P2P-Role: Uma Arquitetura de Controle de Acesso Baseada em Papéis para Sistemas Colaborativos Peer-to-Peer

Rafael da Rosa Righi, Felipe Rolim Pellissari, Carla Merkle Westphall

[1] Programa de Pós-Graduação em Ciência da Computação – PPGCC
Laboratório de Redes e Gerência (LRG) – Universidade Federal de Santa Catarina
Caixa Postal 476 - 88040-900, Florianópolis, SC

{rrighi,rolim,carla}@lrg.ufsc.br

Abstract. *The collaborative Peer-to-Peer systems are distributed systems where each user acts as a client and server of resources. The discovery, representation and protection of these resources are the main challenges in Peer-to-Peer networks. This paper defines a role-based access control architecture (RBAC) specific to Peer-to-Peer systems and in this way, contributes to fortify the security of these type of models. The prototype implemented validates the model and provides manners for security policy management of resources in each node of Peer-to-Peer network.*

Resumo. *Os sistemas colaborativos Peer-to-Peer apresentam uma forma de computação distribuída onde cada participante atua como cliente e servidor de recursos. A descoberta e representação desses recursos, juntamente com a sua proteção, são os principais desafios que envolvem esse tipo de computação. Este artigo define uma arquitetura de controle de acesso baseada em papéis específica para as redes Peer-to-Peer e busca, assim, contribuir para a escrita de aplicações colaborativas mais robustas. O protótipo implementado valida o modelo e provê meios para que cada nó da rede Peer-to-Peer gerencie a política de segurança dos seus recursos.*

1. Introdução

As redes Peer-to-Peer são sistemas distribuídos sem controle centralizado ou organização hierárquica, nas quais o programa que é executado em cada elemento é equivalente em funcionalidade [Luca Caviglione, 2004]. Esses sistemas possibilitam que os usuários sejam, além de consumidores de recursos, os próprios responsáveis por disponibilizá-los. Percebe-se então, as diversas diferenças desse modelo de computação em relação ao esquema cliente-servidor amplamente implementado na internet.

A computação em redes colaborativas exige que muitos paradigmas existentes nas aplicações cliente-servidor atuais sejam repensados afim de serem adaptados a este novo modelo. Esse é o caso da segurança computacional, a qual precisa prover as propriedades confidencialidade, disponibilidade, integridade [Carl E. Landwehr, 2001] também às redes Peer-to-Peer. Este artigo define uma arquitetura de controle de acesso para redes colaborativas chamada **P2P-Role** e possibilita que cada elemento administre individualmente a política de proteção sob seus

recursos. O modelo de controle de acesso adotado é o RBAC[1] [Elisa Bertino, 2003, Sabrina Vimercati e Stefano Paraboschi e Pierangela Samarati, 2003] pois, além de ser um modelo sólido e aceito, é flexível, já que nele as permissões não são designadas diretamente aos usuários, mas indiretamente através dos papéis.

O artigo descreve também o desenvolvimento de uma aplicação Peer-to-Peer real. Essa aplicação segue os princípios de segurança citados na arquitetura de controle de acesso elaborada e objetiva avaliar os efeitos da união da modelo RBAC com as redes colaborativas.

Entre as motivações que impulsionaram esta pesquisa estão o rápido crescimento das redes colaborativas, que expandiram suas funcionalidades para além do compartilhamento de arquivos, e o artigo publicado por [Neil Daswani e Hector Garcia-Molina, 2003] (*Stanford Peers Group*) , o qual relaciona os problemas de segurança ainda não solucionados nas redes Peer-to-Peer. Esse trabalho classifica as necessidades de segurança das comunidades Peer-to-Peer em quatro grupos – disponibilidade, autenticidade de conteúdo, anonimidade e controle de acesso – e apresenta o que falta em cada um deles. Em especial, o modelo P2P-Role preocupa-se com o último grupo de requisitos de segurança, sobretudo com a construção de uma arquitetura de controle de acesso que propicie a cada usuário da rede estabelecer políticas de autorização sob seus recursos.

São mostrados na pesquisa aspectos de segurança internos a cada elemento da rede; porém outras questões são igualmente significativas, como a criptografia do canal de comunicação entre os pontos e a verificação da identidade de cada participante P2P (evitar que um nó malicioso assuma o lugar de outro indevidamente). Esses temas estão inclusos na plataforma JXTA[William Yeager e Joseph Williams, 2002] – plataforma para o desenvolvimento de aplicações P2P que utiliza um conjunto de protocolos (descoberta de dados, nomeação dos nós, etc) que permitem que cada dispositivo da rede virtual colabore com os restantes. O JXTA não menciona um método específico para a proteção do acesso aos recursos da rede Peer-to-Peer. Um dos objetivos da arquitetura P2P-Role é possibilitar que suas idéias e resultados sejam adotados por outras plataformas e projetos que contemplam o tema Peer-to-Peer.

