos em energia com uma redução mais acentuada do tráfego.

6. Referências

- [1] J. Rabaey, J. Ammer, J. L. da Silva Jr., D. Patel. "PicoRadio: Ad-hoc Wireless Networking of Ubiquitous Low-Energy Sensor/Monitor Nodes". IEEE WVLSI 2000.
- [2] W. Heinzelman, A. Chandrakasan and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks" Proceedings of the 33rd Hawaii International Conference on System Sciences, January 2000.
- [3] S. Lindsey, C. S. Raghavendra, "PEGASIS: Power Efficient Gathering in Sensor Information Systems," IEEE ICC 2001.
- [4] Joana Kulik, Wendi Heinzelman, and Hari Balakrishnan. "Negotiation-based Protocols for Dissemination Information in Wireless Sensor Networks". IEEE/ACM Mobicom 1999.
- [5] Rahul C. Shah and Jan Rabaey, "Energy Aware Routing for Low Energy Ad Hoc Sensor Networks", IEEE Wireless Communications and Networking Conference 2002.
- [6] Chalermek Intanagonwiwat, Ramesh Govindan, and Deborah Estrin. "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks". Mobicom, 2000.
- [7] J. Heidemann, F. Silva, C. Intanagonwiwat, R. Govindan, D. Estrin, and D. Ganesan, "Buiding Efficient Wireless Sensor Networks with Low- Level Naming". 18th ACM Symposium on Operating Systems Principles, 2001.
- [8] William Adjie-Winoto, Elliot Schwartz, Hari Bvalakrishnan, and Jeremy Lilley. "The design and implementation of an intentional naming system". 17th ACM Symposium on Operating Systems Principles, 1999.
- [9] Bhaskar Krishnamachari, Debora Estrin, Stephen Wicker. "Modeling Data-Centric Routing in Wireless Sensor Networks". IEEE INFOCOM 2002.
- [10] A. Boulis, S. Ganeriwal, M. Srivastava. "Aggregation in Sensor Networks: An Energy-Accuracy Trade-off". Sensor Network Protocols and Applications, Special Issue of Elsevier Ad Hoc Networks Journal, 2003.
- [11] L. A. Klein. "Sensor and Data Fusion Concepts and Applications". Second Edition. Tutorial Texts in Optical Engineering SPIE PRESS Vol. TT355, 1999
- [12] The Network Simulator ns-2, <http://www.isi.edu/nsnam/ns>, acessado em 21 Ago 2003.

A Tecnologia CDMA: Revisão Teórica e Aplicações em Sistemas *Wireless*

Diego Moreira da Rosa¹, Fabio Irigon Pereira¹, Juergen Rochol¹

¹Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil
{diegoro,fip,juergen}@inf.ufrgs.br

Abstract. *The world is demanding more from wireless communications technologies than ever before. In this context, CDMA (Code Division Multiple Access) distinguishes itself from other technologies, being recently choosen to be the basis of third generation wireless systems. This paper presents a review of the basic concepts of the spread spectrum technique, over wich CDMA was built, as well as an overview of the applications of CDMA technology in wireless systems, from the IS-95 second generation standard to the third generation standards cdma2000 and WCDMA.*

Resumo. *O mundo tem exigido cada vez mais das tecnologias de comunicações wireless. Nesse contexto, desponta a tecnologia CDMA (Code Division Multiple Access), sendo recentemente escolhida para servir de base para os serviços de terceira geração. Esse trabalho apresenta uma revisão teórica dos conceitos básicos de spread spectrum, técnica sobre a qual está fundamentado o CDMA, bem como um panorama geral das aplicações de CDMA em sistemas celulares, do padrão IS-95 ou cdmaOne de segunda geração aos padrões de terceira geração cdma2000 e WCDMA.*

1. Introdução

O mundo tem exigido cada vez mais das tecnologias de comunicações *wireless*. Cada vez mais pessoas no mundo inteiro estão utilizando serviços baseados em comunicações móveis. Adicione-se a isso o advento da terceira geração (3G) de sistemas *wireless* com seus serviços de transmissão de dados e aplicações inovadoras e tem-se as tecnologias *wireless* sendo obrigadas a oferecer muito mais do que a alguns anos atrás.

É nesse contexto que a tecnologia CDMA desponta, sendo utilizada largamente em serviços celulares de segunda geração desde 1995 e recentemente sendo escolhida para servir de base para os serviços 3G. Baseados na técnica de espalhamento espectral ou *spread spectrum* (SS), os sistemas CDMA garantem o acesso de múltiplos usuários ao meio simultaneamente e utilizando a mesma banda de frequências através de um esquema de divisão por código. A robustez desses sistemas quanto a ruído e interferência estão entre as principais vantagens que levaram a tecnologia CDMA a atingir tamanho sucesso. O objetivo deste trabalho é apresentar uma revisão teórica dos conceitos básicos de SS bem como apresentar um panorama geral das aplicações de CDMA em sistemas celulares, passando pelo padrão IS-95 ou cdmaOne de segunda geração e pelos padrões de terceira geração cdma2000 e WCDMA.

Este trabalho está dividido em quatro seções incluindo esta introdução. Na seção 2, são apresentados os fundamentos de *spread spectrum*. Na seção 3, são apresentadas algumas características do padrão IS-95 ou cdmaOne de segunda geração bem como dos padrões de terceira geração. Por fim, na seção 4 são apresentadas conclusões e sumário.

2. Fundamentos de *spread spectrum*

O sinal utilizado convencionalmente para a transmissão de dados, como, por exemplo, em sistemas com múltiplo acesso por divisão de frequência, utiliza toda sua potência em uma parcela estreita da largura de banda disponível para a transmissão, economizando assim recursos para outros rádio-transmissores operarem em outras faixas de frequência. O problema da banda estreita surge quando um segundo transmissor, intencionalmente ou não, transmite um sinal na - ou muito próximo da - faixa de frequência do primeiro transmissor, interferindo na comunicação.

SS é uma forma de modulação em que o sinal, com uma banda relativamente estreita, é espalhado em uma faixa de frequência mais larga através da utilização de um código independente dos dados, o qual é conhecido pelo receptor. A principal vantagem do uso de *spread spectrum* é sua resistência à interferência.

2.1. Sequências Pseudo-Aleatórias

O espalhamento de sinal se utiliza de uma sequência pseudo-aleatória (PN), binária e periódica, que pode ser gerada a partir de um registrador de deslocamento realimentado. A saída do registrador de deslocamento em cada ciclo de clock é denominada *chip*.

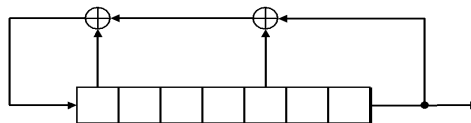


Figura 1. Exemplo de um gerador de sequências pseudo-aleatórias implementado através de um registrador de deslocamento.

Quando a lógica de realimentação é constituída apenas por somadores módulo 2 (função lógica “ou exclusivo”), o gerador é dito linear. Neste caso, o estado em que todas as entradas são zero é proibido, pois aí a saída da lógica de realimentação seria sempre zero e o registrador não sairia desse estado. Assim o maior período possível para um gerador linear é $2^n - 1$ *chips*, onde n é o número de bits do registrador de deslocamento. Conforme [1], quando essa situação é atingida, garante-se que a sequência tem algumas características interessantes: em cada período, o número de zeros é igual ao número de 1s menos 1; metade dos dígitos 1 ou 0 estão isolados, um quarto em pares, um oitavo em trios e assim sucessivamente até que termine a sequência; a função de autocorrelação é periódica e se assemelha muito com a correlação de um sinal binário randômico em um período e a potência espectral tem o mesmo envelope (sinc^2). A diferença é que, por ser periódica, a transformada de uma sequência PN é discreta, enquanto que a de uma sequência aleatória de bits é contínua.

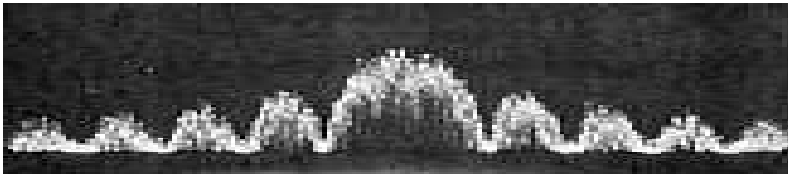


Figura 2. Potência espectral de um sequência pseudo-aleatória

2.2. Espalhamento Espectral por Sequência Direta

Pode-se identificar dois tipos principais de espalhamento espectral: espalhamento espectral por sequência direta ou *direct sequence spread spectrum* (DS-SS) e espalhamento espectral por saltos de frequência ou *frequency hop spread spectrum* (FH-SS). O primeiro realiza o espalhamento através da modulação direta dos dados pela sequência binária. O segundo realiza o espalhamento através de saltos aleatórios da frequência de transmissão. Como a tecnologia CDMA utiliza DS-SS, este trabalho se restringirá à análise desta técnica.

Na técnica DS-SS, proteção contra interferência é conseguida através do uso de uma banda muito superior à necessária para a transmissão. Como o sinal utilizará toda a banda, a densidade de potência em qualquer ponto será baixa, de modo que o sinal transmitido assume um comportamento de ruído, misturando-se no ambiente e dificultando a sua recepção por receptores indesejáveis. Este aumento de banda utilizada é conseguido através da modulação do sinal que se quer transmitir pela sequência PN (de banda mais larga) da seção anterior. Lembramos aqui que, ao multiplicarmos dois sinais, o espectro de frequência do sinal obtido será a convolução dos espectros dos dois sinais originais. Se a largura de banda desses sinais for muito diferente, a largura de banda do sinal resultante será muito próxima da largura da maior banda, ou seja, a sequência PN atuará como um “espalhador” do sinal original na faixa espectral disponível. Na prática, cada bit da mensagem transmitida será quebrado em vários outros dependendo do código utilizado.

Usualmente se utiliza uma taxa de chips entre 1 Mchips/s e 100 Mchips/s [2]. Pegamos como exemplo um sinal de informação de 10 KHz e 1 Watt de potência. A potência média desse sinal por unidade de banda é de 100 μ Watt/Hz. O mesmo sinal se espalhado com uma sequência PN de 10 Mchips/s (de banda 10 MHz) terá uma potência média de 0.1 μ Watt/Hz. O receptor multiplicará mais uma vez o sinal recebido pela sequência PN, reconstituindo o sinal original de banda estreita, que depois é filtrado por um filtro passa-baixas removendo o ruído adicionado pelo canal, que estará espalhado em toda a banda utilizada. Para que o sistema funcione corretamente, é necessário que a sequência PN no receptor esteja sincronizada com a do transmissor.

2.3. DS-SS com Phase-Shift Keying

No algoritmo anterior, apesar de utilizarmos uma forma de modulação, a transmissão foi realizada em banda-base. O padrão CDMA utiliza uma faixa de frequência pré-determinada, tornando uma segunda modulação necessária para deslocar o sinal já espalhado para a faixa de frequência do canal a ser utilizado. O padrão IS-95 da TIA

define a utilização de modulação em quadratura ou *quadrature phase shift keying* (QPSK) para a transmissão do sinal da estação base para os telefones móveis e a utilização de modulação em quadratura com offset (O-QPSK) para o sentido inverso. Essa diferença se deve à maior imunidade do sinal O-QPSK quanto à não-linearidades do amplificador utilizado na transmissão, facilitando o projeto dos telefones móveis [5].

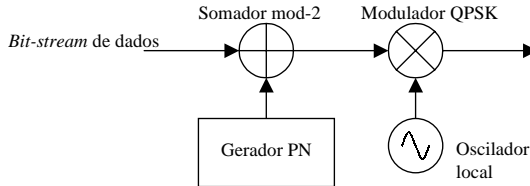


Figura 3. Diagrama de um transmissor DS-SS com QPSK

2.4. Ganho de Processamento

Como vimos no item 2.2, após a modulação, o sinal original é restaurado, enquanto o ruído continua com um espectro largo, que acaba por ser filtrado por um filtro passa-baixas. Segundo [2], o ganho em relação sinal ruído ou *signal to noise ratio* (SNR) obtido pelo uso de *spread spectrum* é:

$$PG = T_b/T_c \quad (1)$$

Onde T_b é o tempo de transmissão de um bit, e T_c o tempo de um chip. Isso quer dizer que um sinal espalhado com uma taxa de chips 1000 vezes maior do que a taxa de bits, terá sua SNR aumentada em 60 Db (1000 vezes) em relação ao mesmo sinal se não aumentado. Isso é que possibilita a utilização de potência tão baixa a ponto de não interferir na comunicação já existente ou permitir a sobreposição de vários transmissores na mesma banda no CDMA.

2.5. Probabilidade de Erro e Margem de Interferência

Existem na literatura várias propostas de cálculo da probabilidade do erro em SS. Entre elas pode-se destacar *Standard Gaussian Aproximation* (SGA) e *Improved Gaussian Aproximation* (IGA) [3]. Aqui utilizaremos a forma mais simplificada de cálculo. A probabilidade de haver um erro como demonstrado em [2] pode ser expressada assim:

$$P_e = (1/2) * \text{erfc}((E_b/N_o)^{1/2}) \quad (2)$$

$$E_b = P * T_b \quad (3)$$

$$N_o = J * T_c \quad (4)$$

Onde E_b e N_o são energia e P e J são potência de bit e chip respectivamente. Pode-se reescrever a equação (1) substituindo o tempo de bit $T_b = E_b/P$ e o tempo de chip $T_c = N_o/J$:

$$J/P = PG/(E_b/N_o) \quad (5)$$

E definir margem de interferência como sendo a relação entre a potência do ruído e a potência do sinal. Em escala logarítmica, a equação toma a seguinte forma:

$$M = J/P \quad (6)$$

$$M_{db} = 20 \cdot \log(PG) - 20 \cdot \log(E_b/N_o) \quad (7)$$

Supondo uma comunicação na qual o tempo de bit seja $T_b = 102.4 \mu s$ e $T_c = 100$ ns, segundo a equação (1), tem-se $PG = 1024$. O que significa dizer que a sequência PN precisa ter um período mínimo de 1023 chips e o tamanho mínimo do registrador de deslocamento para gerá-la é 10 bits. Digamos que, para uma boa recepção, a probabilidade de erro não deve passar de 10^{-5} . Da fórmula (2) tem-se que para $E_b/N_o = 10$ a probabilidade do erro fica em $0.387 \cdot 10^{-5}$. Logo,

$$M = 20 \cdot \log(1024) - 20 \cdot \log(10) = 40,2 \text{ db}$$

O que significa dizer que, para esse caso, o sinal pode ser detectado com segurança, mesmo que o ruído seja 40 db ou 102,3 vezes mais potente que o sinal transmitido.

2.6. Interferência mútua e *gold codes*

O uso de *spread spectrum* em CDMA se justifica por permitir que vários transceptores utilizem o mesmo canal ao mesmo tempo. Porém, para minimizar o efeito mútuo entre transmissão, toma-se alguns cuidados ao escolher a sequência PN utilizada no espalhamento espectral. Para explicar esses cuidados, começaremos introduzindo o conceito de ortogonalidade: duas funções serão ortogonais se a média do produto entre elas for zero [4]. Em telecomunicações, diz-se que, se dois sinais são ortogonais, então é teoricamente possível construir um receptor que responda a um deles enquanto rejeita completamente o segundo. Exemplos de funções ortogonais são: senóides de diferentes frequências; senóides em quadratura (diferença de fase de 90°); pulsos não sobrepostos (TDMA); e funções de Walsh, utilizadas no padrão IS-95.

Contudo, as funções ortogonais são limitadas. Além disso, a grande maioria não é ortogonal consigo mesma se atrasada no tempo, o que torna o fenômeno de caminhos múltiplos ou *multipath*, muito comum em sistemas wireless, um grave problema. Portanto, muitas vezes são utilizadas funções quase ortogonais muito mais numerosas e também quase ortogonais com cópias suas atrasadas no tempo. O preço que se paga pelo uso de sequências PN não completamente ortogonais é que acaba havendo interferência entre transmissores, a qual acaba sendo suprimida pelo ganho de processamento da seção 2.4. Contudo, há um limite para essa interferência, o que explica o famoso *near-far problem*, ou seja, um transmissor muito mais próximo da base pode acabar obstruindo a comunicação de outros agentes mais distantes, limitando o uso do meio. Esse problema é resolvido através de um controle da potência de transmissão.

