MARCELO MESSORA MIRANDA

SEGURANÇA DA INFORMAÇÃO EM REDES PEER-TO-PEER (P2P)

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

LAVRAS
MINAS GERAIS – BRASIL
2009

MARCELO MESSORA MIRANDA

SEGURANÇA DA INFORMAÇÃO EM REDES PEER-TO-PEER (P2P)

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Área de Concentração:

Redes de Computadores

Orientador:

Prof. DSc. Luiz Henrique Andrade Correia

LAVRAS
MINAS GERAIS - BRASIL
2009

MARCELO MESSORA MIRANDA

SEGURANÇA DA INFORMAÇÃO EM REDES PEER-TO-PEER (P2P)

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Aprovada em 24 de novembro de 2009

_____

Prof. MSc. Reginaldo Ferreira de Souza

_____

Prof. DSc. Rêmulo Maia Alves

_____

Prof. DSc. Luiz Henrique Andrade Correia
(Orientador)

LAVRAS
MINAS GERAIS - BRASIL
2009

DEDICATÓRIA

Dedico este trabalho a meu pai, Lamartine, à minha mãe, Marly e a meu irmão, Luciano.

AGRADECIMENTOS

Agradeço toda a minha família e amigos que me deram o apoio necessário durante a minha formação acadêmica. Meus sinceros agradecimentos a todos vocês, que tanto me incentivaram nos momentos mais importantes da minha vida. Agradeço também a todos os professores que me passaram o conhecimento necessário para a minha formação. Muito Obrigado!

RESUMO

O grande avanço nas áreas de telecomunicações e de redes de computadores, aliado à grande redução de custos dos recursos computacionais, motivou a proliferação das redes por todos os segmentos da sociedade. Nesse contexto, surgiu uma classe de sistemas e aplicações que utilizam recursos distribuídos para executar funções críticas de um modo descentralizado, as aplicações P2P. Com a expansão do interesse de uso de aplicações P2P, especialmente em ambientes corporativos, a segurança desse tipo de sistema acaba sendo um dos principais entraves para a sua franca utilização. Nesse sentido, o objetivo deste trabalho é apresentar e discutir dois temas que tem recebido a atenção tanto da comunidade científica como da indústria: Redes Peer-to-Peer (P2P) e Segurança de Informação. Ao final do trabalho pode-se constatar que o sistema de redes P2P são muito atrativos e difundidos mundialmente por vários motivos. São sistemas escaláveis, ou seja, não possuem um ponto central de falhas ou gargalo na forma de um servidor central; são redes que resistem melhor a ataques intencionais como os de negação de serviço assim como tem o poder de atrair um grande número de usuários em função dos benefícios oferecidos pela coletividade, sem, no entanto, abrir mão da autonomia de seus participantes. Entretanto, apesar de resistir melhor a ataques intencionais, são suscetíveis a falhas de segurança ou apresentam algum tipo de vulnerabilidade. Nesse sentido, é necessário que as aplicações sejam confiáveis e seguras, considerando que cada aplicação P2P possui seus requisitos próprios de segurança e que devem ter métodos de segurança instanciados conforme as suas necessidades.

Palavras-chave: peer-to-peer, overlay, segurança da informação, redes descentralizadas.

ABSTRACT

The great progress in the areas of telecommunications and computer networks, combined with the large reduction in cost of computer resources, led to the proliferation of networks for all segments of society. In this context, a class of systems and applications that use distributed resources to perform critical functions in a decentralized way, P2P applications. With the expansion of interest in use of P2P applications, especially in corporate environments, the security of such system has been a major constraint to their use france. Accordingly, the objective of this review was on two themes that cross has received the attention of both the scientific community and industry: Peer-to-Peer (P2P) and Information Security. At the end of the review you can see that the system for P2P networks are very attractive and widely circulated for several reasons. Systems are scalable, or do not have a central point of failure or bottleneck in the form of a central server, networks that are better to resist intentional attacks such as denial of service and has the power to attract a large number of users in function of the benefits offered by the community, without however, giving the autonomy of its participants. However, despite the best resist intentional attacks are susceptible to security flaws or show some kind of vulnerability. Therefore, is necessary that applications are reliable and secure, considering that each P2P application has its own security requirements and must be instantiated as methods for their security needs.

SUMÁRIO

LISTA DE FIGURAS

Pág.

LISTA DE TABELAS

Pág.

1. INTRODUÇÃO

1.1 Contextualização

O grande avanço nas áreas de telecomunicações e de redes de computadores, aliado à grande redução de custos dos recursos computacionais, motivou a proliferação das redes por todos os segmentos da sociedade. Isso trouxe consigo um aumento na diversidade de recursos e serviços oferecidos, o que, por sua vez, tem aumentado a complexidade das redes. Não bastassem esses fatos, os sistemas computacionais ainda apresentam grande heterogeneidade dos padrões de redes, sistemas operacionais, equipamentos, etc (Vilanova, 2006).

Nesse contexto, surgiu uma classe de sistemas e aplicações que utilizam recursos distribuídos para executar funções críticas de um modo descentralizado, as aplicações *Peer-to-peer* (P2P). Aplicações de compartilhamento de arquivos, serviços de mensagens instantâneas (*Instant Messaging*), processamento distribuído, *webcaching*, jogos, disseminação de conteúdo, backup distribuído e telefonia IP entre outros são exemplos de aplicações que envolvem P2P.

Os serviços introduzidos pelos sistemas P2P apresentam características inovadoras, e sua utilização vai além de simples compartilhamento de arquivos ou computação distribuída, eles também

podem ser utilizados para ajudar a resolver problemas de outras áreas críticas, como por exemplo, a gerência de redes (Granville et al., 2005).

Antes do surgimento das aplicações P2P, a utilização da Internet por usuários comuns consistia em uma rede praticamente cliente/servidor, com diversos clientes requisitando conteúdo e serviços publicados por servidores com endereços fixos registrados no DNS (Domain Name System). Com a inovação tecnológica e a popularização de diversos dispositivos com acesso a Internet, passamos a contar com uma rede na qual esses dispositivos também são capazes de fornecer recursos, porém nem sempre estarão conectados ou utilizando os mesmos endereços. O aumento da largura de banda, e a maior disponibilidade de acesso fizeram com que os usuários sentissem a necessidade de uma rede mais colaborativa, na qual a busca de conteúdo e serviços, além da interação com outros usuários, deixa de ser privilégio de alguns servidores. Essas talvez tenham sido as motivações para o constante crescimento das redes e aplicativos P2P (Truelove, 2001).

Esses aplicativos têm como objetivo compartilhar os custos de manutenção das aplicações entre os usuários (por exemplo, nas aplicações de troca de arquivos, o espaço de armazenamento é fornecido pelos usuários), possibilitar a agregação de recursos e a interoperabilidade, aumentar a autonomia dos sistemas, garantir o anonimato e a privacidade dos usuários e oferecer suporte a ambientes dinâmicos, onde os dispositivos entram e saem constantemente. A independência de cada nó da rede traz características muito desejadas nas redes atuais, como a descentralização, a escalabilidade e a tolerância à falhas (Vilanova, 2006).

1.2 Motivação

Ao almejar que redes P2P sejam amplamente adotadas, elas precisam estar protegidas contra a ação de nós maliciosos. Esses podem fornecer, propositalmente, respostas incorretas a requisições tanto no nível de aplicação quanto no de rede. No primeiro caso, retornando informações não verdadeiras em resposta a uma busca, na tentativa de censurar o acesso a determinados objetos. No segundo, fornecendo informações falsas sobre rotas, visando particionar a rede.

1.3 Definição do Problema

Com a expansão do interesse de uso de aplicações P2P, especialmente em ambientes corporativos, e diante dos freqüentes ataques sofridos pelo sistema, a segurança desse tipo de sistema acaba sendo um dos principais entraves para a sua franca utilização.

Frente a este problema, o objetivo da segurança, no que tange à informação, é a busca da disponibilidade, confidencialidade e integridade dos seus recursos e da própria informação.

1.4 Objetivo

O objetivo principal deste trabalho é discorrer sobre um tema que tem recebido a atenção tanto da comunidade científica como da

indústria: a importância da Segurança da Informação em Redes *Peer-to-peer* (P2P).

1.5 Organização da Monografia

Este trabalho é organizado em cinco Capítulos, conforme descritos a seguir:

O Capítulo 1 apresenta uma introdução ao tema discutido e o objetivo do trabalho proposto.

No Capítulo 2 é descrita a metodologia utilizada para a elaboração deste trabalho, onde são mostradas formas e técnicas de pesquisa empregadas.

A fundamentação teórica sobre P2P e Segurança da Informação é apresentada no Capítulo 3.