O artigo está organizado em 5 seções. A seção 2 é responsável por exibir alguns aspectos teóricos sobre Peer-to-Peer, os quais objetivam transmitir idéias necessárias à compreensão deste documento. Na seção 3 é apresentado a arquitetura de controle de acesso para redes colaborativas P2P-Role e suas particularidades. A seção 4 descreve a aplicação construída para legitimar a arquitetura desenvolvida e aborda como o modelo RBAC se integra a cada nó P2P. O artigo encerra na seção 5 com a conclusão, a qual reúne as principais idéias e resultados da pesquisa, além de citar os possíveis complementos sobre ela, a cargo de trabalhos futuros.

2. Aspectos Teóricos

A definição de sistemas Peer-to-Peer não é consenso na comunidade científica, principalmente devido às semelhanças entre computação colaborativa P2P, computação em grid

[1] *Role Based Access Control.*

e aglomerado de computadores (*cluster*) [Domenico Talia e Paolo Trunfio, 2003]. Visto isso, percebe-se a necessidade de haver, antes da descrição do modelo de controle de acesso para redes P2P, uma contextualização sobre as idéias fundamentais que circundam esse paradigma de computação distribuída. O método de controle de acesso RBAC possui um significante grau de amadurecimeto na comunidade científica (padrão pelo NIST –*National Institute of Standards and Technology*– desde 2001) e sua descrição está inclusa ao pacote da aplicação P2P-Role (veja o endereço web na seção 4.1).

2.1. Redes Colaborativas Peer-to-Peer

O termo P2P tem sido aplicado para um grande conjunto de tecnologias. Em geral, o P2P descreve um ambiente onde computadores conectam-se uns aos outros em um sistema distribuído, o qual não usa um ponto centralizador para rotear ou conectar o tráfego entre os elementos [William Yeager e Joseph Williams, 2002]. Por minimizar o papel dos elementos centralizadores, os sistemas P2P tendem a ser imunes à censura, monopólios, regulamentos e outros exercícios atribuídos às autoridades centralizadoras [Philip E. Agre, 2003].

As redes colaborativas P2P emergiram a partir de 2000, principalmente com as aplicações para distribuição de arquivos. Os equipamentos conectados no ambiente colaborativo formam uma rede virtual sobre a rede de dados subjacente (IP, no caso da internet). Sistemas P2P trazem conectividade para as bordas da rede, permitindo que qualquer equipamento conectado se comunique e colabore com os demais [Djamel Sadok, 2003].

O modelo de rede colaborativo proporcionado pela computação aos pares tem sido apontado como uma das maiores transformações da internet nos últimos anos e com tendência de expansão [Alfred W. Loo, 2003]. Esse modelo é atrativo por várias razões: os sistemas P2P oferecem meios para agregar e utilizar recursos geograficamente distribuídos; o custo para a criação dos ambientes colaborativos é baixo e não requer investimento maciço em equipamentos; a natureza descentralizada desses sistemas torna-os inerentemente robustos a falhas ou ataques intencionais; alta escalabilidade para tratar do crescimento de elementos que se juntam a rede P2P.

Os principais modelos de computação Peer-to-Peer são o híbrido e o puro. No modelo híbrido, utilizado pelo Napster (aplicação para o compartilhamento de músicas), existe um ponto centralizador responsável por catalogar e indexar as informações localizadas nos membros da rede. Nesse esquema, o elemento contata o servidor central quando está a busca de uma informação. Logo após receber os resultados do servidor, o participante da rede decide qual resposta lhe parece ser a melhor e abre uma conexão diretamente com o provedor do recurso, estabelecendo a computação Peer-to-Peer. A utilização de um meio centralizador pode ser positiva, pois diminui a complexidade da computação nos pontos. Porém ele será um elemento central de falha – se ele cair a rede fica sem funcionalidade.

Já no modelo puro de computação colaborativa, as entidades comunicam-se umas com as outras diretamente, também para tomar as decisões de controle. O problema de organizar uma base de informações distribuída e robusta, juntamente com métodos para a localização de conteúdos, formam o núcleo de qualquer sistema P2P puro. Os algoritmos mais recentes para a estruturação de dados nesse modelo usam uma tabela de hash distribuída para registrar os recursos. Dessa forma, a cada item é designado um identificador

único e, quando um elemento deseja encontrar uma informação, ele calcula o identificador dessa informação e repassa a requisição para seus vizinhos[2]. Quando um vizinho recebe um pedido ele verifica se não contém a informação requisitada; caso possuir, responde prontamente ao requisitor. Do contrário, ele reenvia o pedido a seus vizinhos, executando o mesmo protocolo [Hari Balakrishnan e David Karger e Robert Morris, 2003].