3. Aplicações em Sistemas Celulares

Como descrito em [6], os primeiros estudos sobre SS datam do início do século passado, e suas primeiras aplicações em sistemas de comunicações de dados ocorreram no período da 2ª Guerra Mundial. Essas aplicações aconteceram em sistemas militares, nos quais a principal vantagem do espalhamento espectral é a imunidade à interferência de outros sinais. Nos anos 70 e 80 houve um crescimento no interesse pela aplicação de SS em sistemas comerciais e as primeiras propostas de sua utilização em sistemas celulares datam do final dos anos 70. No começo dos anos 90, a Qualcomm propôs um sistema de telefonia móvel baseado em CDMA (cdmaOne), o qual acabou sendo adotado, em julho

de 1993, pela TIA (*Telecommunications Industry Association*) no padrão IS-95 (*Interim Standard 95*). Entre as principais características do cdmaOne estão:

- compatibilidade com a antiga rede baseada no sistema AMPS;
- divisão da banda de 25 MHz em 10 canais duplex de RF com 1,25 MHz; cada canal de RF com 64 canais digitais de 9600 bps diferenciados através de códigos ortogonais baseados em *Walsh Code*; dos 64 canais, 55 de telefonia, 7 para serviço de mensagens e 2 para controle de acesso e sincronismo;
- modulação QPSK no downlink e O-QPSK no uplink;
- rígido controle de potência para minimizar o problema de interferência entre canais.

Em 1992, o ITU (*International Telecommunication Union*) iniciou um projeto no sentido de planejar a evolução dos chamados serviços 3G, os quais definiam uma rede celular para dados de alta velocidade e deveriam substituir os serviços 2G baseados principalmente nos sistemas D-AMPS, cdmaOne e GSM. Esse projeto é hoje conhecido como IMT-2000. Entre os sistemas 3G baseados no IMT-2000, destacam-se dois, ambos baseados na tecnologia CDMA: o cdma2000, definido pela TIA, e o WCDMA definido pelo ETSI (*European Telecommunications Standards Institute*). Esses dois sistemas possuem semelhanças herdadas do padrão IS-95; no entanto, possuem diferenças que os tornam incompatíveis. Entre as principais diferenças pode-se citar:

- cdma2000 possui compatibilidade com a rede cdmaOne, enquanto o WCDMA mantém compatibilidade com o sistema GSM;
- cdma2000 utiliza a técnica *multicarrier cdma* além da tradicional *direct sequence* a fim de obter taxas maiores mantendo compatibilidade com o sistema IS-95;
- taxa de 4.096 Mcips/s para o WCDMA e de 3.84 Mcips/s para o cdma2000;

4. Conclusões

Foi apresentada uma revisão dos conceitos e características da técnica de espalhamento espectral utilizada na implementação das tecnologias CDMA, cobrindo seqüências binárias, DS-SS, modulação em fase, probabilidade de erro e imunidade a interferência. Também foi apresentado um panorama das aplicações de CDMA em sistemas celulares.

5. Referências

- [1] Haykin, Simon (2001) "Communication Systems", 4ª ed., John Wiley & Sons Ltd.
- [2] Carne, E. Bryan (1999) "Telecommunications Primer", 2ª ed., Prentice Hall.
- [3] Lee, S. J., Kim, T. S. e Sung, D. K. (2001) "Bit Error Probabilities of DSSS Multiple Access Systems", IEEE Transactions on Communication Vol 49 No 1, John Wiley & Sons Ltd.
- [4] Karn, Phil (1997) "Introduction to Spread Spectrum", Digital Communications Conference.
- [5] Harte, Lawrence (1999) "CDMA IS-95 for Cellular and PCS", McGraw-Hill.
- [6] Goodman, David J. (1997) "Wireless Personal Communications Systems", Addison Wesley Longman, Inc.

Efeito da Precisão Numérica do Conversor sobre a Taxa de Erros de um *Software Radio*

Diego Moreira da Rosa¹, Luigi Carro¹

¹Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil
{diegoro,carro}@inf.ufrgs.br

Abstract. *The great processing capabilities of currently available microprocessors made possible the implementation of the first software radios, data transmitters/receivers based on software, in wich tasks like modulation/demodulation and encoding are totally implemented using digital signal processing techniques. This paper presents a proposal of a software radio based on quadri-phase shift keying and on a phase synchronization algorithm based on the recursive Costas loop as well as an analysis of this system behavior due to variations in the numerical precision of incoming data.*

Resumo. *Os avanços tecnológicos das últimas décadas e o surgimento de processadores cada vez mais rápidos permitiram o surgimento dos software radios, sistemas transmissores/receptores de dados baseados em software, nos quais tarefas como modulação/demodulação e codificação são totalmente implementadas através de técnicas de processamento digital de sinais. Esse trabalho apresenta uma proposta de software radio utilizando modulação QPSK (Quadri-Phase Shift Keying) e um algoritmo de sincronização de fase baseado no loop de Costas recursivo bem como analisa o comportamento desse sistema em relação à precisão numérica dos dados de entrada.*

1. Introdução

Os avanços tecnológicos das últimas décadas e o surgimento de processadores cada vez mais rápidos permitiu o surgimento dos *software radios*, sistemas transmissores/receptores de dados baseados em software, nos quais tarefas como modulação/demodulação e codificação são totalmente implementadas através de técnicas de processamento digital de sinais ou *DSP (Digital Signal Processing)*. Entre as principais vantagens desses sistemas em comparação com os sistemas analógicos estão a alta imunidade ao ruído, o baixo consumo de potência e a flexibilidade.

Usualmente esses sistemas são compostos por quatro componentes básicos: uma antena (no caso de sistemas móveis), um circuito analógico simples para o pré-processamento do sinal, um conversor AD/DA (Analogico-Digital/Digital-Analogico) e um processador. O circuito analógico na entrada do conversor é composto normalmente de um multiplicador e alguns filtros que adaptam a banda do sinal recebido pela antena à largura de banda do conversor. Nos *software radios*, o conversor AD/DA representa um fator importante em relação ao custo e à eficiência basicamente devido a três motivos:

- um conversor com maior largura de banda possibilita a minimização do circuito analógico necessário na sua entrada;
- conversores mais velozes e com maior precisão numérica permitem uma maior eficiência no processamento digital do sinal e um aumento nas taxas de transmissão;
- conversores com as características acima são normalmente complexos de se implementar e possuem um custo elevado.

Dada a importância do conversor AD/DA na implementação dos *software radios*, torna-se necessário um estudo detalhado do comportamento desses sistemas dadas variações do conversor utilizado. Os objetivos desse trabalho são apresentar uma proposta de *software radio* baseado em modulação QPSK e utilizando um algoritmo de sincronização de fase baseado no loop de Costas recursivo bem como analisar o comportamento desse sistema em relação à precisão numérica dos dados de entrada.

Este trabalho está dividido em quatro seções incluindo esta introdução. Na seção 2, é apresentada uma proposta para um sistema transmissor/receptor de dados baseado em software. Na seção 3, são apresentados os métodos utilizados nas simulações e os resultados obtidos. Por fim, na seção 4, são apresentadas conclusões e sumário.

2. Sistema Proposto

O tipo de modulação escolhido para o sistema foi o QPSK, dada sua facilidade de implementação e sua eficiência espectral. Análises mais aprofundadas das técnicas de modulação digital podem ser encontradas em [1] e [2]. Como o receptor deve estar sincronizado (em fase) com o transmissor para a demodulação correta do sinal, esse sistema também é chamado de detecção coerente o *coherent-QPSK*. Na modulação QPSK, a fase de uma portadora senoidal assume valores discretos representando valores binários. As quatro fases utilizadas para a codificação dos dados foram: $\pi/4$, $3\pi/4$, $5\pi/4$ e $7\pi/4$. Cada deslocamento de fase é denominado um símbolo e cada símbolo representa dois bits ou um *dibit*. A figura 1 associa cada *dibit* a sua fase correspondente.

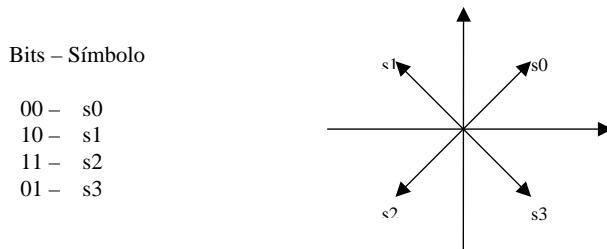


Figura 1. Associação de cada *dibit* a sua fase correspondente

O sinal transmitido em um sistema QPSK pode ser definido como sendo:

$$s_i = (2 \cdot E/T)^{1/2} \cdot \cos[2 \cdot \pi \cdot f_c \cdot t + (2 \cdot i - 1) \cdot \pi/4], \quad 0 \leq t \leq T \quad (1)$$

onde $i = 1, 2, 3$ e 4 corresponde a cada um dos símbolos; E é a energia transmitida por símbolo; T é a duração do símbolo e f_c é a frequência da portadora. Utilizando identidades trigonométricas, a equação acima pode ser escrita como

$$s_i = (2E/T)^{1/2} \cos[(2i-1)\pi/4] \cos(2\pi f_c t) - (2E/T)^{1/2} \sin[(2i-1)\pi/4] \sin(2\pi f_c t) \quad (2)$$

onde pode-se identificar duas funções ortonormais também definidas como portadoras em quadratura:

$$p_1 = (2/T)^{1/2} \cos(2\pi f_c t) \quad (3)$$

$$p_2 = (2/T)^{1/2} \sin(2\pi f_c t) \quad (4)$$

A partir das definições acima e baseando-se em [3], pode-se definir os blocos básicos de um transmissor e de um receptor QPSK. Na figura 2, tem-se o diagrama de blocos do transmissor QPSK proposto.

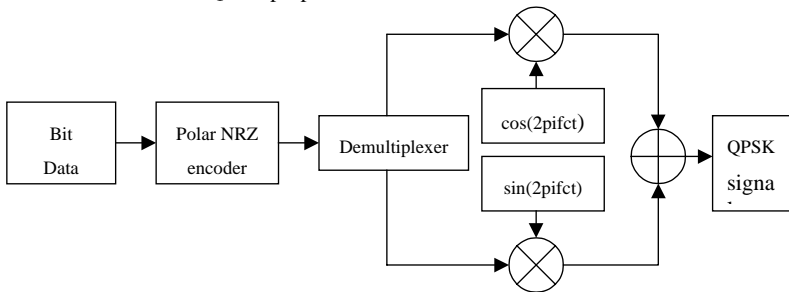


Figura 2. Diagrama de blocos do transmissor QPSK

O primeiro estágio é um codificador polar-NRZ que transforma a sequência binária da entrada em uma sequência polar, ou seja o valor zero é representado por -1 . O demultiplexador separa a sequência em bits de índices pares e ímpares. Os bits pares irão multiplicar a portadora em fase e os ímpares a portadora em quadratura. Por fim, a soma das duas portadoras moduladas gera o sinal QPSK final.

Na figura 3, tem-se o diagrama de blocos do receptor QPSK proposto.

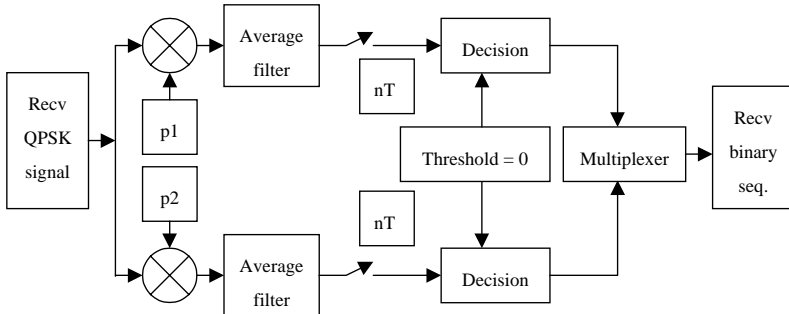


Figura 3. Diagrama de blocos do receptor QPSK

Inicialmente o sinal QPSK recebido é separado em dois fluxos. Os fluxos são multiplicados pela portadora em fase e em quadratura respectivamente. Após, os sinais passam por um filtro de média ou integrador com período igual ao período de um símbolo. Os sinais filtrados são então amostrados também com período de um símbolo. Se a saída for maior que zero, então é detectado um símbolo 1 e, caso contrário, zero. Por fim, as duas seqüências são multiplexadas em uma única seqüência que vem a ser a estimativa da seqüência binária enviada pelo transmissor. É importante notar que qualquer um dos blocos apresentados acima pode ser facilmente implementado em software, sendo esse fato requisito principal no projeto de um *software radio*.

O esquema apresentado acima para demodulação QPSK funciona apenas quando o transmissor e o receptor estão com suas portadoras sincronizadas em fase. Pequenas diferenças de fase podem causar resultados totalmente errados. Para resolver este problema, é preciso um bloco de sincronização da fase. Este bloco pode ser visto na figura 4.

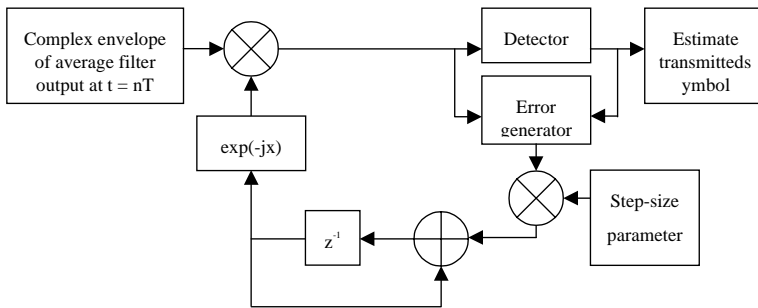


Figura 4. Diagrama de blocos do sub-sistema de sincronização da fase

O bloco acima pode ser visto como uma generalização recursiva do loop de Costas [3]. Uma boa aproximação para a função geradora do erro é

$$e[n] = \text{Im}[a_n^* x_n e^{-j^* \text{phi}}] \quad (5)$$

onde a_n^* é o complexo conjugado do sinal estimado e $x_n e^{-j^* \text{phi}}$ é a saída dos filtros de média após o ajuste de fase. Considerando os módulos dos números complexos como 1, podemos interpretar a função que calcula o erro como sendo o seno da diferença de fase entre o sinal recebido e o sinal estimado.

Pode-se ver ainda que o ajuste de fase é calculado de forma recursiva da seguinte maneira:

$$\text{phi}[n+1] = \text{phi}[n] + \text{step} * e[n] \quad (6)$$

onde phi é o ajuste de fase calculado e step é o parâmetro de correção ou *step-size parameter*.

Para efeitos de sincronização, foi definido que os dados seriam enviados em frames, sendo cada frame composto por 156 bits divididos da seguinte maneira:

- 20 bits em zero para sincronização da fase;

- UW: 8 bits com a sequência s3 s1 s1 s3 para correção de ambigüidade;
- Dados: 128 bits de dados.

A UW ou *unique word* é necessária devido a uma ambigüidade de 90 graus inerente ao algoritmo de sincronização. Em outras palavras, o algoritmo pode convergir para uma fase errada com offset de 90, 180 ou -90 graus. A unique word é utilizada da seguinte maneira:

- Se $-45^\circ < \text{diferença de fase} < 45^\circ$, então o algoritmo produz uma estimativa de fase que está sincronizada com a fase da portadora do transmissor. A UW é recebida corretamente como s3 s1 s1 s3;
- Se $45^\circ < \text{diferença de fase} < 135^\circ$, então o algoritmo produz uma estimativa de fase com uma diferença de $+90^\circ$ em relação a fase da portadora do transmissor. A constelação é rotacionada de $+90^\circ$ e a UW é recebida como s0 s2 s2 s0;
- Se $135^\circ < \text{diferença de fase} < 225^\circ$, então o algoritmo produz uma estimativa de fase com uma diferença de $+180^\circ$ em relação a fase da portadora do transmissor. A constelação é rotacionada de $+180^\circ$ e a UW é recebida como s1 s3 s3 s1;
- Se $225^\circ < \text{diferença de fase} < 315^\circ$, então o algoritmo produz uma estimativa de fase com uma diferença de -90° em relação a fase da portadora do transmissor. A constelação é rotacionada de -90° e a UW é recebida como s2 s0 s0 s2.

Dessa forma, basta verificar a UW após o seu recebimento e, de acordo com o valor recebido, rotacionar o sinal pela fase correspondente.