Finalmente, o Capítulo 4 apresenta as considerações finais sobre o assunto abordado neste trabalho.

2. METODOLOGIA

Segundo Gil (1999), a ciência tem como objetivo maior chegar à veracidade dos fatos, e, para que um conhecimento seja considerado científico, torna-se necessário utilizar métodos que sejam aceitos pela comunidade científica. Sendo assim, é fundamental o delineamento de um método científico que forme um conjunto de procedimentos intelectuais e técnicos adotados para se atingir o conhecimento.

A seguir, apresenta-se o método científico que foi utilizado neste trabalho de pesquisa.

2.1 Caracterização da Pesquisa

Nesta etapa são descritos o tipo de abordagem e a perspectiva do estudo, bem como o delineamento e as técnicas de coleta de dados usadas para atingir o objetivo do trabalho.

2.1.1 Abordagem

Utilizou-se neste trabalho uma abordagem qualitativa de estudo. Este método difere, em princípio, do quantitativo à medida que não emprega um instrumental estatístico como base do processo de análise de um problema. Não pretende numerar ou medir unidades ou categorias

homogêneas. Esta abordagem de um problema, além de ser uma opção do investigador, justifica-se, principalmente, por ser uma forma adequada para entender a natureza de um fenômeno social (Richardson,1999).

O autor acrescenta que os estudos que empregam uma metodologia qualitativa podem descrever a complexidade de determinado problema, analisar a interação de certas variáveis, compreender e classificar processos dinâmicos vividos por grupos sociais, contribuir no processo de mudança de determinado grupo e possibilitar em maior nível de profundidade, o entendimento das particularidades do comportamento dos indivíduos.

2.1.2 Perspectiva de Estudo

Estruturou-se o presente trabalho com base no panorama atual sobre os sistemas *Peer-to-peer* e principais aspectos relacionados à segurança da informação. Não houve um delineamento detalhado do tema sob uma perspectiva histórica**.**

2.2 Delineamento da Pesquisa

Para Trivinos (2006), os estudos exploratórios permitem ao investigador aumentar sua experiência acerca de determinado problema.

Michel (2005) acrescenta que essencialmente, o estudo exploratório ou pesquisa bibliográfica é uma fase da pesquisa cujo

objetivo é auxiliar na definição de objetivos e levantar informações sobre o assunto objeto de estudo. Porém, o estudo exploratório ou pesquisa bibliográfica pode ser considerado uma forma de pesquisa, na medida em que se caracteriza pela busca, recorrendo a documentos, de uma resposta a uma dúvida, uma lacuna de conhecimento. Este tipo de pesquisa procura explicar um problema a partir de referências teóricas publicadas em documentos, desta maneira dispensando a elaboração de hipóteses.

Em relação aos propósitos que se sugeriu, este trabalho teve um cunho exploratório, relativo à abordagem detalhada sobre P2P, procurando aumentar a experiência acerca do problema além da tentativa de preencher uma lacuna de conhecimento.

Quanto às formas de investigação existentes, pode-se classificar esta pesquisa como bibliográfica. O levantamento bibliográfico, que é a essência do estudo exploratório, deve ser constituído a partir de materiais já elaborados, fundamentos teóricos assim como livros e artigos científicos.

Por fim, é importante ressaltar que se utilizou, neste trabalho, o levantamento bibliográfico referente ao acervo existente para a coleta de dados, assim como de artigos científicos e outros materiais muitas vezes disponíveis na *Internet*.

No próximo capítulo será abordado a fundamentação teórica sobre redes P2P, para então posteriormente, serem apresentados os principais aspectos associados à segurança da informação.

3. REVISÃO DE LITERATURA

3.1 Sistemas P2P

Conforme Detsch (2005), não existe um consenso na definição exata do que seja uma rede *Peer-to-Peer* (*Peer-to-peer*). A rede *Peer-to-peer*, geralmente abreviado para P2P, é uma arquitetura distribuída de rede composta de participantes que compartilham seus recursos (como poder de processamento, armazenamento em disco ou largura de banda de rede) diretamente à disposição dos outros participantes da rede, sem a necessidade de instâncias de coordenação centrais (tais como servidores ou hosts estáveis) (Schollmeier, 2002).

Milojicic et al. (2002) definem o termo "*Peer-to-peer*" (P2P) como uma classe de sistemas e aplicações que utilizam recursos distribuídos para executar funções críticas de um modo descentralizado. Os recursos incluem poder de processamento, dados, banda, e presença. A função crítica pode ser processamento distribuído, troca de arquivos, comunicação e colaboração, ou serviços de plataforma.

A maior parte dos serviços de Internet são distribuídos utilizando o tradicional modelo cliente/servidor, ilustrado na Figura 3.1. Nesse modelo os clientes utilizam um protocolo de comunicação específico para acessar um recurso específico e grande parte do processamento envolvido no serviço ocorre no servidor. Esse modelo tem a grande desvantagem de possuir um ponto central de falhas, além do fato de que com o

crescimento do número de clientes o servidor pode ficar sobrecarregado (Vilanova, 2006).

O cliente em um modelo cliente/servidor tem um P2Pel passivo, ou seja, pode efetuar pedidos e serviços mas não pode disponibilizar serviços a outros clientes. Uma outra abordagem para serviços distribuídos é o modelo *Peer-to-peer* (P2P), o qual dá a máquinas individuais a capacidade de fornecer serviços umas às outras (Vilanova, 2006). Ao contrário de uma rede cliente/servidor, redes P2P podem não depender de servidores centrais, disponibilizando uma rede plana e interconectada, como apresentado na Figura 3.2.

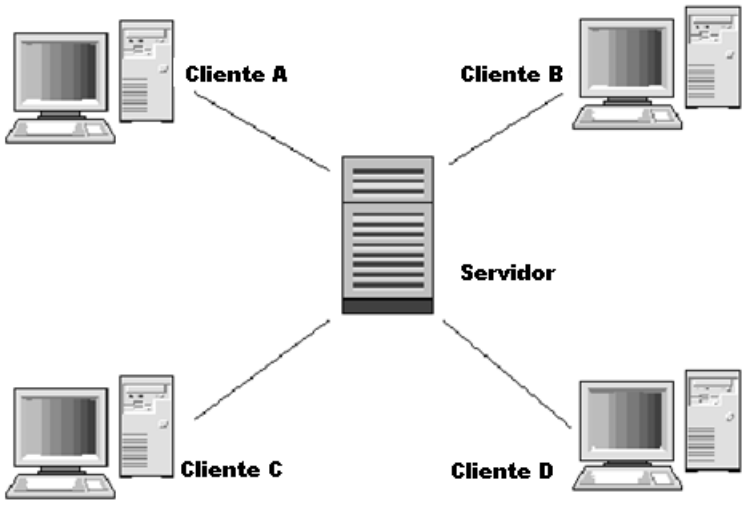


Figura 3.1 Modelo Cliente/Sevidor. Fonte: Vilanova, 2006.

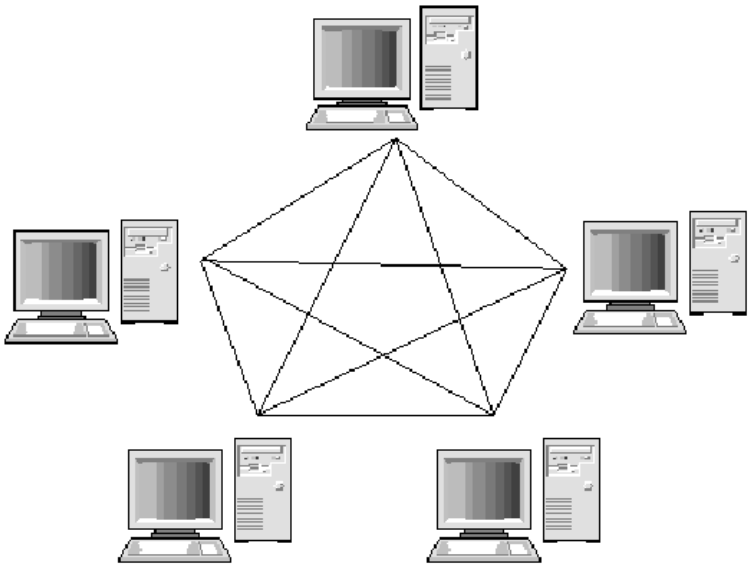


Figura 3.2. Modelo P2P. Fonte: Vilanova, 2006.

Em redes cliente-servidor uma entidade provê o serviço e a outra consome o serviço. Isto gera uma diferença básica da arquitetura cliente-servidor tradicional porque em P2P, a entidade pode funcionar tanto como cliente (fazendo requisições de serviços) como servidor (servindo serviços a outras entidades), enquanto na arquitetura tradicional cliente-servidor, a entidade só pode funcionar como uma de cada vez. Para Rocha (2006) tal explicação elimina dúvidas e confusões conceituais envolvendo redes P2P e arquitetura cliente-servidor.