O compartilhamento de arquivos e outros recursos entre os pares de entidades abre um enorme potencial para hackers, vândalos e ladrões subverterem os equipamentos envolvidos. A distribuição de músicas na internet pode não precisar de requisitos de segurança, porém em ambientes P2P corporativos a proteção do canal de comunicação e dos próprios elementos é indispensável. Em particular, a autorização possibilita que cada entidade administre as permissões sob seus recursos, e assim coloque em prática a política de controle de acesso que lhe convier. O modelo de autorização para redes colaborativas P2P é o tema da seção 3.

3. Arquitetura de Controle de Acesso P2P-Role

Nas redes Peer-to-Peer cada elemento disponibiliza seu conjunto de recursos para os demais membros da rede virtual. Normalmente, esses recursos estão acessíveis a todos e não existe distinção entre elementos do sistema P2P. Porém, em algumas situações os nós podem desejar limitar o acesso sob determinadas informações, sendo necessária a adoção de um mecanismo de controle de acesso que contemple o paradigma de ambientes colaborativos P2P.

Este artigo explora esse fato e apresenta um esquema de controle de acesso baseado no RBAC para redes Peer-to-Peer. A Figura 1 exibe os elementos da rede P2P, os quais são compostos por uma aplicação específica – igual em todos os nós – e pelo módulo de autorização do RBAC; e também o servidor central responsável por catalogar todos os recursos dispostos na rede. Duas questões são relevantes nessa figura: ela informa que o modelo de rede P2P será o híbrido (embora a arquitetura de autorização também se adapte facilmente às redes P2P puras) e que cada elemento será o único responsável pela atribuição de permissões aos seus próprios recursos – a decisão de autorização não é distribuída.

Segundo [Pascal Fenkam e Schahram Dustdar e Engin Kirda, 2002], existe algumas características indispensáveis em uma arquitetura de controle de acesso voltado às redes Peer-to-Peer. Esse estudo apresenta o modelo DUMAS (*Dynamic User Management and Access Control*), utilizado para gerência de permissões em ambientes P2P com dispositivos móveis, e informa as seguintes propriedades dos sistemas de controle de acesso: descentralização do controle, suporte a vários protocolos de autorização, interface para administração em cada participante da rede e escalabilidade. O **P2P-Role** buscou preencher esses requisitos no processo de elaboração da arquitetura de controle de acesso para sistemas Peer-to-Peer.

A anonimidade é uma característica importante das aplicações Peer-to-Peer. Ela permite que exista uma troca de informações entre os usuários da rede P2P sem a identificação explícita dos autores dos recursos, dos nomes dos

[2]Cada elemento da rede possui uma lista de endereços IP de alguns vizinhos. A distribuição das requisições pelo nó assemelha-se ao processo de *flooding*.

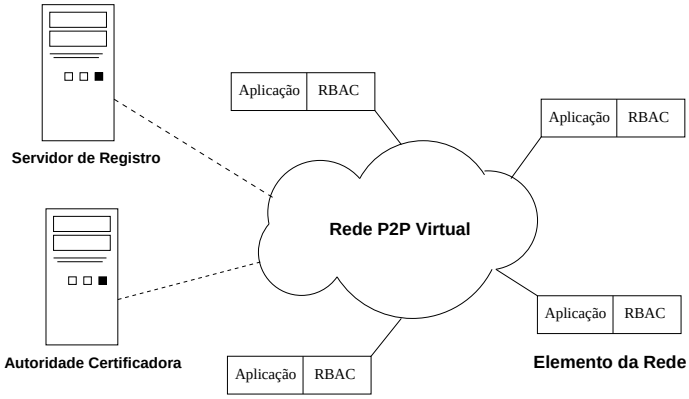

Figura 1: Rede Colaborativa P2P híbrida

nós envolvidos na comunicação e do usuário receptor da informação. Conforme [Neil Daswani e Hector Garcia-Molina, 2003], essas peculiaridades são difíceis de serem atingidas quando os mecanismos de proteção de redes colaborativas P2P são mais rígidos. Esse é o caso do controle de acesso, onde será necessário uma sistemática para identificar (perde anonimidade) claramente nomes de usuários, recursos e nós na rede Peer-to-Peer.

A aplicação P2P desse modelo de rede, quando chamada, irá conectar-se automaticamente com o catalogador de conteúdo e passará a ele o título – na forma de identificador universal – de cada contribuição do elemento para a rede colaborativa. Não é informado nenhum dado sobre as permissões associadas aos recursos. Independente do nível de proteção que o elemento dá a cada recurso, todos devem ser publicados junto ao servidor de registro.