O número de ciclos das portadoras por símbolo foi definido como 4 e o número de amostras por ciclo da portadora como 10. Por exemplo, para uma taxa de transmissão efetiva de 64 Kbits/s tem-se uma frequência de amostragem de $(78K/2) \cdot 4 \cdot 10 = 1.56$ Msamples/s. O parâmetro de correção do algoritmo de sincronização foi fixado em 0.5, já que valores menores não atingiam a sincronização no tempo esperado e valores maiores inseriam muito erro após a convergência.

3. Simulações e Resultados Obtidos

A fim de se obter uma avaliação da taxa de erros, o sistema descrito acima foi simulado através da implementação de scripts do Matlab. Adicionou-se ruído gaussiano branco ao sinal gerado pelo transmissor para simular um canal AWGN (*Additive White Gaussian Noise*), utilizando-se para isso a função "awgn" do pacote "Communications Toolbox" do Matlab. Foram feitas simulações variando-se a relação sinal-ruído entre os valores lineares 0.1, 0.2, 0.5, 1 e 10. Variou-se ainda a precisão numérica do sinal na entrada do receptor entre os valores 1, 2, 4 e 8 bits. Para a obtenção de uma média da taxa de erros, foram enviados 100 frames (12800 bits) para cada caso citado acima, sendo que os 128 bits de dados da cada frame foram preenchidos com os caracteres ASCII da string "hello dsp world!". Os resultados das simulações podem ser vistos na tabela 1.

Precisão numérica (em bits)	Relação sinal-ruído (valores lineares) / Número de erros a cada 100 bits				
	0.1	0.2	0.5	1	10
1	51.00	41.07	25.60	3.39	0
2	44.01	28.77	6.13	0.05	0
4	41.59	22.42	1.41	0	0
8	38.76	21.10	0.94	0	0

Tabela 1. Variação do número de erros com a precisão numérica

A partir da interpretação da tabela, vê-se que nenhuma precisão numérica alcançou uma taxa de erros satisfatória para relações sinal-ruído menores ou iguais a 0.5, ou seja, quando o ruído possui potência duas vezes maior que a potência do sinal. Vê-se também que, com uma precisão de apenas 1 bit por amostra, conseguiu-se obter nenhum erro na transmissão dos 12800 bits desde que a potência do sinal seja dez vezes maior que a do ruído. Além disso, com precisões numéricas de 4 e 8 bits por amostra, obteve-se taxas de erros baixas até em situações em que a relação sinal ruído é igual a um, ou seja, quando a potência do ruído se iguala a potência do sinal.

4. Conclusões

Foi apresentada uma proposta de implementação de um *software radio* utilizando modulação QPSK e um algoritmo baseado no loop de Costas recursivo para sincronização da fase. Foi também estimado, através de simulações, a imunidade a ruído do sistema proposto de acordo com variações da precisão numérica do conversor AD utilizado na entrada do receptor.

Mostrou-se que, com conversores com menos que 8 bits de precisão, o sistema proposto não alcança taxas de erro satisfatórias para canais com menos de 0.5 de relação sinal-ruído. Mostrou-se também que, para conversores de precisão de apenas 1 bit, foi possível receber o sinal com baixas taxas de erro em canais com uma relação sinal-ruído de 10. Por último, viu-se que conversores com até 4 bits de precisão garantem taxas de erro baixas para canais que não introduzam um ruído com potência maior que a do sinal.

5. Referências

- [1] Simon, Marvin K. (1994) "Digital Communication Techniques", Prentice Hall Inc.
- [2] Neto, R. Sampaio, Fortes, J. M. P., Pinho, M. S. (1995) "Comparação entre Técnicas para Detecção Coerente de Portadoras PSK em Comunicações Móveis", 13º Simpósio Brasileiro de Telecomunicações.
- [3] Haykin, Simon (2001) "Communication Systems", 4ª ed., John Wiley & Sons Ltd.

Modelo de arquitetura para simulação de redes móveis sem fio ad hoc no Simmcast

Daniela Saccol Peranconi, Hisham H. Muhammad, Marinho P. Barcellos

¹PIPCA- Programa de Pós-Graduação em Computação Aplicada
Centro de Ciências Exatas e Tecnológicas
Unisinos - Universidade do Vale do Rio dos Sinos
Av. Unisinos, 950 - São Leopoldo, RS - CEP93022-000

{danielap,hisham,marinho}@exatas.unisinos.br

Abstract. *Research in ad hoc mobile wireless networking presents a series of challenges, caused by factors such as mobility, restrictions imposed by portability, vulnerability and wireless instability and security. Simulation is an important tool in this area. This paper describes the main mobile wireless network simulators and their uses, and based on them, proposes an architectural model for simulation of ad hoc networks for the Simmcast simulation framework.*

Resumo. *A pesquisa na área de redes móveis ad hoc sem fio apresenta uma série de desafios, causada por fatores como mobilidade, restrições impostas pela portabilidade, a vulnerabilidade e instabilidade do meio e segurança. Simulação é uma importante ferramenta nesta área. O presente artigo traça um panorama dos principais simuladores de redes móveis sem fio e seus usos, e a partir destes, propõe um modelo de arquitetura de simulação de redes ad hoc para o framework de simulação Simmcast.*

1 Introdução

A popularização de dispositivos portáteis, tais como telefones celulares e palmtops, e a ascensão das telecomunicações, principalmente de redes celulares, foram alguns dos motivos que alavancaram o crescimento da utilização de redes sem fio. Neste tipo de rede não há ligação física entre os dispositivos envolvidos na comunicação, a qual é feita através de ondas eletromagnéticas que trafegam pelo espaço [16].

As redes sem fio podem ser classificadas em redes com ou sem infra-estrutura. Em redes infra-estruturadas a comunicação dos dispositivos móveis é realizada com um ou mais equipamentos centralizadores (pontos de acesso), não havendo comunicação direta entre dois dispositivos, sempre usando-se um ponto de acesso como intermediário [1]. Por outro lado, as redes sem infra-estrutura, também denominadas redes *ad hoc*, são formadas por dispositivos que formam uma rede de forma cooperativa, sendo capazes de estabelecer uma comunicação direta com os dispositivos que estiverem ao seu alcance [1]. Neste tipo de rede não há uma administração centralizada e cada dispositivo pode tanto funcionar como estação ou roteador [6].

Apesar das vantagens de se utilizar comunicação sem fio, existem desafios intrínsecos a este tipo de ambiente que propiciam um vasto campo para pesquisa. Diferentemente do que ocorre em redes fixas, fatores como mobilidade, portabilidade e comunicação sem fio [7] influenciam no projeto de redes móveis sem fio *ad hoc*.

A habilidade dos dispositivos trocarem sua localização enquanto estão conectados à rede aumenta a volatilidade de algumas informações. Uma rede *ad hoc* é dinâmica, uma

vez que tanto sua topologia quanto os membros que a compõem mudam com frequência [19]. Quando dispositivos deixam a rede ou passam a fazer parte dela, novas rotas devem ser encontradas para que a comunicação entre os dispositivos seja mantida [20]. Além disso, com a mobilidade dos dispositivos, a disponibilidade de serviços não pode ser garantida [7].

A portabilidade traz consigo limitações quanto à capacidade de armazenamento dos dispositivos e, principalmente, quanto ao consumo de energia. O consumo de energia em um ambiente de computação sem fio ocorre não somente enquanto o dispositivo está enviando ou recebendo dados, mas também enquanto este encontra-se inativo, havendo necessidade de boas políticas de gerenciamento do consumo de energia por parte destes dispositivos [7].

Em relação às características dos meios de comunicação sem fio, os maiores desafios encontram-se no gerenciamento de banda. Além da disponibilidade de largura de banda já ser baixa, ainda há necessidade de compartilhamento entre os dispositivos móveis, os quais podem entrar ou sair de uma rede a qualquer momento, fazendo com que ocorra alta variação na largura de banda disponível [7]. As desconexões ocorrem mais frequentemente do que em redes fixas, pois barreiras naturais, como construções ou acidentes geográficos, podem interferir na transmissão do sinal. Isso faz com que dispositivos deixem a rede a qualquer instante, sendo necessária uma reparação de rotas para manutenção da comunicação [20].

Talvez a questão mais problemática em redes móveis sem fio *ad hoc* seja a de segurança. A ausência de entidades centralizadoras, as características dos enlaces sem fio, a natureza volátil de tais redes que podem se dividir em um instante e se agrupar novamente de modo imprevisível e o fato de não se ter como prever o tamanho de uma rede *ad hoc* são algumas considerações a serem destacadas no que se refere a segurança em redes *ad hoc* [8].

Devido a este panorama, a área de redes móveis sem fio *ad hoc* apresenta-se como um desafiador campo de pesquisa, e o uso de simulação mostra-se como uma ferramenta especialmente eficaz, pois apresenta vantagens como facilidade no controle de detalhes e fornece a possibilidade de realização de experimentos com hardware e/ou software que ainda não estão disponíveis [20], além do desenvolvimento de cenários que seriam excessivamente complexos para avaliações experimentais.

Este artigo discute as características mais relevantes no desenvolvimento de simuladores de redes móveis sem fio *ad hoc*, a partir de um estudo sobre alguns simuladores de redes móveis sem fio (Seção 2), destacando as métricas e premissas assumidas (Seção 3). A partir destas características, um modelo de arquitetura para simulação de redes móveis sem fio *ad hoc* é proposto, utilizando o framework de simulação Simmcast (Seção 4).

2 Principais simuladores de redes móveis sem fio

Apesar dos avanços atingidos na área de redes móveis sem fio *ad hoc* nos últimos anos, a maior parte das pesquisas em simulações de rede ainda concentra-se em redes fixas. Alguns dos simuladores de ambientes de computação móvel mais utilizados são ns, GloMoSim e MobiCS. Além destes simuladores, existem alguns ambientes de testes, tais como WiPPET [10] e SWiMNet [3], que podem ser citados.

Ns [4] é um simulador de protocolos de rede que suporta simulações de protocolos organizados em camadas, que tem por objetivo o estudo de escala e interação de protocolos. O simulador ns foi usado originalmente para o estudo do desempenho do protocolo

TCP, e desde então tem sido estendido de modo a suportar trabalhos em várias áreas. Em [5] são descritas as extensões adicionadas aos ns para comunicação sem fio.

GloMoSim [17] (Global Mobile System Simulator) é um simulador paralelo para redes móveis baseado em uma biblioteca modular, implementada usando a linguagem de simulação PARSEC. GloMoSim é um simulador específico para a simulação de redes de computação móvel de larga escala.

MobiCS [14, 13] (Mobile Computing Simulator) é um simulador distribuído de eventos discretos para computação móvel. Dois modos de simulação são disponibilizados ao usuário: modo determinístico e modo estocástico. No modo determinístico o MobiCS atua como uma ferramenta para teste e avaliação da correção de protocolos distribuídos. Já no modo estocástico, um protocolo distribuído é submetido a contínuas simulações, objetivando avaliar o desempenho do protocolo em um cenário mais realístico. Neste modo, o comportamento dos elementos da rede é randômico ou baseado em parâmetros probabilísticos que definem o cenário de simulação.

3 Exemplos de uso de simuladores – métricas e premissas

Extensiva pesquisa tem sido realizada acerca de protocolos na área de redes móveis sem fio *ad hoc*, abordando questões relacionadas à segurança, ao consumo de energia, ao roteamento de mensagens entre os dispositivos formadores da rede, entre outras. Nesta seção, alguns destes estudos serão apresentados, de modo a analisar as métricas e premissas empregadas.

Em [15], é proposto um protocolo de roteamento seguro para redes *ad hoc* (ARAN, Authenticated Routing for Ad hoc Networks). Para comparar o desempenho de ARAN com o protocolo de roteamento AODV [11] e validar o protocolo proposto, os autores realizaram avaliações de ambos protocolos utilizando GloMoSim. O seguinte modelo de mobilidade foi aplicado: as posições iniciais dos dispositivos eram aleatórias, e periodicamente, os dispositivos moviam-se para locais selecionados randomicamente com velocidades e tempos de paradas configurados. As métricas de desempenho avaliadas foram (a) fração de pacotes de dados gerados pelos transmissores e entregues aos devidos destinos, (b) tamanho médio do caminho entre um transmissor e um receptor, (c) atraso médio entre o envio de um pacote de dados e o recebimento do mesmo, (d) atraso médio entre o envio de um pacote de descoberta de rota e o recebimento da primeira resposta correspondente a rota e (e) carga de roteamento, tanto em bytes quanto em número de pacotes.

Outro desafio na área de redes móveis sem fio é o desenvolvimento de políticas de gerenciamento de consumo de energia, uma vez que esta deve ser realizada de forma distribuída e cooperativa. Um framework para gerenciamento de energia sob demanda para redes *ad hoc* é proposto em [18]. Para avaliar a eficiência do framework, um protótipo do mesmo foi implementado no simulador ns e diversas simulações foram conduzidas. A eficiência pode ser avaliada levando-se em conta o tempo durante o qual a rede permaneceu operacional e as transmissões de dados, que deveriam ter poucas perdas e baixa latência. As premissas assumidas foram: os dispositivos estavam randomicamente distribuídos no plano de análise, a rede nunca era particionada, não existiam perdas induzidas por erros, a energia consumida para trocar entre os estados de conservação de energia e ativo não era considerada e todos os pacotes de dados eram do mesmo tamanho.

Outro caso que ilustra a utilização de simuladores pode ser encontrado em [12]. Este artigo propõe um protocolo para multicast atômico entre dispositivos móveis como suporte para comunicação em grupo denominado AM²C (Atomic Multicast Protocol for

Mobile Computing). Para prototipar e simular o AM²C foi utilizado o ambiente MobiCS e algumas premissas foram definidas, sendo elas: a comunicação entre quaisquer duas estações de serviços móveis (MSS) é confiável; as mensagens são entregues na ordem em que são enviadas; as MSSs não falham; a qualquer momento, cada dispositivo móvel no sistema está associado com exatamente uma MSS; se um dispositivo móvel está ativo, ele deve enviar uma confirmação para todas as mensagens recebidas da MSS e enquanto estiver inativo, ele não deve responder qualquer mensagem e um dispositivo móvel somente pode deixar o sistema após enviar todas as confirmações de mensagens que estão pendentes. Foram medidas a percentagem de multicasts cancelados, a duração média de um multicast, o número médio de mensagens por multicast e por dispositivo móvel e o número de mensagens de confirmação adicionais geradas na fase em que todos os dispositivos móveis são informados sobre o estado final do multicast.

4 Simulação de redes *ad hoc* no Simmcast

Simmcast [2] é um framework orientado a objetos de simulação discreta baseado na linguagem de programação Java, construído sobre uma *engine* de simulação de eventos discretos baseada em processos. A interface do framework é definida como uma API que oferece ao usuário tanto operações típicas de comunicação e controle de tempo, quanto um modelo de *threads* cooperativo [9].

Simulações com o Simmcast são desenvolvidas através da extensão das classes do framework por herança, adicionando-se a lógica de protocolo desejada. A configuração de rede a ser avaliada é descrita por um arquivo processado em tempo de execução, sem recompilação. A arquitetura do Simmcast constitui-se de uma série de blocos básicos que descrevem as entidades da simulação: subclasses de *Node* representam os diversos tipos de nodos: estações, roteadores (*HostNode*, *RouterNode*...); objetos da classe *Path* representam fluxos de pacotes entre nodos, tipicamente links de rede. Simmcast oferece ainda primitivas de multicast (classe *Group*) e pacotes de geração de topologias (*TopologyGenerator*), de tráfego (*TrafficGenerator*) e de traços (*TraceGenerator*).

A estrutura modular do Simmcast torna-o propício para uma extensão para pesquisa em redes móveis sem fio *ad hoc*. Para tal, três aspectos devem ser abordados: modelagem da mobilidade, da transmissão sem fio, e de gerenciamento de roteamento descentralizado.

Um framework extensível como o Simmcast deve suportar os diferentes modelos de mobilidade empregados em pesquisa de redes móveis. Entre estes modelos estão o baseado em funções randômicas de localização e mobilidade, e a partir de um modelo geográfico bi- ou tri-dimensional. Isto envolve criar uma nova classe de nodos, *MobileNodes*, que sejam capazes de registrar sua localização no simulador (a fim de cômputo de latências a partir de distâncias) e a variação desta em função do tempo, isto é, sua movimentação.