Os sistemas e as aplicações P2P são distribuídos sem qualquer forma de controle centralizado ou hierarquia organizacional, de tal

forma que o software que está sendo executado em cada nó é equivalente em funcionalidade (Stoica et al., 2001).

Uma classificação para as redes P2P foi proposta por Ge et al. (2003). Nessa classificação as redes P2P foram dividas em três grupos distintos denominados de CIA (*Centralized Indexing Architecture*) ou "Arquitetura de Indexação Centralizada", DIFA (*Distributed Indexing with Flooding Architecture*) ou "Arquitetura de Indexação Distribuída com Inundação" e DIHA (*Distributed Indexing with Hashing Architecture*) ou "Arquitetura de Indexação Distribuída com *Hash*", conforme a Tabela 3.1.

Outra classificação de redes P2P foi realizada por Schollmeier (2006), onde o autor classificou as redes em puras e híbridas. Nessa classificação, com relação as redes puras, os nós são responsáveis por todas as transações entre si (roteamento, autenticação, controle sobre as seções, manutenção de bases de dados), além de gerenciarem todas as informações que sejam relevantes para a aplicação que se utilize dessa rede. Já nas redes híbridas existem servidores centrais responsáveis pela execução de tarefas consideradas como críticas (indexação de informação, aspectos voltados para a segurança - integridade dos dados, segurança no processo de transferência, dentre outras funções).

Apesar de ter apenas poucos anos de uso, o compartilhamento de recursos através de sistemas P2P representa hoje uma considerável fração de tráfego na Internet, em algumas situações até mesmo acima do tráfego Web (Panisson, 2007).

Tabela 3.1 – Classificação de sistemas P2P segundo Ge et al. (2003).

Classificação	Descrição
CIA	Contêm um servidor central ou um cluster de servidores que é responsável por responder os pedidos de busca e realizar todas as tarefas de manutenção da infra-estrutura. O principal exemplo e precursor desta arquitetura foi o *NAPSTER*.
DIFA	Caracterizada pela completa descentralização de seu funcionamento. Os mecanismos de busca e manutenção da infra-estrutura estão distribuídos pela rede, onde cada nó é responsável por manter a listagem dos seus próprios arquivos, e responder quando receber uma busca para um arquivo. Como exemplo, temos o sistema *Gnutella*.
DIHA	Arquitetura, que conforme a DIFA, também possui uma característica totalmente descentralizada. A principal diferença entre as redes DIFA e DIHA está no mecanismo de busca. Na DIHA, cada nó é responsável por um subconjunto do espaço total de índices, onde o nó que entra na rede recebe um espaço do conjunto dos índices dos arquivos. Ao sair da rede, esta deverá designar estes índices para outro nó. As buscas são direcionadas para o nó correto que é o responsável pelo respectivo índice dentro do espaço de índices. Um exemplo deste tipo de arquitetura temos o *Chord*.

Tal demanda pela utilização do sistema P2P tem apresentado certos problemas, como o "*crash*" da rede mundial. Prestadores de serviços de Internet (conhecidos também por *ISPs – Internet Service Provider*) apontaram o grande uso de banda devido ao aumento do tráfego de compartilhamento de arquivos P2P, prejudicando assim a navegação na Web. Comparado a navegação na Web, e-mail ou muitos outros usos da internet onde os dados são transferidos somente em intervalos curtos e em relativamente pequenas quantidades, o compartilhamento de arquivos P2P muitas vezes consiste em uso de banda relativamente pesado devido

à transferência de arquivos em andamento e enxame/coordenação de pacotes da rede (Roettgers, 2009).

3.2 Redes Overlays

Conforme Andersen et al, 2001, uma rede *overlay* (ou rede sobreposta) é uma rede de computadores a qual é construída em cima de outra rede. Os nós na sobreposição podem ser concebidos como sendo conectados por ligações virtuais ou lógicas. Essas ligações dos nós correspondem a caminhos através de ligações físicas na rede subjacente. Por exemplo, muitas redes P2P são redes sobrepostas porque funcionam em cima da Internet. O acesso discado à Internet é uma superposição sobre a rede telefônica. Assim, a rede P2P também configura um tipo de rede *overlay*.

Na Figura 3.3 é apresentado um esquema exemplificando uma rede do tipo overlay, mostrando várias redes sobrepostas, inclusive redes *wireless.*

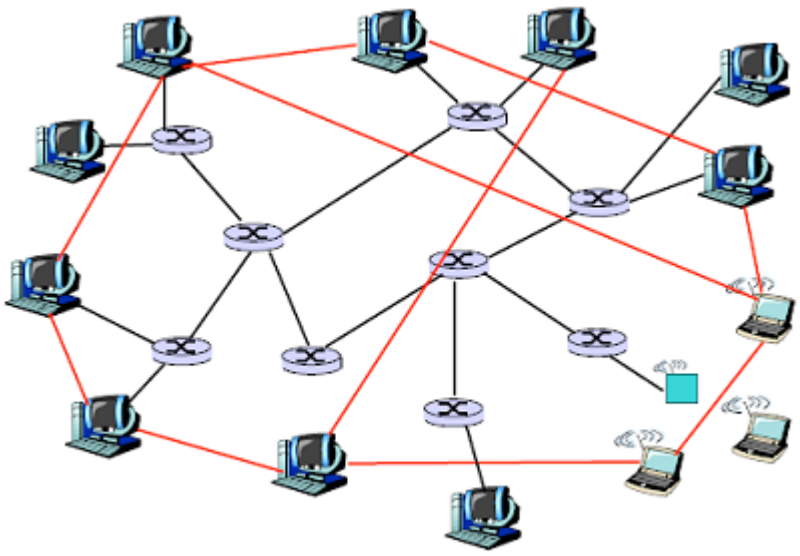


Figura 3.3 – Modelo de rede *Overlay*. **Fonte:** Cardoso, 2009.

3.2.1 Arquiteturas P2P

Redes *Peer-to-peer* normalmente são formadas dinamicamente por adições de nós *ad-hoc*. Em uma rede *ad-hoc*, a remoção de nós não tem impacto significativo na rede. A arquitetura distribuída de uma aplicação em sistema *Peer-to-peer* oferece maior escalabilidade e robustez do serviço.

Sistemas *Peer-to-peer* costumam implementar uma camada de aplicação de rede sobreposta sobre a topologia da rede nativa ou física. Tais superposições são utilizadas para a indexação e descoberta de pares. O conteúdo é tipicamente trocado diretamente sobre a base do Protocolo

de Internet (IP) da rede. Sistemas *Peer-to-peer* anônimos são uma exceção assim como implementar camadas extras de roteamento para encobrir a identidade da origem ou destino de consultas.

Uma das mais utilizadas classificações para as redes *overlay* na comunidade acadêmica é baseada nos mecanismos de consulta e topologia da rede, conforme Rocha et al (2004), que as divide em três categorias:

• Centralizadas. A rede possui um nó central (possivelmente com algumas réplicas para melhorar a confiabilidade e o desempenho) que mantém informações sobre todos os nós e recursos da rede. Todas as consultas são feitas diretamente a esse nó central. Embora mais conhecida devido ao *Napster*, também é muito utilizada nos sistemas de mensagens instantâneas.

• Descentralizadas e não estruturadas. Nessas redes não existe um controle muito rígido sobre a topologia da rede e as consultas são propagadas de nó em nó até que encontrem o destino ou que algum mecanismo de *timeout* encerre o processo. Redes como o *Gnutella* e *Kazaa* utilizam essa arquitetura.

• Descentralizadas e estruturadas. Possuem uma topologia bem definida e utilizam regras para distribuir os dados na rede de modo que facilite a posterior localização dos mesmos. Em geral, as redes baseadas em DHT (*Tabela Hash Distribuída*) se enquadram nessa categoria, como é o caso da Chord, CAN e Pastry.

3.2.2 Exemplos de Redes *Peer-to-peer*

a) Gnutella

O *Gnutella* é um protocolo para compartilhamento de arquivos na Internet que permite a busca de arquivos através de seus nomes, ou partes dele, e a posterior obtenção do arquivo diretamente da máquina de outros usuários. No *Gnutella* não existe um diretório centralizado, como no *Napster*, e as buscas são feitas de forma distribuída. Além disso, também não existe um controle sobre a topologia da rede nem dos locais onde os dados são armazenados (Pinheiro, 2006).