No momento que determinado participante da rede deseja encontrar alguma informação, ele pesquisa o servidor central e este retorna os endereços internet (IP) dos elementos que contém tal conteúdo (ver Figura 2). Nesse momento, o nó que está a procura da informação verá que ela existe em sua rede, mas não saberá se poderá ter acesso completo a ela.

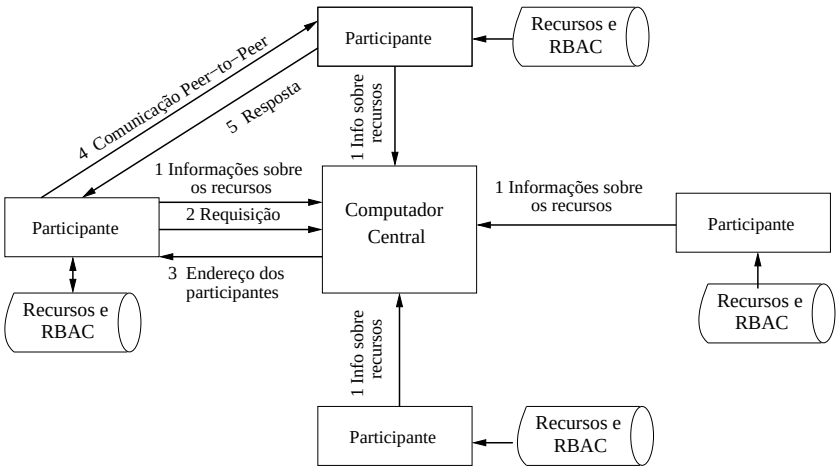

Figura 2: Procura por conteúdo na rede P2P

Cada elemento possui, por padrão, o usuário “anonimo” e o papel “comum”[3]. O usuário “anonimo” está exclusivamente ligado ao papel “comum”. Esse papel possui ligado a si, entre outras, a permissão de observar todos os títulos dos recursos oferecidos pelo elemento. Os direitos relacionados aos recursos oferecidos sem restrições (acesso livre) deverão estar combinados necessariamente ao papel “comum”.

O usuário, logo após autenticar-se com a aplicação P2P remota – via próprio *login* ou através do “anonimo” –, observará os ítens disponibilizados pelo elemento. A cada recurso um número é atribuído e é através dele que o usuário o acessa (acessar pode significar transferir informações para o seu computador pessoal, ver conteúdos na tela, entre outros). Quem permite ou nega os acessos é o monitor de referência, o qual intercepta a chamada de autorização e ativa o módulo RBAC passando o nome do usuário e o recurso alvo como parâmetros.

O RBAC utiliza o modelo baseado em papéis configurado previamente pelo usuário e verifica se a atitude será de negação ou de concessão. No caso de aprovação, o RBAC devolve o código “sucesso” ao monitor de referência e este permite o acesso ao recurso. Na ocorrência de insucesso, o usuário receberá uma mensagem de advertência e o motivo do bloqueio à informação.

A aplicação abre a possibilidade do usuário interagir com o elemento da rede, afim de ser associado/incluso em um papel que possua permissões mais elevadas. Nessa parte da aplicação P2P são requeridos 4 ítens: nome completo, contato (e-mail), motivo por que almeja o acesso ao recurso e, caso possuir, o certificado digital expedido por uma autoridade certificadora de consenso mútuo dos integrantes da rede virtual.

A resposta do elemento ao pedido de inclusão/alteração em seu modelo de autorização RBAC pode ser manual ou automática. Ela será manual no caso do próprio usuário responsável pelo elemento da rede examinar se concorda com os dados. A opção automática refere-se à investigação dos dados e certificado digital por um processo automatizado, o qual segue alguma política de funcionamento[4]. O resultado será informado por e-mail ao usuário, anexo ao seu *login* e senha.

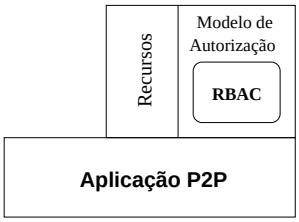

Figura 3: Módulo RBAC unido à aplicação P2P

Muitas vezes, os usuários da rede virtual P2P são conhecidos uns dos outros ou possuem algum vínculo de confiança. Nessas circunstâncias, o responsável pelo elemento da rede pode realizar a manutenção na base de informações do modelo RBAC diretamente, ou seja, sem a necessidade de trocas de certificados e credenciais.

[3]O usuário “anonimo” não possui senha e indica baixo privilégio.

[4]Um exemplo de política é aceitar a requisição de todos os usuários que contenham no e-mail o domínio Y e possuam o certificado válido.