O modelo abstrato de representação de fluxos de dados (onde links são um caso específico de *Path*) e o suporte nativo de comunicação 1-para-n (através de classes *Group*), permitem a modelagem da transmissão sem fio como uma forma de broadcast, onde um grupo corresponde a uma faixa de frequência. Caminhos no Simmcast têm, por definição, latência dinâmica, definida através de uma classe de geração de números. Esta classe, embora normalmente utilizada para definição de distribuições estatísticas, pode ser utilizada para estimação da latência da transmissão de rádio em função da localização dos nodos.

O gerenciamento de roteamento descentralizado pode ser representado no Simmcast de uma forma bastante realística, através da separação dos protocolos de roteamento e aplicação em diferentes *threads* conectadas aos nodos em tempo de execução, permitindo execuções de um experimento com diferentes protocolos de roteamento *ad hoc*, definindo em `MobileNode` a mesma flexibilidade que hoje ocorre com protocolos de roteamento convencionais em `RouterNode`, onde diferentes subclasses de `RoutingAlgorithmStrategy` podem ser conectadas.

5 Considerações finais

Este artigo apresentou um modelo de arquitetura de suporte a simulações de redes móveis sem fio *ad hoc* para o Simmcast, a partir da análise de requisitos dos trabalhos de verificação e validação de resultados de protocolos em redes móveis sem fio *ad hoc* e dos principais simuladores usados nesta área.

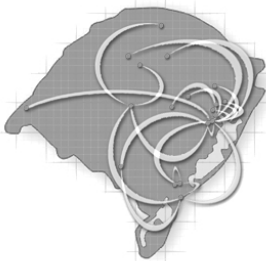
A área de redes móveis sem fio *ad hoc*, apesar dos avanços atingidos nos últimos anos, ainda requer muita pesquisa. Questões como o desenvolvimento de protocolos de roteamento seguros, políticas de gerenciamento eficientes para o controle do consumo de energia dos dispositivos móveis, mecanismos eficientes para descoberta de rotas, dentre outras, ainda representam desafios aos pesquisadores em redes móveis sem fio *ad hoc*.

A partir do momento que uma nova proposta para resolução ou minimização de tais desafios é apresentada, a simulação surge como técnica auxiliar para comprovar a eficiência e desempenho destas propostas. No entanto, nem todos os detalhes relacionados ao contexto real que se deseja simular podem ser assumidos. Se isso ocorresse, os modelos se tornariam tão complexos quanto os sistemas reais. É preciso, ao projetar uma ferramenta para estudo de redes móveis sem fio *ad hoc*, conhecer as premissas utilizadas com o intuito de simplificar a simulação, mas sempre buscando manter o modelo o mais próximo possível da realidade. Este trabalho, assim, combinou um estudo dos trabalhos da área, onde exemplos significativos foram apresentados com maior detalhe, com uma contextualização destas premissas e métricas ao simulador Simmcast, apontando claramente como trabalho futuro a implementação desta arquitetura proposta.

Referências

- [1] Anton, E. R. and Duarte, O. C. M. B. (2002) "Segurança em Redes Sem Fio Ad Hoc: Gerenciamento de Chave de Grupo", XIV Congresso Brasileiro de Automática, Natal, RN, Setembro.
- [2] Barcellos, M., Muhammad, H. and Detsch, A. (2001) "Simmcast: a Simulation Tool for Multicast Protocol Evaluation", XIX Simpósio Brasileiro de Redes de Computadores (SBRC 2001), Maio.
- [3] Boukerche, A., Das, S. K. and Fabbri, A. (2001) "SwiMNet: A Scalable Parallel Simulation Testbed for Wireless and Mobile Networks", *Wireless Networks*, v. 7, pp. 467-486.
- [4] Breslau, L., Estrin, D., Fall, K., Floyd, S., Heidemann, J., Helmy, A., Huang, P., McCanne, S., Varadhan, K., Xu, Y. and Yu, H. (2000) "Advances in Network Simulation", *IEEE Computer*, vol. 33, p. 59-67, Maio.
- [5] The CMU Monarch Project. (1998) "The CMU Monarch Project's Wireless and Mobility Extensions to NS," Agosto. Disponível em <http://www.monarch.cs.cmu.edu/>.
- [6] Cunha, D. de O., Costa, L. H. M. K. and Duarte, O. C. M. B. (2003) "Um Mecanismo de Roteamento para o Consumo Balanceado de Energia em Redes Móveis Ad Hoc", XXI Simpósio Brasileiro de Redes de Computadores (SBRC 2003), Natal, RN, Maio.

- [7] Forman, G. H. and Zahorjan, J. (1994) "The Challenges of Mobile Computing", IEEE Computer, vol. 27, No. 4, p. 38-47, April.
- [8] Martucci, L. A., Carvalho, T. C. M. B. and Ruggiero, W. V. (2003) "Domínios Virtuais para Redes Móveis Ad Hoc", XXI Simpósio Brasileiro de Redes de Computadores (SBRC 2003), Natal, RN, p. 599-614, Maio.
- [9] Muhammad, H. H. and Barcellos, M. P. (2002) "Simulation Group Communication Protocols Through an Object-Oriented Framework", in Proceedings 35th SCS Annual Simulation Symposium, San Diego, April.
- [10] Panchal, J., Kelly, O., Lai, J., Mandayam, N., Ogielski, A. T. and Yates, R. (1998) "WiPPET, A Virtual Testbed for Parallel Simulations of Wireless Networks", 12th Workshop on Parallel and Distributed Simulation (PADS'98), Canadá, p. 162-169, May.
- [11] Perkins, C. E. and Royer, E. M. (1999) "Ad hoc on-demand distance vector routing", 2th IEEE Workshop on Mobile Computer Systems and Applications, New Orleans, February.
- [12] Ribeiro, M. de F., Endler, M. (2002) "Design and Evaluation of a Protocol for Atomic Multicast Wireless Mobile Hosts", IV Workshop de Comunicação sem Fio e Computação Móvel (WCSF2002), São Paulo, p.3-14, Outubro.
- [13] Rocha, R. C. A. da and Endler, M. (2001) "MobiCS: An Environment for Prototyping and Simulating Distributed Protocols for Mobile Networks", 3rd. IEEE International Conference in Mobile and Wireless Communications Networks (MWCN'2001), Recife, Brasil, p. 44-51, August.
- [14] Rocha, R. C. A. da and Endler, M. (2000) "Flexible Simulation of Distributed Protocols for Mobile Computing", 3rd. Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM), Boston, p. 123-126, August.
- [15] Sansgiri, K., Dahill, B., Lavine, B. N., Shields, C. and Royer, E. M. B. (2002) "A Secure Routing Protocol for Ad Hoc Networks", 10th IEEE International Conference on Network Protocols (ICNP'02), Paris, November.
- [16] Santos, A. S. (2003) "Estratégias de Hand-off com Balanceamento de Carga para Computação Móvel", Dissertação de Mestrado, Universidade de São Paulo, Instituto de Matemática e Estatística, Março.
- [17] Zeng, X., Bagrodia, R. and Gerla, M. (1998) "GloMoSim: A Library for parallel simulation of large-scale wireless networks", In Proceedings of the 12th Workshop on Parallel and Distributed Simulation (PADS - 98), p. 154-161, May.
- [18] Zheng, R. and Kravets, R. (2003) "On-demand Power Management for Ad Hoc Networks", In Proceedings IEEE INFOCOM 2003, June.
- [19] Zhou, L. and Haas, Z. J. (1999) "Securing Ad Hoc Networks", IEEE Networks Special Issue on Network Security, November/December.
- [20] Westin, O. (2003) "Performance issues in ad hoc networks", <http://www.nada.kth.se/~owe/exjobb/littstudy.pdf>, Agosto.



Sessão Técnica 6

Tolerância a Falhas

Tolerância a Falhas em Sistemas de Agentes Móveis

Tiago Fioreze, Ingrid Jansch-Pôrto, Lisandro Zambenedetti Granville

Instituto de Informática – Universidade Federal do Rio Grande do Sul
Caixa Postal 15064 – 90501-970 Porto Alegre, RS

{tfioreze, ingrid, granville}@inf.ufrgs.br

Abstract. *Mobile agents belong to an important area of distributed processing. Fault tolerance techniques are required for robust and reliable applications based on mobile agents. The main purpose of this paper is to present fault tolerant approaches used with mobile agents. We start with the basic operations of the mobile agents, followed by some of the main problems faced by applications based on mobile agents. Then, some of the main techniques of fault tolerance in applications based on mobile agents are explained and discussed.*

Resumo. *Agentes móveis fazem parte de uma importante área no campo de processamento distribuído. Técnicas de tolerância a falhas são fundamentais para aplicações robustas e confiáveis baseadas em agentes móveis. O objetivo principal deste artigo é apresentar métodos de tolerância a falhas usados em sistemas de agentes móveis. Este artigo apresenta inicialmente o funcionamento básico dos agentes móveis, seguido de alguns dos principais problemas enfrentados por aplicações baseadas em agentes móveis. Posteriormente, algumas das principais técnicas de tolerância a falhas em aplicações baseadas em agentes móveis serão explicadas e discutidas.*

1. Introdução

Agentes móveis são programas que não estão ligados exclusivamente ao sistema que tenha iniciado sua execução. Eles têm a liberdade de viajar entre computadores pertencentes a uma rede. Criados em um ambiente de execução, eles podem transportar seus estados e códigos para qualquer outro ambiente de execução, onde eles podem voltar a executar [Lange and Oshima, 1998]. Falhas nesses ambientes de execução podem levar para uma perda parcial ou, até mesmo, completa de um agente. Como agentes móveis podem executar em pequenas e grandes redes (e.g. Internet), é plausível que eles enfrentem problemas relativos a atrasos, falhas de comunicação e demais problemas existentes em redes de computadores. Devido a isso, é difícil ao proprietário do agente saber quando um agente criado foi perdido ou se a execução do agente está atrasada devido a problemas no meio de comunicação. Essas incertezas podem originar as seguintes situações:

- o proprietário do agente pode acreditar que o agente criado foi perdido, quando na verdade ele não foi. O proprietário, acreditando nisso, poderia recriar o agente perdido e lançá-lo novamente na rede, o que poderia implicar em execuções simultâneas do mesmo agente em determinados dispositivos.

- O proprietário do agente fica aguardando pelo término da execução, mas o agente foi perdido. Isso caracteriza uma situação bloqueante, uma vez que o proprietário fica bloqueado aguardando a execução do seu agente.

Tolerância a falhas em sistemas de agentes móveis visa eliminar, ou pelo menos minimizar, essas incertezas, seja através da garantia de que o agente chegue ao seu destino ou, no mínimo, através da notificação ao proprietário do agente de problemas em potencial [Pleisch and Schiper, 2000]. Existem várias técnicas de tolerância a falhas aplicadas em sistemas de agentes móveis. Neste artigo, serão apresentadas algumas das técnicas mais conhecidas e relatadas no meio científico. Este artigo não visa indicar qual técnica é a melhor, mas sim expor as características e o funcionamento das técnicas estudadas. As técnicas abordadas neste artigo são: as técnicas baseada em *checkpointing*, em replicação e em replicação com votação.

O restante deste artigo está organizado da seguinte forma. A seção 2 apresentará o funcionamento dos agentes móveis e quais são os tipos de falhas que podem acometê-los. A seção 3 mostrará algumas das técnicas de tolerância a falhas em sistemas de agentes móveis mais conhecidas. Finalmente, a seção 4 conclui este artigo.

2. Agentes Móveis

Basicamente, agentes móveis são programas autônomos de computador que viajam através de um rede de computadores heterogêneos. A figura 1 ilustra diferentes estágios (k) da execução de um agente móvel (a_i) em uma seqüência de máquinas (p_j).



Figura 1: Funcionamento básico de um agente móvel.

Para um melhor entendimento referente ao funcionamento dos agentes móveis, alguns conceitos são importantes.

- **Agente:** Um processo autônomo que viaja entre diferentes computadores de uma rede realizando procedimentos para o qual foi instruído e que é capaz de migrar seu código de uma máquina a outra de forma independente.
- **Lugar:** Refere-se ao ambiente de execução para um agente móvel arbitrário. Este ambiente deve fornecer recursos necessários à execução do agente móvel.
- **Estágio:** Refere-se à fase da execução de um agente móvel arbitrário, em uma localização específica.

Infelizmente não existe *software* ou *hardware* que seja totalmente imune a falhas. Com isso, qualquer ambiente de execução é potencialmente sujeito a falhas. Existem três tipos de defeitos que podem afetar um agente. São eles:

- **Colapso do próprio agente:** É a perda completa do agente móvel. Esse tipo de defeito geralmente não causa o colapso do ambiente de execução ou do computador no qual o agente móvel está realizando a execução de tarefas.

- **Colapso no ambiente de execução:** Um agente móvel precisa de um ambiente propício a sua execução. Se tal ambiente de execução entrar em colapso, por consequência o agente móvel sofrerá o mesmo problema (perda total).
- **Colapso no computador:** Caracteriza-se pela incapacidade do computador se comunicar com os demais computadores de uma rede e também de realizar processamento de tarefas. O colapso no computador leva, por consequência, ao colapso do ambiente de execução e do agente móvel.

Salienta-se que, embora dificilmente o colapso repercuta diretamente nos níveis superiores, uma execução mal-sucedida de um agente pode levar o ambiente de execução ou o computador no qual ele estava executando para um estado inconsistente. Logo, necessita-se o uso de mecanismos que desfaçam ações impróprias realizadas por agentes, de modo que o computador ou o ambiente de execução volte a um estado consistente.

3. Técnicas de Tolerância a Falhas em Sistemas de Agentes Móveis

Agentes móveis estão sendo explorados em várias áreas, tais como: comércio eletrônico, gerenciamento de redes e computação distribuída. Apesar disso, eles serão somente usados em larga escala se alguns importantes problemas (tolerância a falhas e segurança) puderem ser resolvidos de um modo eficaz [Silva et al., 2000]. O que será visto nesta seção são algumas técnicas de tolerância a falhas em sistemas de agentes móveis.

3.1. Técnica baseada em *Checkpointing*

Uma das técnicas usadas na tolerância a falhas em sistemas de agentes móveis é baseada em *checkpointing*. Existem diferentes esquemas de *checkpointing* em agentes móveis [Yeom et al., 2002]. Este artigo não levará em conta as particularidades de cada um, mas sim a idéia básica do *checkpointing* em agentes móveis que é o armazenamento estável dos códigos e estados de um agente móvel arbitrário em um determinado computador.

A técnica baseada em *checkpointing* pode ser explicada informalmente como segue. Quando um agente móvel migra de um computador a outro, levando consigo seu estado atual e seu código, o computador destino armazena essas informações em um local seguro (e.g. sistema de disco) antes de iniciar a execução do agente. Com isso, o agente não é perdido caso o computador destino falhe, já que existem informações armazenadas no computador que permitem ao agente ser executado novamente após a recuperação do computador destino.

Entretanto, enquanto o computador destino estiver desligado, o procedimento de execução do agente ficará bloqueado, e por consequência, o processo no computador que criou o agente (fonte do agente) ficará bloqueado esperando o fim da execução da agente, ou seja, esta técnica de *checkpointing* é bloqueante.

3.2. Técnica baseada em Replicação

Diferentemente da técnica vista na seção 3.1, a técnica baseada em replicação de agentes [Pleisch and Schiper, 2000] evita que a execução de um agente móvel seja bloqueada através da criação de várias cópias do mesmo agente em um determinado estágio. Ao invés de enviar um agente para um único lugar em um determinado estágio, o agente é copiado para um grupo M_i de lugares $\{p_i^0, p_i^1, p_i^2, \dots\}$. Para prevenir que o colapso de uma

máquina afete múltiplos lugares em um estágio S_i , cada lugar p_i^j ($j=0,1,\dots$) é normalmente localizado em computadores diferentes. Se um determinado lugar falhar, a execução do agente em um estágio S_i continua em outro lugar. Por exemplo, a figura 2 mostra um defeito no lugar p_2^0 , mas a execução do agente continua, uma vez que os lugares p_2^1 e p_2^2 estão aptos a executarem o agente.

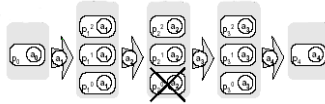


Figura 2: Exemplo de falha no funcionamento de um agente móvel.