É um sistema de compartilhamento de arquivos de topologia *ad hoc*. Todos os nodos são funcionalmente idênticos, ditos *servents*, porque são *servers* (servidores) e *clients* (clientes) ao mesmo tempo. Buscas por arquivos são realizadas através de uma inundação de escopo limitado (chamado "horizonte"). Nodos em que há um casamento entre o nome do arquivo especificado e o conjunto de arquivos publicados pelo nodo enviam uma resposta positiva, pelo caminho reverso no overlay. O nodo requisitor então escolhe um dos nodos que retornaram resposta e faz o download diretamente deste nodo. Não há garantia que um arquivo será localizado,mas o desempenho de buscas é bom para conteúdo popular. A Figura 3.4 mostra um exemplo de arquitetura *Gnutell*a, onde um nodo faz uma inundação para localizar um arquivo, e uma vez encontrado, faz o

download diretamente de um dos nodos que responderam (positivamente). Melhorias foram realizadas em versões mais recentes, através de uma hierarquia de dois níveis, com supernodos responsáveis por indexar informações de outros nodos, tal como ilustrado na Figura 3.4. Além disso, o esquema de buscas foi modificado de forma a diminuir o grau de inundação da rede (Barcelos & Gaspary, 2006).

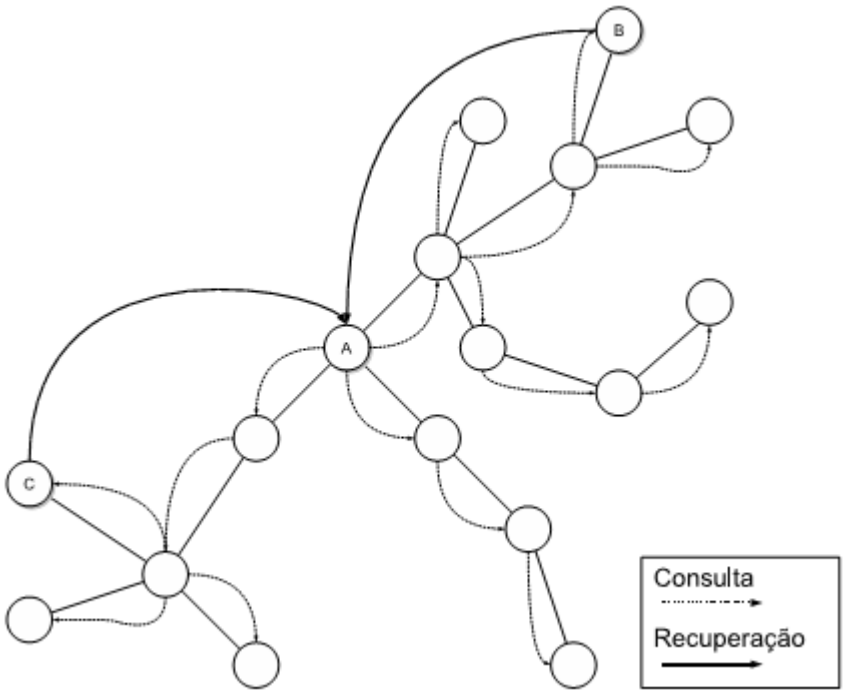


Figura 3.4. *Gnutella:* nodo A faz uma busca por inundação, encontra recurso procurado em Be C, e após interage com os mesmos para obter tal recurso. **Fonte:** Barcelos & Gaspari, 2006).

b) *Chord*

Em redes *Peer-to-peer* estruturadas, as conexões na sobreposição são fixas. Elas costumam usar indexação *Tabela Hash Distribuída* (DHT), como no sistema *Chord* (MIT).

O *Chord* é normalmente utilizado para armazenar pares contendo uma chave e seu valor associado em nós distribuídos pela rede. Posteriormente, o serviço de busca da rede permite que, dada uma chave, seja determinado o nó responsável pela mesma (Stoica et al, 2001).

O espaço de identificadores é formado por um anel conectado, onde cada identificador possui m bits (tipicamente 160 bits). Tanto os nós quanto os dados a serem armazenados são mapeados através de uma função *hash* consistente (Karger et al, 1997, citado por Pinheiro, 2006) para pontos desse espaço de identificadores. O mapeamento dos nós é feito aplicando-se a função *hash* ao seu endereço IP. Cada chave k é armazenada no nó cujo identificador é igual, ou segue *k*, no espaço de identificadores. Para permitir a localização das chaves seria necessário apenas que cada nó mantivesse um apontador para o nó sucessor e o predecessor no anel. Porém, como essa estratégia implicaria em buscas muito ineficientes, cada nó mantém uma tabela de roteamento para outros O (log N) nós, onde N é o número de nós da rede. Supondo que um determinado nó tem como identificador o valor i, a sua tabela conterá apontadores para os nós responsáveis pelos identificadores $i+2^0$, $i+2^1$, $i+2^2$, ... $i+2^{\log N}$. Portanto, essa tabela permite que o roteamento das mensagens seja feito de forma semelhante a uma busca em uma árvore binária, onde a

cada passo o espaço de pesquisa é reduzido a metade. Com essa estratégia o número de nós consultados em uma busca é O(log N) (Pinheiro, 2006).

A Figura 3.5 mostra uma rede *Chord* onde os identificadores possuem um tamanho de 5 bits, de modo que o espaço de identificadores será formado por 32 valores diferentes. Nessa rede existem quatro nós, representados pelos círculos sólidos que, portanto, foram mapeados pela função hash para os identificadores 5, 9, 16 e 25. As linhas entre os nós representam os apontadores mantidos por cada um deles e os números junto a elas indicam os identificadores para os quais o apontador se refere. Evidentemente, os nós 16 e 25 também possuem apontadores, mas esses não são mostrados na figura para não sobrecarregá-la e dificultar a compreensão (Pinheiro, 2006).

c) Kademlia

Infraestrutura de roteamento que usa um mecanismo inovador para roteamento de mensagens e busca de objetos segundo uma métrica de distância entre identificadores de nodos (não de proximidade de rede) baseada em *xor*. A topologia tem a propriedade que toda a mensagem trocada carrega ou reforça informações úteis de contato. O sistema explora essa informação para enviar mensagens de busca assíncronas e paralelas que toleram falhas de nodos sem impor atrasos e timeouts a usuários. Diversas aplicações de P2P estão empregando o algoritmo *Kademlia*: *Overnet*, *eDonkey* e *eMule*, além de *BitTorrent*, que emprega *Kademlia* para permitir o uso de *torrents* sem um *tracker* (Barcelos & Gaspary, 2006).

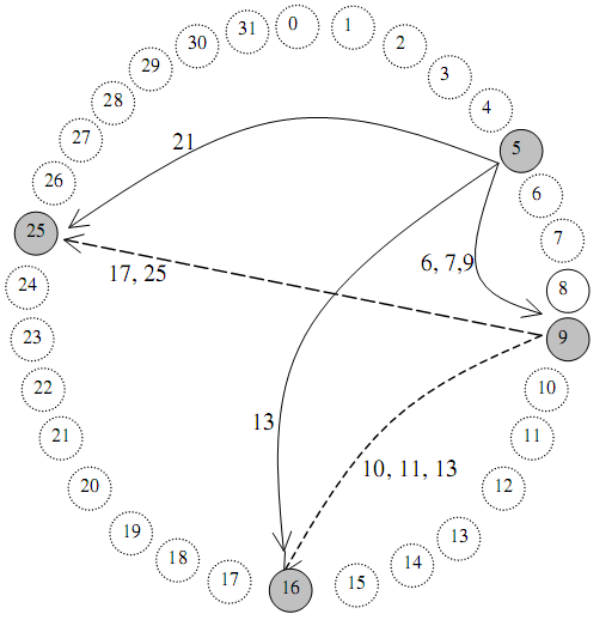


Figura 3.5. Rede *Chord* com 4 nós e 32 identificadores. **Fonte:** Pinheiro, 2006.

3.2 Categorias De Sistemas P2P

Conforme Vilanova (2006), os sistemas P2P podem ser divididos nas seguintes categorias: protocolos, plataformas e aplicações, descritas a seguir na Figura 3.6:

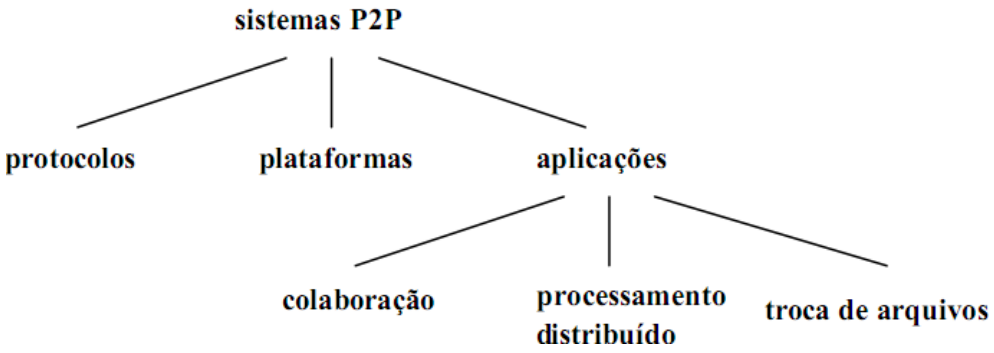


Figura 3.6 - Categorias de Sistemas P2P. Fonte: Vilanova, 2006.