Com relação ao modelo RBAC, como exposto outrora, cada participante da rede tem uma interface de administração de usuários, papéis e permissões, a qual lhe permite organizar sua política de proteção de recursos. Como o modelo de autorização está acoplado à aplicação sob a forma de módulo (espécie de *plug in*), o usuário pode mudar de método de controle de acesso para outro que considerar mais conveniente, mantendo uniforme apenas a chamada feita pelo monitor de referência ao modelo de autorização (observe a Figura 3).

4. Aplicação P2P-Role

Esta seção apresenta o desenvolvimento do protótipo P2P-Role[5], o qual baseia seu método de controle de acesso na arquitetura descrita na seção 3. A discussão sobre essa aplicação é separada em três etapas. Primeiramente, têm-se as principais características que envolvem o protótipo, como o ambiente onde foi construído e as decisões de projeto realizadas. Logo após, têm-se o detalhamento do funcionamento do programa P2P e a especificação das mensagens trocadas entre os membros da rede colaborativa. Na subseção 4.3, por final, encontra-se a definição da interface de administração do sistema RBAC, a qual integra o protótipo P2P-Role.

4.1. Características do Protótipo P2P-Role

A aplicação Peer-to-Peer elaborada é composta de duas partes distintas. A primeira é responsável por gerenciar os módulos cliente e servidor, os quais compõem o núcleo de cada entidade que participa da rede colaborativa. Já a outra parte possui o objetivo de fornecer ao usuário-administrador de um nó P2P uma forma simples de realizar a manutenção sob as tabelas do sistema RBAC e, desta forma, pôr em prática a política de controle de acesso aos seus recursos.

A linguagem de programação utilizada na escrita do protótipo foi o Java, já que este ambiente de desenvolvimento oferece, através do byte-code (código executável de uma máquina virtual Java), um alto nível de portabilidade de código, suporte a multithread nativo (não existe a necessidade de bibliotecas adicionais) e é robusto e seguro [Marinho Barcelos, 2002]. As tecnologias Java presentes nos módulos elaborados são as seguintes: JDBC, o qual possibilita o acesso uniforme a banco de dados diferentes; multithread, para proporcionar a execução de vários fluxos de código simultaneamente e Applet, que viabiliza a composição de programas Java orientados à web.

O protótipo P2P-Role é composto por sete classes (observe seus nomes e propósitos na Tabela 1). As classes `ControlClient` e `ControlServer` estendem a classe `Thread`, ou seja, elas se comportam como fluxos de execução independentes. A classe `Admin` é a única que age como Applet e contribui, dessa forma, para que o responsável pelo nó P2P realize mudanças nos usuários, papéis e permissões aos recursos remotamente.

Em uma aplicação P2P, comumente, o usuário deseja apenas fazer buscas e não disponibiliza nenhum recurso, ou ainda o inverso, onde pretende-se dar exclusividade à oferta de recursos. O protótipo desenvolvido possui a característica de executar o módulo

[5]O protótipo desenvolvido possui o mesmo nome da arquitetura e está disponível para desenvolvedores e cientistas no endereço `http://www.lrg.ufsc.br/~rrighi/p2p_role.tar.gz`.

Classe	Objetivo
`Application`	É a classe principal da aplicação. É ela que cria e comanda os fluxos cliente e servidor.
`ControlClient`	Este fluxo de execução realiza a função de cliente e requisita recursos ao fluxo servidor.
`ControlServer`	Recebe os chamados dos clientes, processa-os e devolve a resposta apropriada.
`Auth`	Possui os métodos para avaliar a autenticidade de um usuário e para verificar quais usuários têm acesso garantido aos recursos. Faz chamadas JDBC às tabelas que compõe o RBAC.
`Message`	Especifica as partes que constituem uma mensagem trocada entre as entidades P2P.
`Configuration`	Possui variáveis que definem nomes de arquivos de registro e personalizações da aplicação P2P.
`Admin`	Esta classe é um Applet no qual o responsável pelo nó P2P realiza a manutenção nas tabelas que compõe o modelo de controle de acesso RBAC.

Tabela 1: Classes que compõe a aplicação Peer-to-Peer

cliente e servidor juntos ou um de cada vez separadamente. Para configurar esse tópico, o responsável pelo nó P2P deve editar a classe `Configuration` e alterar a variável apropriada.

Outras características existentes são a geração de um registro com todas as conexões bem sucedidas no módulo servidor e a possibilidade de um usuário deixar uma mensagem para a entidade a qual ele está se conectando. Como descrito na seção 3, nestas mensagens, por exemplo, um usuário pode solicitar sua inclusão em um papel que detenha maiores privilégios, requisição de mudança de senha ou puramente enviar um comentário qualquer à outra ponta da conexão. Novamente a classe `Configuration` é a que concentra a especificação dos nomes dos dois arquivos que guardarão as informações citadas anteriormente. Percebe-se que os participantes da rede colaborativa, embora possuam o mesmo código Java em execução, podem ser personalizados diferentemente e, assim, refletirem os desejos de seus mantenedores.