Embora esta técnica resolva o problema de bloqueio na execução de um agente móvel, ela é propensa a violar a propriedade dos agentes móveis executarem somente uma vez (*exactly-once property*) em determinado estágio. Uma falsa suposição que o lugar p_2^0 tenha falhado pode levar a execução do agente a_2 no lugar p_2^1 , por exemplo, enquanto o mesmo agente continua sendo executado em p_2^0 . Isso leva para múltiplas execuções de operações de um agente móvel.

3.3. Técnica baseada em Replicação com Votação

No método de replicação baseado em votação [Rothermel and Strasser, 1998], as ações de envio de um agente para determinado lugar, da execução desse agente nesse lugar e do recebimento de um agente em um próximo lugar fazem parte de uma única transação.

Similar ao método de replicação, a técnica de replicação baseada em votação mantém um grupo M_i de lugares $\{p_i^0, p_i^1, p_i^2, \dots\}$ associados a um estágio S_i . A diferença é que, nessa técnica, cada lugar tem uma propriedade associada a ele, o que define uma ordem entre os lugares pertencentes ao mesmo estágio. Cada estágio inicialmente seleciona o lugar com a maior prioridade sendo este denominado de **trabalhador**. Os demais lugares são denominados de **observadores** e têm como função monitorar a disponibilidade do trabalhador. Quando os observadores notam que o trabalhador entrou em colapso, eles selecionam um novo trabalhador entre eles mesmos, através de um protocolo de eleição.

Para preservar a propriedade dos agentes móveis executarem somente uma vez em determinado estágio, um processo de votação foi integrado a um protocolo de *commit* de duas fases [Gray and Reuter, 1994]. Com essa integração, um trabalhador pode somente finalizar uma transação se a grande maioria dos lugares envolvidos na execução de um agente móvel concordarem.

A arquitetura projetada por Rothermel e Straber (1998) consiste de um **gerenciador de transação** que executa o protocolo 2PC (*commit* de 2 fases) e **gerenciadores de recursos** que mantêm dados recuperáveis. Nessa arquitetura, o gerenciador de transação do trabalhador interage com um outro tipo de gerenciador de recursos, denominado **orquestrador**. O orquestrador, que se comunica com os **eleitores** pertencentes ao seu estágio de execução, é responsável pela organização da votação.

3.3.1. Protocolo de Votação

O protocolo de votação é executado entre o orquestrador e o eleitor de um estágio S_i . Primeiramente, o orquestrador envia uma requisição chamada de VOTE (voto) para cada eleitor do seu estágio de execução. O formato da requisição é o seguinte: VOTE (StageId, Tid, OrchId). Nessa requisição estão incluídos o ID (identificador) do estágio atualmente sendo processado (StageId), o ID do orquestrador do processo de votação (OrchId) e o ID da transação que o orquestrador está manipulando (Tid). Ao final do envio das requisições aos eleitores, o orquestrador espera por respostas dos eleitores.

Cada eleitor possui um *buffer* denominado OrchSet, utilizado para armazenar o ID dos orquestradores que fizeram a requisição VOTE. Um eleitor determina seu voto ao orquestrador baseado no conteúdo do OrchSet. As seguintes situações são possíveis:

1. Se OrchSet estiver vazio, significa que o eleitor não votou ainda. Nesse caso, o ID do orquestrador é armazenado no OrchSet e uma resposta YES (StageId, Tid, VoterId) é enviada de volta ao orquestrador. O parâmetro VoterId identifica o eleitor votante.
2. Se OrchSet não estiver vazio, significa que existe mais de um orquestrador competindo pelo voto. Assume-se que L seja o lugar com maior prioridade em OrchSet. Se OrchSet não estiver vazio e OrchId possuir prioridade inferior a L , então o eleitor já votou YES para um lugar com maior prioridade. Nesse caso, o eleitor retorna NO (StageId, Tid, VoterId) ao orquestrador.
3. Se OrchSet não estiver vazio e o OrchSet possuir prioridade maior que L , significa que o eleitor já votou para um lugar com uma prioridade menor. O eleitor então envia ao orquestrador um sim 'condicional' denominado COND_YES (StageId, Tid, OrchSet, VoterId) e adiciona OrchId em OrchSet. Esse voto significa que o eleitor vota YES, contanto que todos os lugares armazenados em OrchSet também votem YES.

Depois de analisar todos os COND_YES e YES dos eleitores e verificar que eles constituem a maioria dos votos, o orquestrador retorna a mensagem *rm_yes* para o gerenciador de transações local informando que a votação transcorreu normalmente.

Na fase 2, se o gerenciador de transações finalizar a transação, ele envia a mensagem *rm_commit* para cada gerenciador de recursos local. Se o gerenciador de transações abortar a transação, ele envia a mensagem *rm_abort* para o orquestrador. Esse envia uma requisição denominada UN_VOTE para todos os eleitores que participaram da votação organizada por ele e que votaram YES ou COND_YES. Após isso, a transação abortada é reiniciada. Os eleitores, ao receberem a requisição UN_VOTE, removem o OrchId em OrchSet, permitindo que um lugar de menor prioridade seja prioritário caso algum lugar com maior prioridade falhe.

3.3.2. Protocolo de Detecção de Defeitos

O trabalhador de um estágio periodicamente envia mensagens I_AM_ALIVE para os observadores. Se um observador não receber as mensagens I_AM_ALIVE por um período de tempo pré-determinado, ele supõe que o trabalhador falhou e propõe uma eleição entre os

demais observadores para selecionar o novo trabalhador. Um observador envia uma mensagem `ARE_YOU_THERE` para todos os observadores com a prioridade maior. Os observadores que estiverem ativos responderão com uma mensagem `I_AM_THERE`. Se nenhuma mensagem chegar dentro de um tempo razoável, o observador que iniciou a eleição será o novo trabalhador. O novo trabalhador envia então a mensagem `I_AM_SELECTED` para os demais e a partir daí a execução de um agente móvel prossegue. Os demais participantes da eleição, ao receberem a mensagem `I_AM_SELECTED` realizam o trabalho de monitoramento do novo trabalhador.

Com esses protocolos, o método de replicação baseada em votação consegue evitar que uma execução de um agente móvel fique bloqueado, além de preservar a propriedade dos agentes móveis executarem somente uma vez em determinado estágio através do protocolo de votação.

4. Conclusão

Neste artigo foi apresentado o funcionamento básico dos agentes móveis e como falhas que ocorrem durante a execução dos mesmos podem influenciar de forma negativa um sistema baseado em agentes móveis. Foram apresentadas três técnicas de tolerância a falhas: as técnicas baseadas em *checkpointing*, em replicação e em replicação com votação. A primeira previne a perda de um agente, mas é dita bloqueante. A segunda evita o bloqueio da execução de um agente, mas é propensa a violar a propriedade de execução única em determinado estágio. A última técnica é um método robusto que evita o bloqueio da execução e preserva a propriedade dos agentes móveis executarem somente uma vez em determinado estágio. Finalmente, cabe lembrar que a vinculação a outros serviços tais como os de detecção de defeitos é extremamente importante na solução adequada das técnicas aqui apresentadas.

Referências

- Gray, J. and Reuter, A. (1994). *Transaction Processing - Concepts and Techniques*. Morgan Kaufmann.
- Lange, D. B. and Oshima, M. (1998). *Programming and Deploying Java Mobile Agents with Aglets*. Addison-Wesley.
- Pleisch, S. and Schiper, A. (2000). Modeling fault-tolerant mobile agent execution as a sequence of agreement problems. In *Proceedings of the 19th IEEE Symposium on Reliable Distributed Systems (SRDS)*, pages 11–20.
- Rothermel, K. and Strasser, M. (1998). A fault-tolerant protocol for providing the exactly-once property of mobile agents. In *Proceedings of the 17th IEEE Symposium on Reliable Distributed Systems (SRDS)*, pages 100–108.
- Silva, L. M., Batista, V., and Silva, J. G. (2000). Fault-tolerant execution of mobile agents. In *Proceedings of the International Conference on Dependable Systems and Networks (DSN)*, pages 135–143.
- Yeom, H. Y., Kim, H., Park, T., and H. Park (2002). The cost of checkpointing, logging and recovery for the mobile agent systems. In *Proceedings of the 2002 Pacific Rim International Symposium on Dependable Computing (PRDC'02)*, pages 45–48.

Alta Disponibilidade aplicada a Computação Móvel

Hélio Antônio Miranda da Silva , Ingrid Jansch-Pôrto

¹Instituto de Informática – Programa de Pós-Graduação em Ciência da Computação
Universidade Federal do Rio Grande do Sul – UFRGS
Caixa Postal 15064 – 91501-970 Porto Alegre, RS

hamsilva@inf.ufrgs.br, ingrid@inf.ufrgs.br

Abstract. *This paper argues on the attainment of fault tolerance in distributed mobile environments. Mobile computation in distributed environments requires the use of techniques that add dependability to these applications. However, achieving fault tolerance in these environments is still a challenge, due to the characteristics of the network connections, as the their mobility, the low bandwidths and the high fault rates. In this paper, mechanisms for recovery using checkpoints are considered to achieve fault tolerance.*

Resumo. *Este artigo discute a obtenção de tolerância a falhas em ambientes móveis distribuídos. A computação móvel em ambientes distribuídos requer o uso de técnicas que adicionem disponibilidade à execução de aplicações. Contudo, a obtenção de tolerância a falhas nesses ambientes ainda é um desafio, devido às características de conectividade dessas redes, como a mobilidade, as baixas larguras de banda e as altas taxas de falhas. Neste artigo, é proposta a utilização de recuperação por checkpoints para obter tolerância a falhas.*

1. Introdução

A computação móvel oferece uma nova perspectiva no desenvolvimento de aplicações distribuídas, onde a dinâmica da mobilidade permite que os *hosts*¹ móveis se comuniquem de qualquer lugar a qualquer instante, mesmo estando em movimento. E assim como em redes estáticas, se faz necessário a utilização de técnicas que tornem a execução de aplicação distribuídas mais confiável e segura. Contudo, a obtenção de tolerância a falhas em ambientes móveis ainda é um desafio, devido às características das redes *wireless*, como a dinâmica da mobilidade, as baixas larguras de banda e as altas taxas de falhas. Assim, esquemas tradicionais de tolerância a falhas [Jalote, 1994] não podem ser diretamente aplicados nesses ambientes [Laamanen et al., 1999].

Neste artigo, são focados esquemas que utilizam recuperação através *checkpoints*. Este esquemas baseiam-se na realização, de tempos em tempos, de *checkpoints* globais que são constituídos por um conjunto de *checkpoints* locais. Assim, após uma falha, o sistema pode retornar até um estado consistente e retomar a execução.

O artigo está disposto da seguinte maneira: na seção 2, é definido modelo de sistema móvel que será utilizado. Na seção 3, serão vistas algumas características relacionadas à computação móvel. Na seção 4, é definida a recuperação de erros através de

¹Host: em português significa "hospedeiro".

checkpoints. Nesta seção, também são apresentadas várias estratégias de realização de *checkpoints* em ambientes móveis. E por fim, a seção 5 mostra as conclusões do artigo.

2. Modelo de Sistema Móvel

Um modelo de sistema móvel é formado por *hosts* móveis (HM) ² e também por *hosts* estáticos [Acharya and Badrinath, 1994]. Os *hosts* móveis podem se mover enquanto mantêm ativa a sua conexão. Os *hosts* estáticos são conectados uns aos outros por redes estáticas. Alguns dos *hosts* estáticos possuem uma interface *wireless* e cada um destes é considerado uma *estação de suporte à mobilidade* (ESM) ³, e têm como objetivo fazer a ligação entre a rede *wireless* e a rede estática. Considera-se que todas as ESMs são estáticas e estão interconectadas por redes estáticas. Por causa do limitado alcance das antenas *wireless*, um HM pode se comunicar com uma ESM somente dentro de uma determinada região, e a esse raio de cobertura da ESM é dado o nome de *célula*. Um HM, em dado instante, pode estar dentro de apenas uma célula. A mobilidade permite aos HMs passar de célula em célula de forma transparente. Quando ocorre passagem de um *host* para uma outra célula, a ESM responsável por este *host* também muda. Assim a ESM da nova célula passa a ser a mediadora entre o HM e a rede estática. Este processo de troca de ESM é chamado *handoff* ⁴.

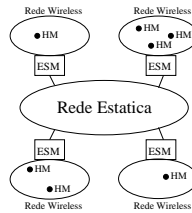


Figura 1: Modelo de Sistema Móvel

3. Aspectos Relacionados à Computação Móvel

Devido às características das redes *wireless*, técnicas tradicionais de obtenção de tolerância a falhas não podem ser diretamente aplicadas. Um dos motivos disso é que muitas destas técnicas foram desenvolvidas tendo-se em mente redes estáticas. Portanto, estes esquemas não prevêm a mobilidade. Características dos ambientes móveis, como a baixa largura de banda, quando comparados com redes estáticas, exigem a minimização da troca de mensagens entre *hosts*, pois altas taxas de comunicação podem causar um grande custo à aplicação distribuída. A probabilidade de avarias em equipamentos aumenta já que os *hosts* são carregados pelos usuários de um lugar para outro. Além disso, falhas transientes como falta de alimentação de energia e desconexões passam a ser frequentes [Neves and Fuchs, 1997].

²No decorrer do artigo, frequentemente, referências a *hosts* móveis serão feitas através da sigla HM.

³No decorrer do artigo, frequentemente, referências as estações de suporte a mobilidade serão feitas através da sigla ESM.

⁴Handoff: em português significa "ato de delegar".

Quanto à utilização de métodos de recuperação por *checkpoints*, surge um novo problema, pois os *checkpoints* não podem ser salvos nos próprios *hosts* como em redes estáticas, já que *hosts* móveis, além de nem sempre terem à disposição recursos como espaço em disco, possuem uma maior probabilidade de falhas [Acharya and Badrinath, 1994]. Por isso, os *checkpoints* devem ser salvos nas ESMs. Contudo, é importante lembrar que se torna necessário adotar critérios na escolha das ESMs onde os *checkpoints* serão armazenados, pois os *hosts*, devido a sua mobilidade, não possuem uma ESM fixa.

4. Estratégias de Recuperação

Na recuperação baseada em *checkpoints*, cada *checkpoint* consiste em um estado da execução da aplicação. Periodicamente cada *host* participante da execução da aplicação distribuída realiza um *checkpoint*, formando dessa forma, um *checkpoint* global que contém o estado da execução da aplicação como um todo. Na ocorrência de falhas, o último *checkpoint* consistente é carregado e a aplicação pode recomençar sua execução a partir desse ponto.

A realização dos *checkpoints* pode ser síncrona ou assíncrona [Acharya and Badrinath, 1994]. Em *checkpoints* síncronos, ocorre coordenação entre todos os participantes para garantir a consistência. Porém, esse esquema apresenta uma sobrecarga significativa gerada pela sincronização entre os processos. Em esquemas assíncronos, cada participante realiza seus *checkpoints* de forma independente dos outros participantes. Nesse, a desvantagem é que um dado conjunto de *checkpoints* locais não necessariamente forma um *checkpoint* global consistente.

Uma estratégia de recuperação por *checkpoints* possui duas componentes: uma estratégia para o salvamento de estado, que determina quando serão realizados os *checkpoints*, e uma estratégia que define onde ficarão armazenados os *checkpoints* quando o *host* trocar de ESM. As estratégias de recuperação por *checkpoint* em *hosts* móveis, definidos por Pradhan et al.(1996), e abordadas a seguir, enquadram-se no conceito de *checkpoints* assíncronos.

4.1. Salvamento de Estado

Conforme visto anteriormente, os *checkpoints* devem ser armazenados em um local seguro. Portanto, o armazenamento nos próprios HM é inadequado, pois, além de estarem mais propensos a falhas do que *hosts* estáticos, os HM podem ter uma capacidade de armazenamento muito limitada. Assim, os *checkpoints* devem ser armazenados na atual ESM do *host*, já que esta, por ser estática, possivelmente terá um maior poder de armazenamento e processamento.

Os *checkpoints* locais são realizados de forma independente para cada HM participante da aplicação. Porém, é necessário determinar com que periodicidade eles devem ser realizados. A estratégia de salvamento de estados pode ocorrer de duas maneiras: sem *logs* ou com *logs*.