3.2.1 Protocolos

Na categoria dos protocolos estão relacionadas tecnologias que definem, ou tentam definir, padrões para comunicação em redes P2P. Esses padrões definidos pelos protocolos são utilizados como base para o desenvolvimento das aplicações ou plataformas P2P (Vilanova, 2006).

Um exemplo de protocolo é *Gnutella* (Ripeanu et al, 2002), provavelmente o mais conhecido, o qual é utilizado para pesquisa e compartilhamento distribuídos de informação. Este é um tipo de rede *open-source*, surgida no final de 2000 utilizada inicialmente por usuários do sistema *Linux*. Esse protocolo define de que modo os pares se comunicam através da rede. Possui uma estrutura altamente descentralizada não havendo mesmo nenhum servidor central sequer. Os usuários constituem a estrutura da própria rede. Entre os programas que utilizam esse protocolo, estão o *BearShare, LimeWire, Azureus* e agora o *Shareaza*.

3.2.2 Plataformas

As plataformas são sistemas que oferecem os componentes P2P básicos, como descoberta, comunicação, segurança e agregação de recursos. Servem como base para o desenvolvimento e utilização das aplicações. As plataformas permitem que desenvolvedores sem conhecimentos avançados dos padrões de comunicação em redes P2P possam implementar aplicações dessa natureza (VILANOVA, 2006).

Após a criação dos sistemas P2P houve o problema da comunicação entre os diferentes sistemas existentes (estes muitas vezes até prestavam o mesmo tipo de serviço). Dessa forma, houve a necessidade de criação de plataformas para o desenvolvimento de redes P2P que permitissem a comunicação entre si. Por exemplo, arquivos compartilhados em sistemas como o *Kazaa, eMule* e *Gnutella* ficam acessíveis exclusivamente dentro de suas próprias redes, levando usuários a manterem instalados em suas máquinas clientes para cada um dos sistemas de compartilhamento de arquivos que pretenda usar (COULOURIS et al, 2005).

Dentre as principais plataformas criadas que buscam permitir esta comunicação estão o *JXTA*(do inglês *juxtapose*), o *Windows Peer-to-Peer Networking* (redes P2P do Windows), o *XNap* e o *Bluetooth* (Haartsen et al, 1998).

O *Windows Peer-to-Peer Networking* é um conjunto de aplicações, baseadas em *Web Sevices*, para estabelecer conexão entre pessoas, sistemas e dispositivos através da Internet. Fornece serviços de

armazenamento de arquivos, gerenciamento das preferências dos usuários, calendário, entre outros. O *JXTA* é um conjunto de protocolos P2P baseados em mensagens XML para o desenvolvimento de aplicativos distribuídos, e possui implementações em *Java* e em *C* (Vilanova, 2006).

3.2.3 Aplicações

As aplicações são os sistemas com funcionalidades específicas e podem ser, por sua vez, subdivididas em outras três categorias: processamento distribuído, colaboração e troca de arquivos. As aplicações de processamento distribuído utilizam o poder computacional disponível dos seus usuários para formar supercomputadores. Esses sistemas dividem grandes tarefas em pequenos pedaços, e distribuem esses pedaços para serem processados pelos dispositivos da rede (Androutsellis-Theotokis & Spinellis (2004) citado por (Vilanova, 2006). Exemplos de aplicações bastante conhecidas estão o *ICQ*, o MSN e o *BearShare*.

3.3 Fundamentos de Segurança da Informação

Todo projeto de Segurança de Informações procura abranger, pelo menos, os processos mais críticos do negócio em questão (Moreira, 2001).

Dessa forma, conforme Detsch (2005), quando se trata de segurança em redes P2P, pode-se identificar dois grandes campos de pesquisa. Um deles se refere a garantir a segurança de uma rede/instituição quando do uso de aplicações P2P quaisquer por parte de seus integrantes. Fazem parte desse escopo medidas como bloquear o tráfego gerado por aplicações de compartilhamento de arquivos (para que a banda da rede não fique comprometida) ou tentar evitar que tais aplicações sirvam de porta de entrada para vírus ou *trojan horses.* O segundo campo de pesquisa tem por objetivo possibilitar a criação de aplicações P2P seguras no que diz respeito ao seu funcionamento interno. Isto implica possibilitar ao programador de aplicações P2P a adição de diferentes aspectos de segurança, como autenticação, confidencialidade, integridade e autorização (Detsch, 2005).

A seguir serão tratados os princípios da robustez e sua importância em redes *Peer-to-peer*, para então, aprofundar no conteúdo de segurança da informação.

3.3.1 Camada de Robustez

O princípio de robustez é uma diretriz geral para o desenvolvimento de software que opera ou controla a infra-estrutura da Internet ou outras redes baseadas em Protocolos da Internet.

O *Internet Engineering Task Force* publica suas pesquisas, comunicações públicas, políticas e especifica padrões de configuração

como uma série numerada de documentos chamados de RFC (*Request for Comments*). No RFC 761 (Transmission *Control Protocol*, 1980) o cientista da computação americano *Jon Postel* resumiu comunicações anteriores sobre critérios de interoperabilidade desejado para o Protocolo de Internet (cf. IEN 111 , RFC 760) da seguinte forma: “Implementações de TCP devem seguir um princípio geral de robustez: ser conservador no que você faz, ser liberal no que você aceita dos outros" (IEN 111, 1979).

A segurança é um dos maiores desafios dessa tecnologia, já que os dispositivos podem atuar como clientes e servidores, ao mesmo tempo, coloca o sistema em posição de risco. Cada participante em um sistema P2P deve proteger seus recursos e serviços da intrusão, tanto dos outros participantes da mesma rede quanto de acessos externos não autorizados (Peermetrics, 2003).

Essa necessidade de segurança exige um constante controle da parte de cada usuário, ou a interação desse usuário com uma terceira parte, capaz de validar a identidade dos usuários. Mas centralizar os mecanismos de segurança é uma solução que anula os benefícios de uma estrutura descentralizada. Outro aspecto relativo à segurança é a utilização de sistemas de criptografia para a transmissão dos dados (Vilanova, 2006).

3.3.2 Principais Aspectos de Segurança da Informação

A segurança em redes de computadores abrange diversos aspectos, que atendem a diferentes objetivos do ponto de vista dos usuários das aplicações (Detsch, 2005). Nesta seção serão abordados os principais aspectos de segurança: Disponibilidade Confidencialidade, Autenticação, Integridade, Não-repúdio, Autorização, Auditoria, Anonimicidade, Confiança e Reputação. Para cada um deles serão apresentados conceitos fundamentais e os desafios/implicações de seu uso em sistemas P2P.

a) Disponibilidade

Este requisito é medido em função da parcela de tempo que um determinado objeto ou serviço está disponível para acesso. No compartilhamento de arquivos se refere ao sucesso das operações de leitura e escrita de dados. Em sistemas P2P para a computação distribuída é a garantia de que os pares estão acessíveis para realizar o processamento esperado (Silva, 2007).

Os esforços da instituição em proporcionar a disponibilidade dos seus recursos, sejam eles sistemas, informações ou processos, ocorrem quando estes necessitam de acesso contínuo e ininterrupto, ou seja, a informação deve estar disponível para a pessoa certa e no momento em que ela precisar (Moreira, 2001).

Segunda Silva (2007), a principal contramedida para ataques de disponibilidade refere-se à replicação. Os principais ataques à disponibilidade estão baseados em negação de serviço DoS – *Denial of Service* ou "negação de serviço" e ataques de roteamento.

A negação de serviço pode ser realizada no nível de rede ou no nível da rede de sobreposição. Já ataques de roteamento referem-se a anomalias no encaminhamento das mensagens ou o seu simples descarte, ambos causados por pares maliciosos para desviar as mensagens do seu destino final. Normalmente ataques de roteamento são dependentes do tipo de rede P2P: estruturada e não estruturada (descrita no item 3.2.1).

Diversos tipos de ataques são descritos por Sit & Morris (2002), citado por Barcellos & Gaspary (2006). Segue abaixo quatro tipos associados a negação de serviços.