4.2. Funcionamento do Protótipo P2P-Role

Para colocar a aplicação Peer-to-Peer em funcionamento deve-se executar a classe `Application` e passar como parâmetro o endereço internet do nó P2P que possui os recursos cobiçados. A função dessa classe é ativar os fluxos de execução cliente e servidor (por padrão, os dois são acionados) e ficar esperando até que eles terminem.

O fluxo fornecedor de recursos abre um soquete do tipo servidor e espera por conexões na porta especificada na classe `Configuration`. Já o fluxo cliente conecta-se ao fluxo servidor do nó alvo e requisita o nome de usuário e senha para posteriormente enviá-los para validação. Quando o pedido de autenticação chega ao fluxo servidor, é criado um objeto da classe `Auth` e chamado seu método para verificação de usuários, repassando a mensagem recém recebida. Este objeto, através de pesquisa realizada nas

tabelas que formam o sistema RBAC, retorna se o usuário é válido ou não. Nesse momento, o fluxo servidor envia ao fluxo cliente um objeto `Message` com a resposta da autenticação.

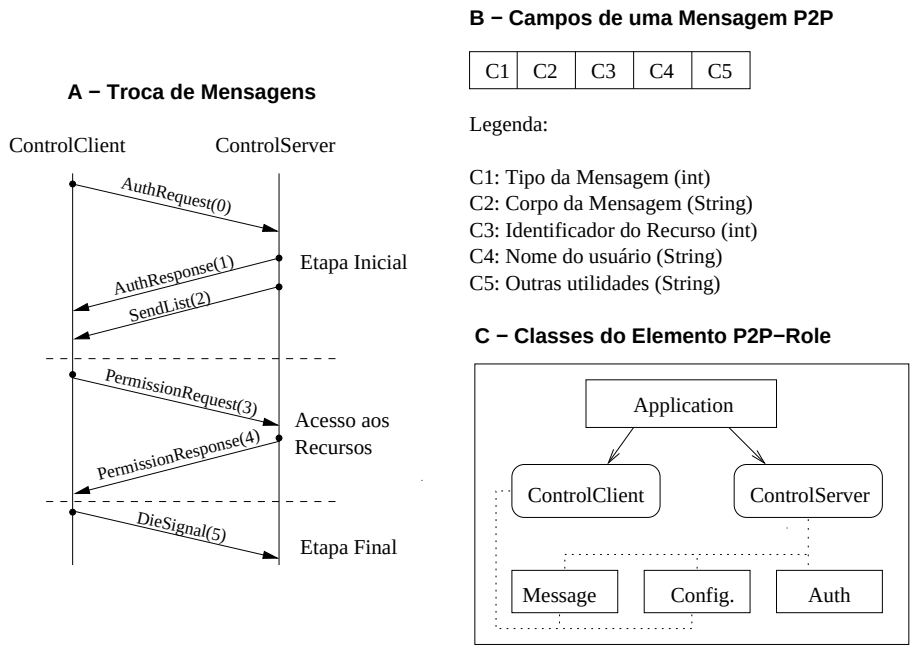

Figura 4: Mensagens trocadas entre os membros da rede P2P

Existem duas possibilidades possíveis de resposta de autenticação: sucesso ou insucesso. Caso acontecer insucesso, o fluxo cliente fecha a conexão e volta a requisitar as credenciais do usuário e o fluxo servidor retorna a escutar na porta alocada. A opção sucesso irá deixar o fluxo cliente esperando pela vinda de uma lista contendo os recursos disponíveis no servidor. Este, por sua vez, realiza a gravação em disco de algumas informações características da sessão aberta, chama o objeto `Auth` para criar a lista apropriada e, logo após, a envia ao outro extremo da conexão. Nesse instante, os dois nós P2P que se comunicam estão cientes que a autenticação ocorreu bem e que a etapa inicial da conversa entre eles foi cumprida.

Na próxima etapa da conexão, a aplicação P2P que está realizando a função de cliente faz pedidos por recursos e espera o retorno de seu par na comunicação. A lista de recursos que o cliente recebeu é composta por um identificador do recurso e seu nome. O usuário irá digitar o identificador desejado e o cliente envia um objeto `Message` ao servidor contendo o pedido de permissão. Para verificar se o usuário ativo pode acessar determinado recurso, ele invoca o objeto da classe `Auth` e seu método apropriado. Esse objeto usa a seguinte lógica para verificar a permissão: pesquisa todos os papéis associados com o recurso desejado e, logo após, observa se o nome do usuário ativo está ligado a algum desses papéis "com acesso garantido". Se sim, o usuário encontra a informação procurada, que no caso desse protótipo é um campo de dados da tabela "direito" do modelo RBAC (observe todas as mensagens trocadas na Figura 4).