4.1.1. Sem registros de logs

Nessa estratégia, cada vez que um evento ocorre, seja o recebimento ou envio de uma mensagem ou uma entrada de dados do usuário, um novo *checkpoint* deve ser realizado. O *checkpoint* é armazenado na ESM corrente. Isto é, um novo salvamento de estado é realizado após ocorrência de qualquer evento que mude o estado do *host*. Após uma falha, quando a *host* móvel reinicia, ele pode buscar o último *checkpoint* realizado que está armazenado na sua atual ESM, diminuindo assim o tempo de recuperação. Porém, é importante lembrar que, a freqüente transmissão de estados através da rede é a desvantagem deste esquema.

4.1.2. Com registros de logs

Nesta estratégia, os *checkpoints* para cada HM são realizados em intervalos regulares de tempo. Os eventos, causadores de mudança de estado, que ocorrerem entre *checkpoints*, serão registrados através de *logs*. Quando um evento de mudança de estado ocorre, antes do processamento deste, é feito o registro de *log* que é enviado para a ESM corrente. Depois de armazenar o *log*, a ESM envia uma mensagem de confirmação para o HM. Os registros de *log* permanecem na ESM até que o próximo *checkpoint* seja realizado. Após uma falha, quando o *host* reinicia, ele pode retomar sua execução a partir do mais recente *checkpoint* e dos registros de *log* armazenados na ESM.

A vantagem desta abordagem é que transferir apenas um registro de *log* é menos oneroso do que transferir um novo *checkpoint*. Com isso, ocorre a diminuição da sobrecarga causada pela troca de informações entre o HM e a ESM. Porém, a desvantagem é que, no caso de falhas, a recuperação não é feita através da simples reativação do estado definido no *checkpoint* como na abordagem sem *logs*, já que também os registros de *log* definem ações a serem novamente realizadas.

4.2. Handoff

Em ambientes móveis, conforme pode ser observado na figura 2, um *host* pode movimentar-se para uma nova célula e passar a ter uma nova ESM. Assim, torna-se necessário adotar uma nova abordagem para realizar a recuperação de *checkpoint* e/ou registros de *logs* que potencialmente podem estar espalhados através de várias ESMs.



Figura 2: Ocorrência de Handoffs

Este problema pode ser resolvido através da troca de informações referentes ao estado da aplicação entre a antiga ESM e a nova ESM do HM. A seguir, são definidas três estratégias da troca de informações durante o processo de *handoff*: a pessimista, a preguiçosa⁵ e a astuciosa⁶ [Pradhan et al., 1996].

⁵Tradução utilizada para a palavra *lazy* da língua inglesa.

⁶Tradução utilizada para a palavra *trickle* da língua inglesa.

4.2.1. Estratégia Pessimista

Nesta estratégia, quando um *host* se move para uma nova célula, o seu *checkpoint* é transferido para a nova ESM. No caso de estarem sendo utilizados *logs*, estes também são transferidos. A velha ESM do *host*, após receber a confirmação de recebimento do *checkpoint* da nova ESM, pode excluir a sua cópia do *checkpoint*. A desvantagem desta abordagem é o grande volume de informações que é trocado durante o processo de *handoff*.

4.2.2. Estratégia Preguiçosa

Na estratégia preguiçosa, os *checkpoints* não são transferidos durante o *handoff*; em vez disso, são criadas listas que determinam quais ESMs o *host* visitou, e a partir destas listas o *checkpoint* e os *logs* podem ser recuperados. Após o *handoff*, a nova ESM recebe uma mensagem que contém a localização da ESM anteriormente visitada pelo *host*. Já que o *host* tem liberdade para mover-se através de várias células, como resultado disso surge efetivamente uma lista de ligação entre as várias ESMs visitadas. Porém, quando um novo *checkpoint* é realizado, a atual ESM informa este evento para a antiga ESM, para que esta possa apagar a sua cópia do *checkpoint* e mais todas informações referentes àquele *host*. Este processo ocorre até que todas as ESMs da lista de ligação excluam as informações sobre aquele HM.

Esta estratégia diminui a sobrecarga decorrente da troca de informações entre ESMs durante o processo de *handoff*, se comparada com a estratégia pessimista. Contudo, o tempo de recuperação aumenta, já que possivelmente será necessário buscar informações espalhadas em várias ESMs.

4.2.3. Estratégia Astuciosa

Na estratégia preguiçosa, devido a uma grande mobilidade do *host*, o *checkpoint* e os *logs* podem ficar espalhados por várias ESMs o que aumentaria consideravelmente o tempo de recuperação. Portanto, visando atenuar este problema, e para manter os baixos custos de *handoff* da estratégia preguiçosa, é proposta a estratégia astuciosa. Esta estratégia garante que o *checkpoint* estará perto da atual ESM. Considerando que a distância entre a atual ESM e o anterior é de um "salto", esta estratégia garante que o *checkpoint* estará a no máximo um salto de distância da ESM atual.

Para conseguir isso, a nova ESM passa uma mensagem de controle para a ESM anterior do HM. Com recebimento desta mensagem, a ESM anterior trará para junto de si o último *checkpoint* e os possíveis *logs*. Assim, no caso de falha, para realizar a recuperação, a atual ESM pede o *checkpoint* e os *logs* que estão armazenados na ESM anterior. Quando um novo *checkpoint* for realizado e conseqüentemente for armazenado na ESM atual, a ESM atual enviará uma notificação autorizando a ESM anterior a excluir as informações referentes àquele HM.

4.3. Comparação entre estratégias

Segundo Pradhan et al.(1996), em esquemas tradicionais de recuperação, a taxa de falhas é o fator determinante na escolha do esquema de recuperação adequado. Já que em sistemas

wireless, não somente a taxa de falhas, mas também a taxa de mobilidade dos *hosts* e a largura de banda disponível devem ser levadas em consideração nessa escolha.

A tabela 1 mostra uma vinculação entre as estratégias e ambientes diversos, associados por Pradhan et al.(1996), de acordo com sua adaptabilidade. Segundo os próprios proponentes, é possível observar que não existe um esquema eficiente para todos os ambientes, embora a predominância seja de esquemas de salvamento que adotam *logs*.

Mobilidade	Largura de Banda	Taxa de Falhas	Esquema
Alta	Baixa	Baixa	Com <i>log</i> e preguiçosa
		Alta	Sem <i>log</i> e astuciosa
	Alta	Todas	Com <i>log</i> e astuciosa
Baixa	Todas	Todas	Com <i>log</i> e preguiçosa

Tabela 1: Esquemas de recuperação ótimos

5. Conclusão

Ambientes móveis apresentam-se como um desafio à área de tolerância a falhas graças às suas características inerentes como a mobilidade e baixa largura de banda. Uma maneira de obter-se a disponibilidade e a confiabilidade necessárias nesses sistemas é através da utilização de recuperação por *checkpoints*. Portanto, este artigo mostrou diferentes modos de realizar *checkpoints* em ambientes móveis. Entre os objetivos visados através das estratégias de recuperação propostas destacam-se a minimização da troca de informação entre *hosts* móveis e a adoção de uma estratégia de *handoff* que não cause sobrecarga devido à troca de mensagens e nem altos custos para a recuperação.

Referências

- Acharya, A. and Badrinath, B. (1994). Checkpointing distributed applications on mobile computers. *Proceedings of the Third International Conference on Parallel and Distributed Information Systems*.
- Jalote, P. (1994). *Fault Tolerance in Distributed Systems*. Prentice Hall, Englewood Cliffs, New Jersey 07632.
- Laamanen, H., Alanko, T., and Raatikainen, K. (1999). Dependability issues in mobile distributed system. *Pacific Rim International Symposium on Dependable Computing*.
- Neves, N. and Fuchs, W. K. (1997). Adaptive recovery for mobile environments. *Communications of the ACM*, 40(1):68–74.
- Pradhan, D. K., Krishna, P., and Vaidya, N. H. (1996). Recoverable mobile environment: Design and trade-off analysis. In *Symposium on Fault-Tolerant Computing*, pages 16–25, Texas - USA. IEEE.

Tolerância a Falhas em Sistemas de Armazenamento de Dados

Márcio Joel Barth^{1,2}, Edvar Bergmann Araujo²

¹Companhia de Processamento de Dados do RGS – PROCERGS
Praça dos Açorianos, s/n – 90010-340 – Porto Alegre – RS – Brasil

²Instituto de Ciências Exatas e Tecnológicas – Centro Universitário Feevale
RS 239, 2755 – 93352-000 – Novo Hamburgo – RS – Brasil

marcio-barth@procergs.rs.gov.br, edvar@feevale.br

Abstract. *The increasing use of computer networks as a way of sharing and accessing information resources have generated great pressure on the current data storage systems. Such systems became limited in their ability for manipulation and recovery of bigger data volumes, which became more and more heterogeneous and distributed in the network environment, arising the need for the implementation of management systems for data storage. These systems' new generations allow information to be gathered together in a single place of the company, thus allowing easier access to its localization, publication, safety, redundancy and administration. This way it is necessary that a storage system is supplied of appropriate techniques of fail tolerance to guarantee the wanted reliability, without the applications or users become aware of the employed techniques.*

Resumo. *A crescente utilização de redes de computadores como forma de compartilhamento e acesso a recursos de informação tem gerado grande pressão sobre os sistemas atuais de armazenamento de dados. Tais sistemas vêm apresentando limitações no que diz respeito à manipulação de grandes volumes de dados, que se tornam cada vez mais heterogêneos e distribuídos nos ambientes de rede, surgindo a necessidade da implantação de sistemas de gerenciamento de armazenamento de dados. As novas gerações desses sistemas permitem que informações sejam reunidas em um único lugar da empresa, facilitando assim a sua localização, publicação, segurança, redundância (backup) e gerenciamento. Desta forma, é necessário que um sistema de armazenamento seja suprido de técnicas de tolerância a falhas adequadas para garantir a confiabilidade desejada, sem que as aplicações ou os usuários tomem conhecimento das técnicas empregadas.*

1. Introdução

Recentemente, novas tecnologias como *Fibre Channel*¹, *Clustering*² e *Storage Networking* estão transformando o cenário de armazenamento que está caminhando em dois sentidos simultâneos: (i) recentralização, eliminando-se as ilhas existentes hoje; e (ii) externalização, separando o armazenamento da ligação física (*'bus attached'*) com

¹ *Fibre Channel*: tecnologia de rede projetada para altas taxas de transferência entre dispositivos de armazenamento e computadores.

² *Clustering*: coleção de computadores que são interconectados (tipicamente em altas velocidades) para o propósito de promover maior disponibilidade de serviços e/ou desempenho (via balanceamento de carga). Geralmente os computadores em *cluster* possuem acesso a uma área de armazenamento comum, e utilizam softwares especiais para coordenar as atividades dos componentes dos computadores.

os servidores. Estas tendências são o que se chama de modelo de computação ‘*information centric*’.

No modelo “*information centric*”, as informações são colocadas no centro do negócio e as plataformas de processamento são conectadas aos equipamentos de armazenamento. O modelo transcende plataformas e ambientes operacionais, tendo como objetivo a integração de todas as informações, fornecendo uma visão simples e única. É uma mudança significativa se comparado ao modelo ‘*server centric*’ atual, onde o processador é a peça chave da computação e o limitador da capacidade de acesso às informações [HDS, 2003].

2. Tolerância a falhas em sistemas de armazenamento

Cada vez mais as empresas utilizam sistemas de armazenamento a fim de garantirem maior disponibilidade e confiabilidade das informações, em muitos casos configurando-se em modelos de armazenamento do tipo DAS, NAS ou SAN. Para garantir o perfeito funcionamento destes sistemas torna-se necessário que os subsistemas de armazenamento possuam a capacidade de continuar funcionando mesmo quando ocorrer falha em um disco ou outro componente (possivelmente com redução do nível de desempenho). A seguir são comentados vários aspectos referentes à tolerância a falhas em subsistemas de armazenamento que devem ser considerados.

2.1 RAID

O sistema RAID (*Redundant Array of Independent Disks*) – alguns fabricantes utilizam o termo “*Inexpensive*” no lugar de “*Independent*” - surgiu em 1987 pelos pesquisadores Patterson, Gibson e Katz, da Universidade da Califórnia, Berkeley. É um método que combina vários discos em uma única unidade lógica. Um *disk array* RAID oferece tolerância a falhas e melhores taxas de transferência do que um *drive* único ou um grupo de *drives* independente.

A configuração de um RAID pode ser realizada através do próprio sistema operacional (*software*), caso o mesmo ofereça o serviço, ou pela controladora (*hardware*), que neste caso, é o modo mais aconselhável por oferecer maior desempenho e liberar o sistema operacional desta tarefa [MOURA, 2001].

O RAID constitui-se a base para todas as funcionalidades esperadas num sistema de armazenamento em termos de proteção dos dados, tolerância a falhas, altos níveis de desempenho, grande capacidade de armazenamento e escalabilidade.

A implementação de um sistema RAID é possível utilizando-se as controladoras SCSI e IDE, que permitem a conexão e a configuração de vários discos a fim de obter-se as vantagens que o RAID proporciona.

Podem ser citadas três características fundamentais dos RAID’s [SNIA,2003]:

- Vários discos acessados em paralelo fornecem uma taxa de I/O superior a de um único disco;
- O armazenamento de dados de modo redundante em vários discos oferece melhor tolerância à falha;

- O uso da tecnologia “*Hot-plug*” possibilita trocar um dispositivo com falha sem que o servidor tenha que ser desligado.

Há diferentes números que especificam o nível de segurança implementada em um produto RAID. Os níveis mais comuns são 0, 1 e 5 e estão entre os mais utilizados pelas corporações. A sua escolha deve ser determinada de acordo com as necessidades das aplicações que executam na empresa.

2.2 Falhas de interconexão

Técnicas de espelhamento e configuração RAID de discos protegem contra falhas em discos, mas os discos não são o único ponto que pode falhar. O barramento ou conexão de fibra SAN que conectam os dispositivos de armazenamento aos servidores, também pode falhar, tornando os dados inacessíveis.

O volume RAID na figura 1A possui uma interconexão entre cada disco e a controladora RAID do servidor. Se uma conexão falhar, somente um disco ficará inacessível e o volume poderá continuar funcionando através da reconstrução dos dados.

No volume da figura 1B, ao contrário, os discos compartilham a mesma conexão. Se a conexão falhar, mais de um disco poderá ficar inacessível e a reconstrução do volume poderá ficar comprometida. A conclusão é que o máximo de cuidado na configuração dos volumes é necessária para um nível máximo de tolerância a falhas [BARKER, 2002].

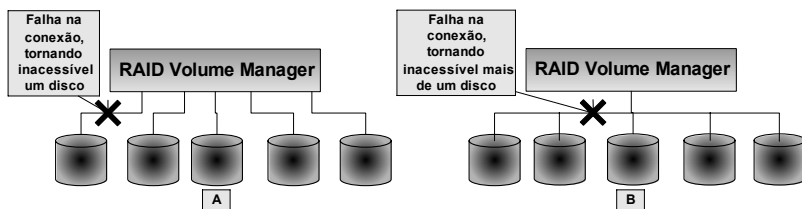


Figura 1- Interconexão individual de discos implementando tolerância a falha.

2.3 Falha nas controladoras RAID

Controladoras RAID são projetadas para garantir tolerância a falhas, embora a mesma possa falhar causando um impacto sobre a disponibilidade dos dados. Assim o controle para tolerância a falhas é tipicamente ativado com duas ou mais controladoras conectadas ao mesmo disco e servidor. Na figura 2, todos os discos estão conectados por duas controladoras RAID externas, que trocam mensagens entre si assegurando que ambas estão ativas. Caso uma controladora deixe de receber mensagens da outra, ela compreende que deve assumir a comunicação dos discos da controladora que falhou para não comprometer a disponibilidade dos dados [BARKER, 2002].

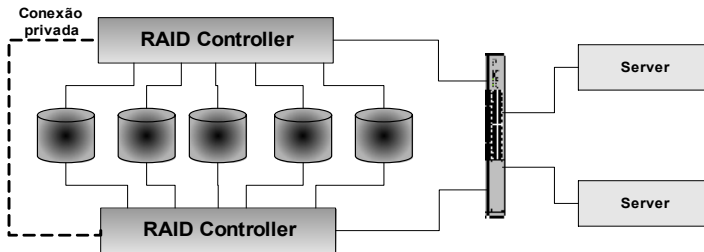


Figura 2 – Tolerância a falhas em controladoras RAID externas.