O primeiro consiste em um nó malicioso funcionar corretamente para buscas, mas quando solicitado, negar o serviço, ou seja, a existência de um objeto sob sua responsabilidade (ou se recusar a enviar uma resposta). Este tipo de ataque, que é trivialmente detectável por nós corretos, pode ocorrer tanto em overlays estruturados como não estruturados. Como defesa, um *overlay* P2P pode implementar replicação (na camada de armazenamento). Em geral, deve-se evitar pontos únicos de responsabilidade, e replicação pode permitir que não exista um único nó responsável pela replicação ou acesso às réplicas.

O segundo tipo de ataque descrito é a sobrecarga de nós específicos, através de um ataque DoS convencional. Neste caso, um nó

correto sob ataque ficaria incomunicável, fazendo com que o mesmo fosse eliminado do overlay por outros nós. Este ataque deve ser combatido através da alocação aleatória de identificadores, uso de réplicas de objetos e serviços, e sua dispersão física na rede.

O terceiro tipo descrito é o ataque distribuído de negação de serviço (também conhecido como DDoS, um acrônimo em inglês para *Distributed Denial of Service*). Um computador mestre (denominado "Master") pode ter sob seu comando até milhares de computadores ("Zombies" - zumbis). Repare que nestes casos, as tarefas de ataque de negação de serviço são distribuídas a um "exército" de máquinas escravizadas (Wikipédia, 2009). O ataque consiste em fazer com que os Zumbis (máquinas infectadas e sob comando do Mestre) se preparem para acessar um determinado recurso em um determinado servidor em uma mesma hora de uma mesma data. Passada essa fase, na determinada hora, todos os zumbis (ligados e conectados à rede) acessarão ao mesmo recurso do mesmo servidor. Como servidores *web* possuem um número limitado de usuários que pode atender simultaneamente ("slots"), o grande e repentino número de requisições de acesso esgota esse número de *slot*, fazendo com que o servidor não seja capaz de atender a mais nenhum pedido. Dependendo do recurso atacado, o servidor pode chegar a reiniciar ou até mesmo ficar travado. Vírus conhecidos criados para a distribuição de rotinas de ataque de negação de serviço incluem "*Codered*", "*Slammer*", "*MyDoom*" e "*MyPenis*", "*MyBalls*" , que escravizam o infectado. Segue na Figura 3.7 um diagrama exemplificando um ataque.

O quarto tipo de ataque é o de entrada e saída acelerada de nós (*churn* excessivo ou "mudanças bruscas de posição"). Em sistemas de alta disponibilidade, que buscam manter seus objetos sempre disponíveis apesar da ausência de certos nós, é necessário que objetos sejam copiados, ou do nó que deixa o *overlay* em caso de saída proposital, ou de réplicas em caso de falha. Como estas operações possuem custo associado, um ataque que provoca a entrada e saída rápida de nós tem o potencial de sobrecarregar determinados nós ou um segmento da rede e causar uma negação de serviço aos demais nós do *overlay* P2P. Entretanto, se os nós maliciosos precisam se envolver nestas operações, então seus próprios recursos seriam exauridos, o que reduziria bastante o poder de um atacante.

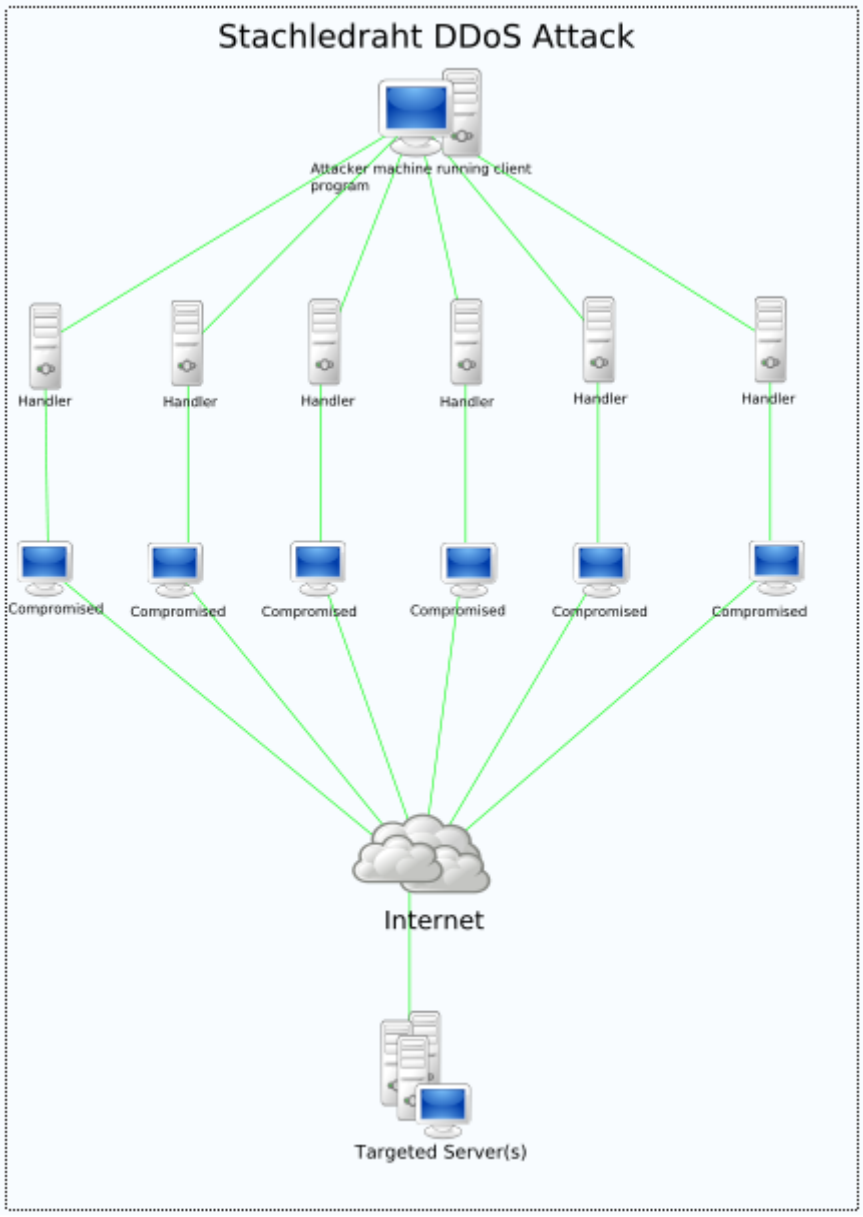


Figura 3.7. Diagrama de um Ataque *DDoS Stachledraht*. **Fonte:** Wikipédia, 2009.

Em Sit & Morris (2002), citado por Barcellos & Gaspary (2006), descreve-se ainda o ataque de mensagens não solicitadas, onde um nó malicioso engenha uma situação em que envia mensagens de resposta não solicitadas, interferindo por exemplo em buscas. A melhor defesa para este tipo de ataque seria empregar técnicas padrão de autenticação, tal como assinaturas digitais ou códigos de autenticação de mensagens (MACs - *Message Authentication Code*). Como assinaturas digitais são dispendiosas computacionalmente e MACs requerem chaves compartilhadas, *nonces*[1] podem ser adicionados a mensagens como forma de defesa, exigindo-se que o mesmo valor fornecido na mensagem de requisição seja incluído na mensagem de resposta.

Ataques de roteamento são anomalias de roteamento em que mensagens transmitidas através do *overlay* são desviadas para longe de seu destino, para nós maliciosos, ou descartadas. Este tipo de ataque aumenta a chance de falhas na busca, e possui impacto negativo no custo e desempenho do sistema. Para que este tipo de ataque ocorra, as informações de roteamento ou sobre outros nós em nós corretos são "envenenadas" por nós maliciosos. Isto ocorre, por exemplo, quando nós maliciosos respondem com rotas incorretas a mensagens de busca (Barcellos & Gasparay, 2006).

[1] derivado de *number used once*, indica número usualmente aleatório empregado para evitar que uma mensagem prévia possa ser usada em um ataque.

b) Confidencialidade

A confidencialidade é a propriedade que visa manter o sigilo, o segredo ou a privacidade das informações, evitando que pessoas, entidades ou programas não-autorizados tenham acesso às mesmas (Moreira, 2001). A confidencialidade garante que somente as partes envolvidas na comunicação serão capazes de processar as mensagens enviadas (Silva, 2007).

Dessa forma, confidencialidade é o aspecto que garante a proteção dos dados transmitidos quanto a sua monitoração por parte de entidades não autorizadas. A confidencialidade pode ser aplicada em vários níveis, desde toda uma transmissão de dados entre dois usuários até individualmente por mensagem ou mesmo protegendo apenas campos específicos de uma mensagem. Adicionalmente, confidencialidade pode também estar relacionada a evitar a análise do tráfego por terceiros através da identificação de origem/destino, freqüência, duração ou outras características do fluxo de mensagens.