As outras possibilidades que o usuário possui, além de requisitar recursos, são as de deixar mensagens escritas no servidor (processo descrito na subseção 4.1) e fechar a conexão ativa. Se o item escolhido é o fechar (identificado pelo número 999), o cliente

envia um objeto `Message` informando que vai "morrer"; fato que possibilita que ambas partes fechem juntas a conexão ativa e nenhum problema aconteça com os soquetes. A aplicação P2P, nesse ponto, questiona o usuário se ele pretende abrir uma conexão diferente com outro nó da rede colaborativa ou terminar o fluxo cliente. Caso outra conexão for aberta, todo o ciclo de conversação recomeça.

A aplicação P2P-Role basea-se no endereço IP para identificar os participantes da rede P2P. Essa característica fere o princípio da anonimidade da comunidade Peer-to-Peer, porém favorece o mantenedor da aplicação, que saberá através dos registros do P2P-Role qual a procedência dos usuários que buscaram seus recursos. O trabalho desenvolvido por [Sergio Marti e Hector Garcia-Molina, 2003] aborda o tema anonimidade em sistema colaborativos e cita métodos alternativos ao endereço IP para serem usados na identificação de usuários[6]. Um desses métodos é o uso de certificados digitais em conjunto com uma infraestrutura de chaves públicas.

4.3. Administração do Modelo de Controle de Acesso RBAC

O sistema de controle de acesso existente em cada nó da rede Peer-to-Peer é o RBAC. No protótipo construído este sistema foi idealizado em seis tabelas do banco de dados MySQL (versão 4.01), com destaque para as seguintes: usuário, papel e direito. As tabelas restantes são o resultado do relacionamento que há entre os usuários e papéis e entre os papéis e os direitos. O módulo de administração é o responsável por gerenciar o conteúdo existente nessas tabelas e assegurar que os propósitos de segurança planejados para determinado membro da rede sejam obedecidos.

Para realizar tarefas administrativas é necessário uma autenticação prévia. Através dessa etapa são verificados dois requisitos: se o nome do usuário e senha conferem e se este usuário está associado ao papel "Administrador". Caso essas exigências sejam satisfeitas, o usuário-administrador tem acesso à barra de menus do programa, a partir do qual ele fará suas tarefas.

A barra de menus é composta de sete ítens: usuário, papel, direito, usuário-papel, papel-direito, hierarquia e ajuda. Seus nomes informam ao administrador como atingir o seu propósito, ou seja, se for necessário incluir um novo usuário, o menu "usuário" deve ser procurado.

Os menus Usuário, Papel, Direito possuem os mesmos nomes de opções. Os três possibilitam as tarefas adicionar, remover e listar registros das tabelas do RBAC. Os outros menus são responsáveis por estabelecer as associações entre os registros do modelo de acesso (por exemplo, para associar o usuário "anonimo" com o papel "comum" deve-se dirigir ao menu Usuário-Papel) e por estabelecer as relações de "pai e filho" entre os papéis (por exemplo, papel Funcionário é pai do papel Professor). O último menu chama-se Ajuda e disponibiliza informações sobre o RBAC, além de conter o item "Fechar". Ao executar esse procedimento, o administrador encerra sua sessão de trabalho e a tela de autenticação novamente fica a espera de um usuário[7].

[6] A anonimidade não é o objetivo principal do P2P-Role, motivo pelo qual esta aplicação utiliza um mecanismo simples (IP) para a identificação dos nós da rede.

[7] Os *snapshots* da aplicação **p2p-role**, as instruções para a geração das tabelas do RBAC e um exemplo de uso do sistema também estão disponíveis no endereço web descrito na seção 4.1.

5. Conclusão

As redes Peer-to-Peer possibilitam a colaboração entre os elementos de rede e seu potencial é imenso, desde o compartilhamento de recursos até o comércio eletrônico entre comunidades. As aplicações P2P precisarão ser seguras e confiáveis para atingirem as corporações e ambientes críticos. Nesse sentido, este trabalho colabora para o robustez dessas aplicações, já que apresenta uma forma eficaz de cada participante proteger o acesso às suas informações.

A adoção do esquema de controle de acesso baseado em papéis (RBAC) para preencher o modelo de autorização em redes Peer-to-Peer está entre as principais decisões de projeto desta pesquisa. O RBAC mostrou ser flexível e de fácil entendimento, o que favorece sua manutenção pelos usuários, já que não precisarão possuir conhecimentos específicos de computação para designarem permissões aos seus recursos.