2.4 Tolerância a falhas transparentes e não transparentes

Mesmo que ambas as controladoras da figura 2, utilizem a mesma infra-estrutura para conectarem-se ao servidor, os volumes presentes devem necessariamente possuir diferentes endereçamentos (por exemplo, *Fibre Channel LUNs (Logical Unit Number)*). Quando uma controladora falha, a outra toma conta dos discos e apresenta seus volumes com o mesmo endereçamento utilizado pela controladora que falhou. Com a menor possibilidade de erro durante a transição, a troca de controladora é transparente para a aplicação. Os mesmos volumes são endereçados para o mesmo endereço, isso, é claro, se a outra controladora estiver gerenciando tudo.

Isto é uma solução elegante, mas ainda possui um ponto de falha, se a infra-estrutura da rede SAN falhar, então todo o acesso aos dados será perdido. Uma solução para este problema é conectar cada controladora RAID em um segmento SAN separado.

Com um subsistema de entrada/saída configurado desta forma, se um dos segmentos da SAN falhar, todos os dados permanecerão acessíveis, porque se a conexão até o servidor continuar funcionando o controle de todos os discos estará ativo pela outra controladora.

Alguns subsistemas equipam cada controladora RAID com duas ou mais conexões, como ilustrado na figura 3. Assim não somente servidores, mas também as controladoras RAID poderão estar conectadas aos dois segmentos da SAN. Esta configuração elimina a necessidade de substituir todos os discos para uma controladora se um segmento da SAN falhar [BARKER, 2002].

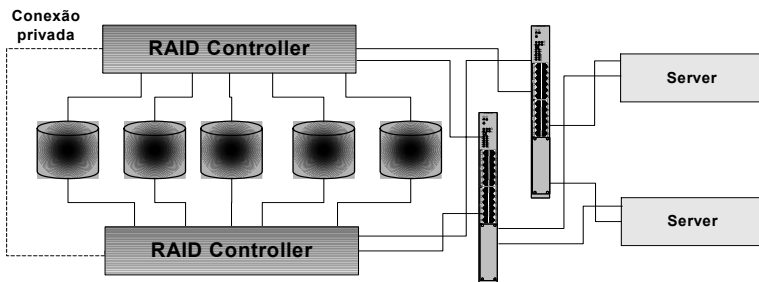


Figura 3 – Interconexão de controladoras RAID completamente redundantes.

2.5 Operações atômicas e integridade de dados

Muitas controladoras RAID possuem *cache* reverso para aumentar o desempenho de entrada/saída. Se uma controladora RAID ficar sem alimentação de energia e com uma atualização pela metade (por exemplo, uma cópia do espelhamento dos dados completa e a outra não), o volume pode retornar com dados corrompidos em algum momento no futuro. Por exemplo, se a energia falhar e somente uma cópia de um bloco de dados de um espelhamento estiver completa, uma leitura futura daquele bloco poderá retornar com resultados diferentes dependendo de qual disco espelhado for selecionado para satisfazer a solicitação de leitura.

Se a controladora RAID for realmente tolerante a falhas, deve proteger contra a perda de estado e conteúdo de *cache* quando faltar energia (ou quando a controladora falhar). A solução mais utilizada para este problema para as controladoras RAID interconectadas é a comunicação de mudança de estado e a respectiva mudança no conteúdo de sua *cache* reversa entre as controladoras. Independente da tecnologia implementada, o mais importante para os usuários de controladoras RAID é que os estados das operações e os dados de *cache* sejam preservados no caso da controladora falhar, não havendo assim, perda de dados [BARKER, 2002].

2.6 Replicação de sistema de armazenamento

Com a replicação dos dispositivos de armazenamento, todos os blocos gravados no sistema de armazenamento primário são replicados para dispositivos de igual capacidade para cada local secundário, sem considerar o significado dos blocos de dados replicados. Assim aumenta-se a tolerância a falhas através de um *site* para recuperação de desastres, que é um *datacenter* secundário localizado distante o suficiente para continuar as operações se o *site* primário sofrer um desastre irreversível [MOURA, 2002].

Muitas tecnologias de replicação permitem ao administrador de sistema escolher entre duas opções:

Replicação síncrona: para cada atualização da aplicação gravada no *storage* principal é esperada uma validação do *storage* secundário antes que seja considerada completa a transação.

Replicação assíncrona: com o *storage* secundário podendo ficar defasado em relação ao *storage* principal, o mesmo grava as informações em tempos pré-determinados.

A replicação síncrona simplifica a conversão de dados entre um *storage* secundário em relação ao *storage* principal após um desastre, porque muito poucos dados estão no canal de transmissão. Mas a replicação síncrona de acordo com a forma de conexão utilizada pode afetar as aplicações adversamente, pois ficam esperando pelo *storage* secundário por longos tempos de resposta para completar uma transação.

A replicação assíncrona essencialmente elimina o tráfego da rede e o desempenho do *storage* secundário e dos tempos de resposta da aplicação. Embora sem atualização de transação executa a recuperação dos dados depois de um desastre de forma mais complexa do que a síncrona, pois as réplicas do *storage* secundário podem estar sensivelmente desatualizadas.

3 Conclusão

A grande dúvida das corporações sempre paira sobre a real necessidade e aplicabilidade de soluções tolerantes a falhas, pois além do alto custo sobre as soluções de armazenamento, a eficácia de implantação desses sistemas não pode ser mensurada. Sem dúvida, com a crescente demanda por recursos de armazenamento que devem estar disponíveis 24xForever, estes métodos de tolerância a falhas serão cada vez mais necessários para a disponibilidade e confiabilidade de qualquer sistema.

Desta forma, buscou-se a comprovação da aplicabilidade dos métodos citados no artigo, que se baseiam no método de *fail-stop* (são acionados somente quando existe a falha), através de um estudo de caso em duas empresas de grande porte que possuem equipamentos que já implementam parte das técnicas de tolerância a falhas. Com os testes realizados constatou-se que neste contexto a aplicabilidade dos métodos só é possível utilizando-se equipamentos de mesmo fornecedor, modelo e capacidade. Sendo assim, outros métodos de tolerância a falhas deverão surgir a fim de suprir outras necessidades como, por exemplo, garantir tolerância a falhas entre dispositivos de tamanhos e fornecedores diferentes.

4 Referências

BARKER, Richard & MASSIGLIA, Paul. Storage area networks essentials: a complete guide to understanding and implementing SANs. 2.ed. – New York: John Wiley & Sons, 2002. 498 p.

High Digital Storage. Pensando em armazenamento. Disponível em: <http://www.hdsinfo.com.br/inpdfs.htm>. Acessado em: agosto 2003.

MOURA, Giedre. Storage quer romper as barreiras. **Network**, São Paulo: v. 3, n. 44, p. 28 – 34, out. 2002.

MOURA, Giedre. Todo o poder para o RAID. Network Computing. Disponível em: <http://www.networkcomputing.com.br/noticias/artigo.asp?id=18674>. Publicado em: 13 nov. 2001. Acessado em: agosto 2003.

Storage Network Industry Association. Disponível em: <http://www.snia.org>. Acessado em: agosto 2003.

Uma Estratégia para Validação Experimental de um Sistema de Comunicação de Grupo*

Gabriela Jacques da Silva^{1†}, Taisy Silva Weber¹

¹Grupo de Pesquisa em Tolerância a Falhas
Programa de Pós-Graduação em Ciência da Computação
Instituto de Informática - Universidade Federal do Rio Grande do Sul
Caixa Postal 15064 - CEP 91591-970 - Porto Alegre - RS
{gjsilva,taisy}@inf.ufrgs.br

Abstract. *This paper presents a strategy to an experimental validation of JGroups group communication system. This kind of system is commonly used as a basic building block for the development of fault-tolerant applications, due to features such as reliable multipoint communication. The validation of this system by fault injection helps to remove errors on the group communication mechanism and to verify the behavior of this system in the presence of faults.*

Resumo. *Este artigo apresenta uma estratégia para a validação experimental do sistema de comunicação de grupo JGroups. Este tipo de sistema geralmente é usado como base para o desenvolvimento de aplicações tolerantes a falhas por apresentar características como, por exemplo, comunicação multiponto confiável. A validação deste sistema por injeção de falhas é essencial na remoção de erros na construção dos mecanismos de comunicação de grupo, como também verificar o comportamento deste na presença de falhas.*

1. Introdução

A medida que sistemas computacionais são incorporados no cotidiano, estes necessitam apresentar características de tolerância a falhas para corresponder a confiança depositada no comportamento correto desses sistemas. Uma das maneiras é a replicação, tanto de *hardware* quanto de *software*. Como técnicas de replicação por *hardware* são bastante caras, foram motivadas várias pesquisas para o desenvolvimento de técnicas de replicação baseadas em *software*.

O uso de múltiplas cópias de uma informação pode gerar problemas de inconsistência, no caso em que as atualizações não sejam feitas na mesma ordem ou então quando estas não são propagadas corretamente para todas as cópias. Comunicação multiponto confiável e ordenamento de mensagens são características oferecidas por sistemas de comunicação de grupo, facilitando portanto a construção de técnicas de replicação [GUERRAOU1 e SCHIPER 1996]. Sistemas de comunicação de grupo apresentam vários outros usos clássicos, como suporte para sistemas operacionais distribuídos, transações distribuídas, replicação de bases de dados, balanceamento de carga, gerenciamento de sistemas, servidores de alta disponibilidade e computação colaborativa

*Projeto ACERTE (472084/2003-8)

†Bolsista do Conselho Nacional de Desenvolvimento Científico e Tecnológico

[CHOCKLER *et al.* 2001]. Devido a esta larga aplicabilidade, faz-se essencial a validação deste tipo de sistema.

Este artigo propõe uma estratégia para a validação experimental de um sistema de comunicação de grupo chamado JGroups [BAN 1998]. Para isto são executados experimentos de injeção de falhas para verificar o comportamento deste *middleware* em presença de falhas. A próxima seção aborda a importância da validação experimental e também a técnica de injeção de falhas. A seção 3 descreve características gerais dos sistemas de comunicação grupo e também JGroups. A seção 4 apresenta a estratégia para validação deste sistema, explicitando os modelos de falhas e o injetor de falhas em uso. A última seção apresenta algumas conclusões e trabalhos futuros.

2. Validação Experimental

Uma fase fundamental no desenvolvimento de sistemas tolerantes a falhas é a fase de validação. Delegar a verificação do funcionamento do mecanismo de tolerância a falhas para uma situação de uso efetivo do *software* (uma falha real) pode gerar consequências completamente desastrosas. Para verificar o funcionamento correto destes mecanismos, os sistemas devem ser validados. A validação pode ser analítica e experimental, sendo estas duas formas complementares. Uma das técnicas usadas para validar experimentalmente um sistema é através da injeção de falhas. A injeção de falhas testa a eficiência dos mecanismos de tolerância a falhas e avalia a segurança de funcionamento dos sistemas, provendo uma realimentação no processo de desenvolvimento [IYER 1995].

Um ambiente de injeção de falhas é formado geralmente por um sistema alvo, um injetor de falhas, uma biblioteca de falhas, um gerador de carga de trabalho (*workload*), uma biblioteca de carga de trabalho, um controlador, um monitor e um coletor e analisador de dados [HSUEH *et al.* 1997]. O funcionamento básico destes ambientes parte da injeção de falhas no sistema alvo com o injetor de falhas. O sistema alvo é alimentado com o gerador de carga de trabalho. A execução do sistema é monitorada pelo monitor que comunica eventos para o coletor de dados. Este coletor rastreia a execução, que pode ser posteriormente analisada pelo analisador de dados.

Ferramentas de injeção de falhas podem ser desenvolvidas tanto em *hardware* quanto em *software*. A decisão entre uma implementação e outra depende não só do tipo de falha que se quer injetar mas também do esforço despendido para a criação de cada uma. O uso de um injetor de falhas implementado em *software* apresenta um menor custo, já que não é necessário um dispositivo específico. Além disso, é mais simples injetar falhas em camadas superiores, como o sistema operacional e a própria aplicação em teste. A estratégia apresentada compreende apenas injetores de falhas implementados em *software*. A seção a seguir trata sobre sistemas de comunicação de grupo e também descreve brevemente o sistema JGroups, que é o sistema alvo da validação.

3. Sistemas de Comunicação de Grupo

Sistemas de comunicação de grupo são *middlewares* para comunicação multiponto confiável. Os membros de um grupo podem não ser apenas processos, mas também objetos, ou ainda qualquer entidade que possa enviar e receber mensagem para/de um grupo [BAN 1998]. Estes *middlewares* também oferecem propriedades como acordo, validade,

integridade e terminação. Outra possibilidade é o ordenamento no recebimento de mensagens, característica necessária em sistemas que necessitam que atualizações de valores sejam realizadas de forma consistente.

Outro serviço geralmente oferecido por estes sistemas é o serviço de *membership*. Este serviço é responsável por gerenciar os membros de cada grupo a cada instante, que pode mudar sempre que um membro se junta ou abandona um grupo (voluntariamente ou em casos de falha). Para representar a cada momento os membros de um grupo é usado o conceito de visão. Os membros de um grupo são comunicados da adesão ou do abandono de um membro através da instalação de uma nova visão.

Através das várias características e serviços oferecidos por um sistema de comunicação de grupo, estes se tornam uma peça básica para a construção de sistemas distribuídos tolerantes a falhas. Um sistema que está sendo usado atualmente em vários projetos é JGroups, que será brevemente descrito na próxima seção.

3.1. JGroups

JGroups [BAN 1998] é uma *toolkit* para comunicação de grupo confiável desenvolvido em Java. A abstração de grupos é feita através de um canal. Todos os membros que realizarem a conexão em canais de mesmo nome fazem parte de um mesmo grupo. Quando um canal é criado pelo usuário, pode ser escolhida a pilha de micro-protocolos que este canal irá usar. Desta forma o usuário tem a liberdade de escolher quais as restrições que o canal deve impor sobre os membros deste, como por exemplo o ordenamento de mensagens. Outro objetivo deste sistema é oferecer abstrações de mais alto nível para a comunicação de grupo, facilitando o uso deste paradigma. Para isso, JGroups oferece vários blocos de *software* prontos com padrões de comunicação de grupo implementados.

Atualmente vários projetos usam JGroups como um bloco auxiliar para prover alta disponibilidade. Entre eles pode citar-se o JBoss [JBoss 2003], que é um servidor de aplicações J2EE gratuito e usa JGroups para a implementação da clusterização de servidores. Outro sistema é Dorothy [PASIN *et al.* 2002]. JGroups neste sistema serve como base para a realização da consistência de réplicas de componentes Enterprise JavaBeans.

Na seção a seguir é apresentada uma estratégia para validar experimentalmente este sistema de comunicação de grupo. Esta validação implica em vários benefícios, como a remoção de erros latentes no mecanismo de comunicação de grupo e também a verificação do comportamento deste em caso de falhas. Realizar experimentos de validação em JGroups auxilia não só na validação deste sistema em si, mas também para a validação das aplicações que o usam como base para alcançar alta disponibilidade.

4. Estratégia para Validação de JGroups

A condução de um experimento de injeção de falhas envolve várias decisões, como o modelo de falhas a ser considerado, a ferramenta injetora de falhas, as aplicações de geração de carga de trabalho, cenários de testes e as métricas para avaliação do sistema alvo.

4.1. Modelo de Falhas

Uma falha ocorre sempre que um serviço não é prestado de acordo com a sua especificação. Dependendo do tipo desta falha ela recebe uma classificação. Como um sistema de comunicação de grupo é executado em um contexto distribuído, o modelo de

falhas que será considerado para este experimento será o modelo de falhas para sistemas distribuídos definido por Cristian [CRISTIAN 1991].

Este modelo descreve falhas de colapso, omissão, temporização e valor. Uma falha por colapso ocorre quando o servidor pára totalmente de responder. Este tipo de falha ainda pode ser classificada em subtipos, levando em consideração o estado do servidor após a sua recuperação. Uma falha de omissão ocorre quando um servidor não responde a uma requisição. Uma falha é considerada de temporização quando uma resposta do servidor ocorre fora do intervalo de tempo especificado, podendo ser tanto uma resposta cedo demais quanto tardia demais. A última geralmente é associada a falhas de desempenho. A falha por valor ocorre quando o valor de retorno de uma requisição é incorreto. Este tipo de falha não será considerada para este experimento de validação. O sistema alvo, o JGroups, está especificado para tolerar apenas falhas de colapso. A estratégia descrita se resume a este modelo, porém pode ser estendida a outros sistemas que possuem modelos de falhas menos restritivos que apenas o de colapso.