A garantia de confidencialidade dos dados está relacionada ao uso de um algoritmo de cifragem, ou criptografia (Detsch, 2005). Algoritmos com esta finalidade podem ser divididos em dois grupos: criptografia simétrica e criptografia assimétrica.

Em algoritmos de criptografia simétrica, ambas as partes comunicantes compartilham uma única chave secreta que é usada tanto na cifragem (no remetente) quanto na decodificação (no destino). Os algoritmos mais difundidos nesta categoria são o *Data Encryption*

Standard (DES), o *Triple Data Encryption Algorithm* (TDEA), também referenciado como *triple*-DES, e o *Rivest Cipher 4* (RC4).

Exemplificando o funcionamento, o DES divide a mensagem original em blocos de 64 bits. A execução de uma série de operações sobre cada bloco utilizando como base uma chave de 56 bits resulta em um bloco cifrado, também com 64 bits.

No caso da criptografia assimétrica, o processo de comunicação segura envolve o uso de duas chaves distintas: uma para cifragem e outra para decodificação. A chave para cifragem pode ser distribuída livremente, sendo por isso chamada também de chave pública. Um dos algoritmos de criptografia assimétrica mais difundidos é o RSA (Rivest et al, 1978). A exemplo do DES e do TDEA, seu funcionamento é orientado a blocos de dados, com a diferença de que é permitido o uso de um tamanho arbitrário de bloco. O algoritmo possibilita também o uso de chaves de qualquer tamanho, sendo 768, 1024 e 2048 bits tamanhos típicos (Detsh, 2005).

b.1) Troca de chaves

Uma das principais etapas da comunicação utilizando cifragem está na troca de chaves entre as partes. Para o caso de criptografia simétrica, a princípio é necessário o uso de um canal seguro para transmissão da chave, ou seja, o emprego de alguma conexão criptografada com outra chave (usando uma terceira entidade como

intermediária, por exemplo). Pode-se também realizar a troca de chaves por um meio físico, sem depender da rede.

Já na criptografia assimétrica, a troca de chaves pode ser feita de maneira mais simples, posto que a chave pública de uma entidade pode ser transmitida livremente, sem a necessidade de um canal seguro entre as estações. Entretanto, a simples transmissão de uma chave pública entre duas entidades A e B não garante a autenticidade das partes: uma terceira entidade C pode se fazer passar por A enviando uma chave pública falsa. Para evitar esse tipo de falsificação, podem ser usados certificados digitais, que utilizam a assinatura de uma entidade confiável (*Certification Authority*) para garantir a autenticidade nas transmissões (Detsch, 2005).

A criptografia de chave pública ou criptografia assimétrica é um método de criptografia que utiliza um par de chaves: uma chave pública e uma chave privada. A chave pública é distribuída livremente para todos os correspondentes via e-mail ou outras formas, enquanto a chave privada deve ser conhecida apenas pelo seu dono (Wikipédia, 2009).

Num algoritmo de criptografia assimétrica, uma mensagem cifrada com a chave pública pode somente ser decifrada pela sua chave privada correspondente.

Os algoritmos de chave pública podem ser utilizados para autenticidade e confidencialidade. Para confidencialidade, a chave pública é usada para cifrar mensagens, com isso apenas o dono da chave privada pode decifrá-la. Para autenticidade, a chave privada é usada para cifrar

mensagens, com isso garante-se que apenas o dono da chave privada poderia ter cifrado a mensagem que foi decifrada com a 'chave pública' (Wikipédia, 2009).

c) Autenticação

A autenticação assegura que alguém (ou algo) é de fato quem (ou o que) afirma ser (Silva, 2007). Em uma transação ou em uma conexão mantida entre duas estações, dois aspectos estão envolvidos. Primeiro, no estabelecimento da conexão, o serviço assegura que ambas as entidades são autênticas. Segundo, o serviço assegura que a conexão não sofre interferência de forma que uma terceira entidade possa se fazer passar por uma das partes com o propósito de envio ou recebimento não autorizado de mensagens (Detshc, 2005).

c.1) Autenticação Com Criptografia

É possível prover autenticação simplesmente usando criptografia simétrica. Assumindo que apenas o remetente e o receptor compartilham uma chave, então apenas o transmissor genuíno é capaz de cifrar corretamente a mensagem para o outro participante. De forma similar, o transmissor da mensagem pode usar um algoritmo de criptografia assimétrica, mas, ao invés de realizar a etapa de cifragem com a chave pública (do receptor), ele utiliza a sua própria chave privada. O receptor, ao conseguir decodificar a mensagem utilizando a chave pública referente ao transmissor, saberá que o remetente realmente é quem afirma ser, já

que apenas o transmissor conhece a chave privada correspondente à chave pública utilizada na decodificação. A utilização desta técnica isoladamente não garante a confidencialidade dos dados transmitidos, uma vez que a chave relativa à decodificação da mensagem é pública (Detsch, 2005).

c.2) Autenticação Sem Criptografia

Apesar de ser possível encriptar e prover autenticação em uma só etapa, muitas vezes é interessante permitir a autenticação sem a necessidade de cifrar a mensagem em si, utilizando apenas uma *tag* de autenticação. Isto permite que a mensagem possa ser lida normalmente, sem necessidade de passar por um processo prévio de decodificação. A verificação da assinatura pode ser feita apenas quando necessário (ou através de amostragem sobre as mensagens recebidas), reduzindo o custo total de processamento (Detsch, 2005).

A técnica código de autenticação de mensagem (*Message Authentication Code* -MAC) envolve a utilização de uma chave secreta, compartilhada apenas entre as duas partes comunicantes, para geração de um pequeno bloco de dados, o código de autenticação. A Figura 3.8 ilustra o funcionamento do mecanismo. Quando uma mensagem é transmitida, realiza-se também o envio do código de autenticação, que é calculado em função da mensagem e da chave secreta. O receptor calcula para a mensagem recebida (usando a chave secreta) o código de autenticação resultante. Se este código coincidir com o código recebido, a

mensagem é autêntica. O algoritmo para obtenção do código pode ser o próprio DES: gera-se a versão cifrada da mensagem e utiliza-se os últimos bits resultantes (tipicamente 16 ou 32 bits) como código de autenticação (Detsch, 2005).

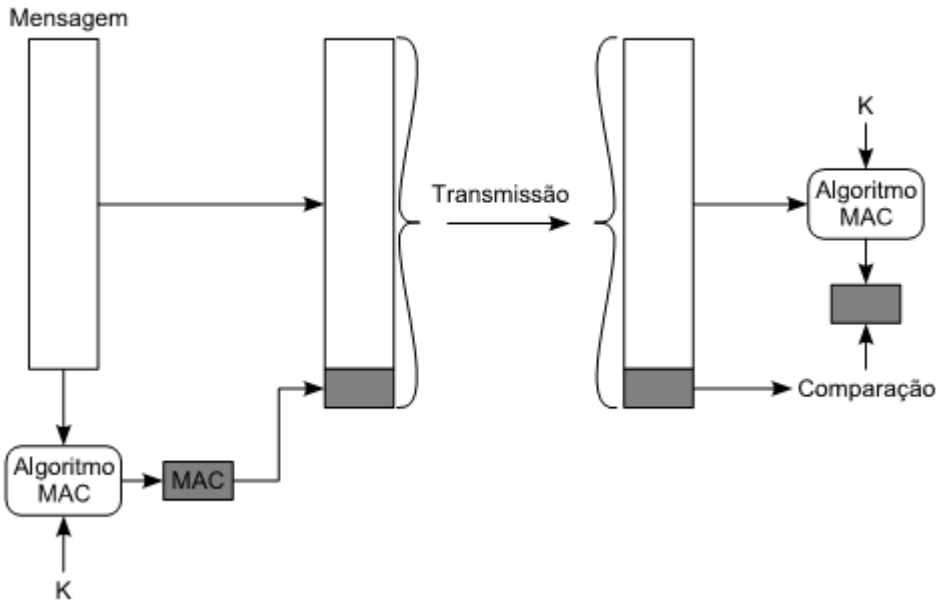


Figura 3.8. Autenticação de Mensagem utilizando código de autenticação (MAC). **Fonte:** (Detsch, 2005).

Uma série de ataques é possível nesse contexto, dentre eles destaca-se *Sybil*, tratado por Douceur (2002). O ataque *Sybil* consiste na falsificação de múltiplas identidades em um sistema. Segundo o autor, em um sistema P2P de larga escala não é possível evitar, pragmaticamente, que um nó malicioso gere múltiplas identidades, de forma que a única forma realmente segura de se gerenciar identidades de

nós é através de uma autoridade certificadora de confiança, podendo ser esta implícita ou explícita.