A opção por um monitor de referência – outra decisão de projeto – para intermediar o processo de autorização, auxilia para que o RBAC seja implementado como módulo (conjunto de programa separado do núcleo). Nessa condição, a principal vantagem é a diminuição da complexidade da aplicação P2P propriamente dita, que poderá orientar seus esforços para seu objetivo (ex: compartilhamento de documentos) e "tercerizar" a autorização a um módulo específico para essa função.

O protótipo P2P-Role auxilia para o entendimento dos conceitos existentes na arquitetura de controle de acesso estabelecida e na visualização de suas características. Seu modelo de classes e métodos pode servir para nortear a escrita de outras aplicações P2P, principalmente aquelas que exigem segurança. O P2P-Role possibilitou verificar que a implantação de um controle de acesso na comunidade P2P pode prejudicar a anonimidade inerente a esses sistemas distribuídos. Assim, projetistas de aplicações colaborativas devem perceber que o reforço dos mecanismos de segurança nessas redes geralmente vêm acompanhado de restrições nas principais características desses sistemas.

Finalmente, como sugestões de trabalhos futuros, indica-se duas tarefas. A primeira é dar suporte a infraestrutura de chave públicas na aplicação P2P-Role, assim com acontece no trabalho elaborado pelos pesquisadores [Wooyoung Kim e Sven Graupner e Akhil Sahai, 2002]. Dessa maneira os mecanismos de autenticação e autorização já existentes no P2P-Role podem unir-se a outros como criptografia e certificados digitais. A outra tarefa envolve pesquisar os mecanismos de proteção que foram incluídos na computação em grid e verificar as possíveis portabilidades de idéias desses ambientes para o paradigma de computação aos pares (P2P) (veja [Mike Surridge e Colin Upstill, 2003]).

Referências

Alfred W. Loo (2003). The Future of Peer-to-Peer Computing. *Communications of ACM*, 46(9):57–61.

Carl E. Landwehr (2001). Computer security. *International Journal of Information Security*, 1(1):3–13.

Djamel Sadok (2003). Computação Colaborativa (P2P). Grupo de Trabalho da Rede Nacional de Pesquisa. Disponível em http://www.rnp.br/_arquivo/gt/2003/p2p.pdf.

Domenico Talia e Paolo Trunfio (2003). Toward a Synergy Between P2P and Grids. *IEEE Internet Computing*, July/August 2003 issue:94–96.

Elisa Bertino (2003). RBAC Models – Concepts and Trends. *Computer and Security*, 22(6):511–514.

Hari Balakrishnan e David Karger e Robert Morris (2003). Looking Up Data in P2P Systems. *Communications of ACM*, 46(2):43–48.

Luca Caviglione (2004). The "Dark Side" and The "Force" Of The Peer-to-Peer Computing Saga. *Peer-to-Peer Journal*, 1(4):1–11. http://www.p2pjournal.com.

Marinho Barcelos (2002). Programação paralela e distribuída em java. *In: Escola Regional de Alto Desempenho - ERAD*, pages 179–181. ISBN: 8588442167, São Leopoldo, RS.

Mike Surridge e Colin Upstill (2003). Grid Security: Lessons for Peer-to-Peer Systems. *In: Proceedings of the Third International Conference on Peer-to-Peer Computing (P2P'03)*, pages 2–7. Linköping, Sweden.

Neil Daswani e Hector Garcia-Molina (2003). Open Problems in Data-Sharing Peer-to-Peer Systems. *In: 9th International Conference on Database Theory (ICDT 2003)*, pages 1–15. Siena, Italy.

Pascal Fenkam e Schahram Dustdar e Engin Kirda (2002). Towards an Access Control System for Mobile Peer-to-Peer Collaborative Environments. *In: Proceedings of Eleventh IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'02)*, pages 95–100.

Philip E. Agre (2003). P2P and the Promise of Internet Equality. *Communications of the ACM*, 46(2):39–42.

Sabrina Vimercati e Stefano Paraboschi e Pierangela Samarati (2003). Access control: principles and solutions. *Software —Practice & Experience*, 33(5):397–421. ISSN:0038-0644.

Sergio Marti e Hector Garcia-Molina (2003). Identity Crisis: Anonymity vs. Reputation in P2P Systems. *In: Proceedings of the Third International Conference on Peer-to-Peer Computing (P2P'03)*, pages 134–141.

William Yeager e Joseph Williams (2002). Secure Peer-to-Peer Networking: The JXTA Example. *IEEE IT Professional*, 4(2):53–57.

Wooyoung Kim e Sven Graupner e Akhil Sahai (2002). A secure platform for peer-to-peer computing in the internet. Technical Report HPL-2001-324, Hewlett Packard Laboratories. http://www.hpl.hp.com/techreports/2001/HPL-2001-324.pdf.