4.2. Ferramenta de Injeção de Falhas

O modelo de falhas considerado inclui falhas de omissão, temporização e colapso. Para provocar falhas de omissão e temporização é necessário provocar falhas no envio de mensagens. Com isso, a falha deve ser injetada no sistema de troca de mensagens. Um meio de provocar falhas por colapso é fazer com que os outros membros do grupo não recebam mais mensagens do membro falho. Provocar o corte completo do sistema de comunicação de um membro do grupo provocará a percepção de uma falha por colapso nos outros membros do grupo, já que não será possível trocar mensagens com este membro em colapso.

Para injetar falhas de comunicação foi escolhido o injetor de falhas GOOFI (*Generic Object-Oriented Fault Injection*) [AIDEMARK *et al.* 2001]. Esta ferramenta é genérica, já que não é presa a nenhuma técnica específica de injeção de falhas. Desta maneira é construído um ambiente de fácil adaptação para injeção de falhas necessárias para um determinado sistema alvo. Esta ferramenta é altamente portátil por ser desenvolvida em Java e usar um banco de dados compatível com a linguagem SQL. A arquitetura desta ferramenta pode ser vista na figura 1.

A arquitetura da ferramenta pode ser dividida em três camadas. A GUI (*Graphical User Interface*) é usada para configurar e iniciar os experimentos de injeção de falhas. A camada central inclui as classes `Java FaultInjectionAlgorithms`, `Framework`, `TargetSystemInterface`. Estas classes definem métodos abstratos que devem ser especificados de acordo com o algoritmo de injeção de falhas e o sistema alvo do experimento. Neste trabalho estas classes serão estendidas para gerar falhas de comunicação e para interagir com JGroups. A última camada da arquitetura é a interface com o banco de dados. Esta é usada para armazenar informações sobre o sistema alvo, os experimentos de injeção de falhas e os dados capturados de cada experimento.

Um modo complementar proposto para injetar falhas no sistema JGroups é através da extensão e criação de alguns micro-protocolos, que podem ser configurados no momento da criação de um canal. JGroups dispõe de dois protocolos para atraso e descarte de mensagens. Estes podem ser estendidos para adequá-los aos experimentos. O micro-protocolo de atraso pode ser usado para injetar falhas de temporização e o de descarte para falhas de omissão. Para injeção de falhas de colapso, todas as mensagens de

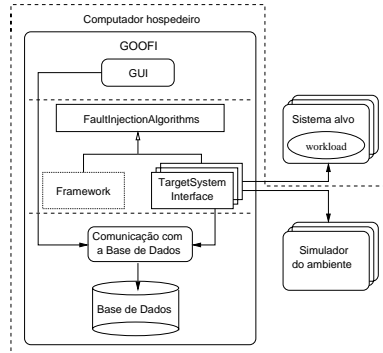


Figura 1: Arquitetura da ferramenta GOOFI

todos os canais de comunicação do emissor devem ser descartadas após a primeira omissão.

4.3. Geração de Carga de Trabalho

Para gerar carga de trabalho são executadas aplicações que usam serviços essenciais ou freqüentemente oferecidos por um sistema de comunicação de grupo. Um serviço que deve ser bastante explorado é o de *membership*, já que várias aplicações baseiam-se neste serviço para prover alta disponibilidade. Outros serviços que devem ser testados são o de ordenação total, causal e FIFO. Um sistema que usa intensivamente ambos serviços é Dorothy e está sendo utilizado como carga de trabalho.

4.4. Cenários de Testes

A carga de trabalho usa a *middleware* de comunicação de grupo para manutenção de consistência de réplicas. Para a condução de experimentos em ambientes com objetos replicados são necessários no mínimo cinco servidores. Um cenário adequado é injetar falhas em um único servidor que contenha uma das réplicas e monitorar o comportamento dos outros servidores (membros do grupo de replicação). O efeito esperado de uma injeção de falha de colapso em um membro é a troca de visão do grupo nos membros não-falhos. Para cada falha de colapso provocada deve corresponder uma troca de visão, refletindo a exclusão do membro falho do grupo.

Atualmente estão sendo feitos testes de injeção de falhas em um único servidor. Esta abordagem centralizada tem a vantagem de facilitar o controle do experimento, porém reduz os experimentos para somente falhas simples (um único servidor por vez). Posteriormente serão feitos testes para provocar falhas de colapso em dois ou mais servidores simultaneamente ou ainda falhas em cascata.

4.5. Métricas de Avaliação

A primeira métrica que deve ser obtida é a cobertura de falhas do detector de falhas do JGroups. Esta medida pode ser tomada de forma indireta computando o número de mudanças de visão devido ao colapso do servidor. Dividindo-se este número pelo número de falhas injetadas no experimento chega-se a cobertura de falhas. Esta medida permite uma realimentação no desenvolvimento do mecanismo de detecção de falhas do JGroups.

As duas outras métricas de avaliação são as perdas de desempenho provocadas por falhas e a disponibilidade do sistema. Estas medidas são importantes para o desenvolvedor de aplicações tolerantes a falhas que se baseiam neste sistema de comunicação de grupo. Um modo de obtenção da queda de desempenho devido a presença de uma falha é através da comparação do tempo médio de resposta de um cliente do grupo de replicação em duas situações. A primeira situação é quando o serviço está livre de falhas e a segunda é em presença de falhas. A carga de trabalho deve ser a mesma em ambas as situações.

Para obter a medida de disponibilidade do sistema é necessário verificar o tempo de recuperação do sistema após a ocorrência de uma falha. Inicialmente esta métrica pode ser medida calculando a diferença entre o tempo de resposta ao cliente com e sem a presença de falhas no servidor. Este tempo adicional é considerado o tempo de indisponibilidade ou de recuperação.

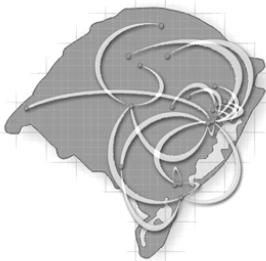
5. Conclusão

A validação experimental de sistemas de comunicação de grupo é essencial, já que freqüentemente estes são usados como um bloco de suporte a aplicações tolerantes a falhas. A validação destes sistemas traz benefícios não somente para o próprio sistema, mas também para as aplicações que o usam como *middleware* de comunicação de grupo.

O artigo apresentou uma estratégia para conduzir experimentos de validação usando o injetor de falhas GOOFI no sistema alvo JGroups. Uma extensão desta estratégia será usada, após a validação de JGroups, para validar aplicações de alta disponibilidade baseadas em replicação de componentes, como providas pelo sistema Dorothy que se encontra em fase final de implementação.

Referências

- AIDEMARK, J.; VINTER, J.; FOLKESSON, P.; KARLSSON, J. *GOOFI: Generic Object-Oriented Fault Injection Tool*. In Proceedings of International Conference on Dependable Systems and Networks 2001. Gotemburgo, Suécia. Julho 2001.
- BAN, B. *JavaGroups - Group Communication Patterns in Java*. Department of Computer Science, Cornell University. Julho 1998.
- CHOCKLER, G. V.; KEIDAR, I.; VITENBERG, R. *Group Communication Specifications: A Comprehensive Study*. ACM Computing Surveys, Vol. 33, No. 4, pp. 427-469. Dezembro 2001.
- CRISTIAN, F. *Understanding Fault-Tolerant Distributed Systems*. Communications of the ACM, Vol. 34, No. 2, pp. 56-78. Fevereiro 1991.
- GUERRAOU, R.; SCHIPER, A. *Fault-Tolerance by Replication in Distributed Systems*. In International Conference on Reliable Software Technologies. Springer Verlag, Lecture Notes in Computer Science 1088. 1996.
- HSUEH, M.; TSAI, T. K.; IYER, R. K. *Fault Injection Techniques and Tools*. Computer, Vol. 30, No. 4, pp. 75-82. Abril 1997.
- IYER, R. K. *Experimental Evaluation*. In Proceedings of 25th International Symposium on Fault-Tolerant Computing. Pasadena, Estados Unidos. Junho 1995.
- JBoss J2EE Application Server. <http://www.jboss.org>
- PASIN, M.; WEBER, T. S.; RIVEILL, M. *A Multi-layer Architecture for High Available Enterprise JavaBeans*. Anais do III Workshop em Tolerância a Falhas. Búzios, Rio de Janeiro. Maio 2002.



Resumos

Minicursos e Oficinas

Minicurso I

Título: Avanços rumo à Integração de Tecnologias de Gerenciamento de Redes e Segurança

Ministrantes:

- Luciano Paschoal Gasparly (UNISINOS)
- Leonardo Lemes Fagundes (UNISINOS)

Resumo:

Com a popularização das redes de computadores e a sua utilização como suporte a aplicações críticas, a gerência e a segurança das redes passaram a ser pontos chave nas organizações. No entanto, tem-se observado a utilização de duas estruturas paralelas, uma para atender cada área, o que acarreta em dificuldade para obter uma visão integrada da infraestrutura de hardware e software utilizados na organização.

Este minicurso busca, através de exposições teóricas e demonstrações

- a) apresentar técnicas que permitem aproximar as tecnologias de gerência das de segurança
- b) discutir aspectos favoráveis e não favoráveis dessa integração.

O curso apresenta como utilizar o protocolo SNMP e a observação de objetos das MIBs II, RMON e RMON2 para detectar, baseado em anomalia, diversos tipos de ataques (ex: varreduras e negação de serviço). Em seguida, o curso descreve um agente SNMP, proposto pelo nosso grupo de pesquisa, que permite a detecção baseada em assinaturas (complementando a abordagem anterior). Serão apresentadas, ainda, interfaces para os sistemas de segurança existentes em algumas plataformas comerciais de gerenciamento e técnicas para a integração e a correlação de logs e alertas dos sistemas de segurança com os gerados pelas plataformas de gerenciamento.

Minicurso II

Título: Comunicação Multicast em Middleware CORBA

Ministrante: Alysson Neves Bessani

Resumo:

O objetivo deste minicurso é introduzir conceitos e abstrações de comunicação de grupo, apresentando os tipos de difusão (multicast) possíveis, e como eles podem ser introduzidos na arquitetura CORBA, baseando-se nas principais propostas encontradas na literatura e nas recentes especificações UMIOP e FT-CORBA da OMG. Para isso, o minicurso fará uma introdução geral do universo CORBA abordando sua arquitetura de referência, protocolos, objetos de serviço e alguns exemplos de programação neste ambiente. Além disso, será dado uma ênfase nas especificações UMIOP e em nossos esforços em propor soluções para torná-lo confiável (ReMIOP). Alguns exemplos de programação serão apresentados para que os participantes saibam como utilizar multicast no CORBA.

O perfil desejado, mas não obrigatório, do aluno é conhecimentos básicos de redes de computadores e sistemas distribuídos.

Minicurso III

Título: Bluetooth

Ministrantes:

- Ana Cristina Benso da Silva (PUC-RS)
- Andrei Oliveira da Silva (PUC-RS)
- Fabrício D'ávila Cabral (PUC-RS)

Resumo:

Bluetooth é uma das tecnologias disponíveis atualmente para a criação de redes sem fio. Neste minicurso serão abordados aspectos históricos, tecnológicos e práticos, visando proporcionar conhecimento básico sobre esta tecnologia. O objetivo específico é oferecer conhecimento teórico e prático que permita aos usuários instalar e configurar uma rede Bluetooth.

Oficina I

Título: Demonstração das Ferramentas de Simulação para Redes de Computadores

Equipe:

- Ricardo Hernandez Fernandes (II-UFRGS)
- Oscar N. Mori (II-UFRGS)
- Leonardo Sewald (II-UFRGS)
- Roberto Saltz Rosenfeld (II-UFRGS)
- Fernando M. Silveira (II-UFRGS)
- Lara Dalto de Souza (II-UFRGS)

Resumo:

Esta oficina pretende demonstrar três diferentes ferramentas para simulação de Redes de Computadores, quais sejam:

- 1) UC Berkeley, LBL, USC/ISI, Xerox PARC, Virtual InterNetwork TestBed (VINT) - Network Simulator: NS-2.
- 2) National Chiao Tung University (Taiwan): NCTUns.
- 3) Global Mobile Information Systems Simulation Library: GloMoSim - Parsec (UCLA).

Simuladores de Redes são ferramentas que geralmente envolvem conhecimento de protocolos e programação em linguagens de descrição de topologia, como Tool Command Language - TCL (Ex: NS-2), Scripts e Java (Ex:GloMoSim). Geralmente todo o simulador vem com ambiente gráfico (GUI) para a visualização da simulação elaborada inicialmente. Os simuladores mostram no final métricas de Redes tais como: Atraso (Delay), Jitter, Vazão (Throughput) e Descarte de datagramas (grau de perda de pacotes). De posse destas métricas, é possível fazer uma avaliação do desempenho da arquitetura da rede definida. Como exemplo serão criadas arquiteturas de redes dos tipos Wired e Wireless e estas serão demonstradas com os três simuladores.

Os participantes da oficina poderão criar suas próprias topologias usando uma ou outra ferramenta aplicando o que foi ensinado.

Oficina II

Título: Suporte a QoS em Roteadores FreeBSD

Equipe:

- Clarissa Marquezan (II-UFRGS)
- Lisandro Zambenedetti Granville (II-UFRGS)
- Ricardo Vianna (II-UFRGS)
- Rodrigo Sanger Alves (II-UFRGS)
- Tiago Fioreze (II-UFRGS)

Resumo:

O suporte a QoS é uma necessidade crítica em equipamentos de redes, principalmente quando a infra-estrutura de redes é utilizada para tráfego multimídia compreendendo

dados, voz e vídeo. Tal suporte a QoS, entretanto, eleva consideravelmente o custo dos equipamentos utilizados, o que em redes pequenas pode tornar a disponibilização de serviços com suporte a QoS proibitivos.

Se por um lado os dispositivos com suporte a QoS possuem um custo mais elevado, soluções alternativas e de baixo custo acabam por se tornar, muitas vezes, a única opção. Diversas redes atualmente são implantadas através do uso de roteadores baseados em equipamentos PCs com várias placas de rede, rodando sistemas operacionais Unix (e.g. Linux, FreeBSD, etc.). Tais soluções mostram-se eficientes e, principalmente, de baixo custo.

O objetivo desta oficina é apresentar o suporte a QoS fornecido por roteadores baseado no sistema operacional FreeBSD. O suporte a QoS é fornecido através do AltQ, que disponibiliza diversas facilidades (e.g. conformação de tráfego, policiamento, reserva de banda, etc.) que, se adequadamente configuradas, podem fornecer serviços bem mais adequados em redes com suporte a tráfego multimídia.

A oficina envolve a apresentação de noções básicas de QoS, uma verificação de roteadores baseados em FreeBSD, e um conjunto extenso de experimentações de configuração e testes do AltQ para fornecimento de QoS.

Oficina III

Título: Instalação, Configuração e Uso de um Agente de Monitoração RMON2 de Código Aberto e Gratuito

Equipe:

- Luciano Paschoal Gasparly (UNISINOS)
- Ricardo Nabinger Sanchez (UNISINOS)
- Lucio Braga (UNISINOS)
- Debora Pandolfi Alves (UNISINOS)

Resumo:

A oficina tem por objetivo familiarizar os ouvintes com a MIB (Management Information Base) RMON2 (Remote Network Monitoring Version 2), bem como com a instalação, a configuração e o uso de um agente de monitoração RMON2 gratuito e de código aberto desenvolvido pelo grupo de pesquisa em Redes de Computadores e Sistemas Distribuídos da UNISINOS.

A MIB RMON 2 opera com protocolos acima do nível de enlace, provendo informações necessárias para a monitoração de protocolos de alto nível e aplicações distribuídas. As informações disponibilizadas por esta MIB viabilizam a gerência de negócios de uma rede, possibilitando aos administradores ter visão do comportamento das aplicações e protocolos de alto nível sendo executados, da taxa de utilização dos recursos envolvidos e dos usuários que mais os consomem. Assim, passam a ter condições de redefinir o fluxo de tráfego da rede, buscando uma melhor utilização destes recursos, e de observar quem se comunica com quem e quais aplicações estão sendo utilizadas, permitindo o estabelecimento de políticas que garantirão a utilização adequada da rede.