Conforme Barcellos & Gaspary (2006), o problema da autenticação em P2P pode ser resolvido através de uma Agência Certificadora ou outro elemento centralizado em que se confie na atribuição de identidades. No entanto, tal esquema não é desejável em um sistema distribuído, particularmente em um overlay P2P com potencial para lidar com milhões de nós, por ser um ponto central de falha e um potencial "gargalo".

d) Integridade

Consiste em proteger a informação contra qualquer tipo de alteração, sem a autorização explícita do autor da mesma (Moreira, 2001).

Assim como confidencialidade, a integridade pode ser aplicada a um fluxo de mensagens, a uma mensagem individual ou a alguns campos da mensagem. Um serviço de integridade orientado à conexão (aplicado a um fluxo de mensagens) garante que as mensagens são recebidas da mesma forma como enviadas, sem duplicação, inserção, modificação ou reordenamento. Já um serviço de integridade baseado em mensagens individuais provê apenas garantias quanto à modificação de mensagens. Serviços de integridade podem ser diferenciados também pela existência ou não de recuperação (*recovery*). Quando uma violação de integridade é detectada, o serviço pode simplesmente reportar esta violação ou prover

mecanismos para automatizar a recuperação dos dados alterados, sem necessidade de intervenção externa (Detsch, 2005).

A utilização de criptografia para a proteção de dados prevê a checagem de integridade dos dados transportados, através da verificação da sua assinatura (*hash*), comparando-a com o resultado do processo antes da transmissão. A perda de integridade pode ser intencional ou não. Independente da forma ou motivo, o maior problema na perda de integridade é o montante que a instituição vai gastar para recuperar ou reconstituir os dados, quando possível (Galvão, 2006).

Visto que a violação dos mecanismos baseados em criptografia é bastante custosa computacionalmente, os ataques à integridade mais típicos são os seguintes (Barcellos & Gaspary, 2006):

• ataque de poluição a arquivos (*file-targeted DoS attack*): um nó malicioso anuncia uma cópia corrompida de um arquivo e, então, a distribui quando solicitado por outro nó. Evidências indicam que isso está sendo realizado pela indústria fonográfica, e existem companhias que vendem esse serviço de ataque publicamente na Internet, como por exemplo *overpeer*;

• ataque de resposta falsa (*false attack reply*): um nó malicioso encaminha normalmente mensagens de busca; entretanto, identifica e intercepta mensagens de resposta (para qualquer arquivo), e modifica a resposta indicando ele mesmo como detentor de uma cópia e com baixíssimo atraso; se selecionado pelo requisitor, o nó malicioso fornece uma cópia corrompida do arquivo.

e) Não-repúdio

O não-repúdio compreende os esforços dispendidos para garantir a autoria de determinadas ações (Moreira, 2001). Dessa forma, o não-repúdio previne que alguma das entidades envolvidas em uma comunicação possa negar alguma operação, como envio ou recebimento de determinada mensagem. Sua principal aplicação está em transações comerciais ou financeiras, onde, eventualmente, uma das partes poderia levar vantagem ao negar a realização de alguma transação (Detsch, 2005).

f) Autorização

Autorização é a capacidade de restrição de acesso aos recursos baseando-se em informações sobre o requisitante, o recurso a ser acessado e os detalhes específicos da requisição (Silva, 2007). Em um contexto de segurança de redes, autorização (ou controle de acesso) é a habilidade de limitar e controlar o acesso aos sistemas e aplicações da estação através dos canais de comunicação. Sistemas de controle de acesso podem descrever não somente quem ou que processo pode ter acesso a um recurso específico, mas também o tipo de acesso que é permitido (Detsch, 2005).

Para Silva (2007), autorização é um requisito importante para a adoção mais ampla de P2P, por exemplo em ambientes corporativos. Mesmo em sistemas de compartilhamento de arquivos, podem ser utilizados métodos para a autorização de downloads conforme as garantias necessárias a determinado ambiente.

A ainda baixa popularidade de mecanismos de controle de acesso em aplicações P2P pode ser explicada, em parte, pelos requisitos a serem satisfeitos por potenciais soluções. O primeiro consiste em definir uma solução que não comprometa a escalabilidade da aplicação; a maioria das propostas para controle de acesso em ambientes mais tradicionais faz uso de serviços centralizados para prover esse tipo de funcionalidade. O segundo requisito consiste em lidar, satisfatoriamente, com o anonimato característico das redes P2P. Ao contrário de sistemas cliente/servidor, em que há um acoplamento forte entre os envolvidos na comunicação, nós em um sistema P2P típico são fracamente acoplados e fornecem pouca informação sobre sua identidade. O terceiro requisito está relacionado a encontrar uma forma de manter o incentivo ao compartihamento, apesar das restrições impostas pelas políticas de acesso associadas ao mecanismo de autorização (Barcellos & Gaspary, 2006).

Nesse sentido, Tran et al. (2005) propõem um *framework* para controle de acesso em aplicações de compartilhamento de arquivos P2P, mesclando aspectos de modelos de reputação e recomendação com esquemas de justiça (*fairness*) e controle de acesso. Nessa solução o nó é considerado como um sistema *standalone* em que os arquivos compartilhados são tratados como objetos que precisam ser protegidos, e os nós que solicitam tais objetos são sujeitos que possuem ou precisam ganhar direitos de acesso. Os arquivos disponíveis são categorizados de acordo com seu tamanho e conteúdo, e têm associado a si dois limiares, que determinam dois aspectos relacionados ao seu acesso. Um nó que solicita a recuperação de um objeto precisa ter valores equivalentes ou

superiores aos limiares associados a esse objeto. Segundo os autores, esses valores são computados com base em quatro escores: *direct trust*, *indirect trust*, *direct contribution e indirect contribution*, cabendo ao próprio nó requisitante coletar, junto a outros nós, recomendações que o habilitem a calcular seus valores perante um determinado nó. Após o término de cada transação, os valores de *direct trust* e *direct contribution* são atualizados de acordo com o grau de satisfação da transação. Esses novos valores afetarão as avaliações de controle de acesso em futuras comunicações entre esses dois nós.

g) Auditoria

A auditoria permite análise do funcionamento dos mecanismos de segurança durante ou após a sua execução. A técnica mais simples e difundida com este objetivo consiste na geração de logs de execução. Através da análise desse log, deve ser possível tanto verificar o correto funcionamento dos mecanismos de segurança, acompanhando a sua execução, quanto possibilitar a identificação de causas e responsáveis por uma falha em algum dos mecanismos. Dependendo da situação, esta análise pode ser feita tanto de forma manual quanto automática, sendo importante haver uma estrutura coerente na geração dos logs (Detsch, 2005).

h) Anonimicidade e Negabilidade

A anonimicidade garante que as identidades reais no sistema permanecerão desconhecidas (Silva, 2007). Ela tem por objetivo evitar que uma estação possa ser identificada em uma comunicação com outra estação ou um servidor qualquer. Na forma mais simples, uma estação envia mensagens para o servidor através de um proxy, como o Anonymizer.com (The Anonymizer, 2009). Desta forma, o servidor não consegue identificar diretamente o remetente original das mensagens. O sistema falha caso o proxy revele a identidade de um usuário ou se um adversário puder observar o tráfego de entrada e saída do Proxy (Detsch, 2005).

Em redes P2P a anonimicidade garante não-identificação: (a) do autor ou do responsável pela publicação de determinado objeto ou serviço; (b) da identidade do par que está armazenando um objeto ou serviço; (c) da identidade e do conteúdo do objeto; e (d) dos detalhes da requisição para recuperar determinado objeto (Dingledine et al., 2001, citado por Silva, 2007).

A privacidade no sistema é obtida usando um esquema semelhante às redes *mix* de Chaum para comunicações anônimas (Chaum, 1981). Ao invés das mensagens serem transportadas diretamente da origem para o destino, elas passam por cadeias de nó a nó em que cada canal é cifrado individualmente, até que a mensagem chegue ao destinatário. Como cada nó na cadeia conhece apenas seus vizinhos imediatos, não há como determinar a identidade dos nós que publicam, dos nós que estão

armazenando os objetos e dos nós de onde partem as requisições para recuperar os objetos (Barcelos & Gasparay, 2006).

Os principais ataques a que soluções de anonimicidade estão vulneráveis são os passivos, em especial os de escuta e análise de tráfego. O objetivo desses ataques é desvendar a identidade dos elementos envolvidos nas comunicações estabelecidas. No caso particular das aplicações P2P, há interesse – ainda – em revelar quem publicou, quem está armazenando e quem solicitou determinados objetos (Barcellos & Gaspary, 2006).

Segundo Theotokis & Spinellis (2004), a negabilidade em aplicações P2P pode ser entendida como um componente do aspecto anonimicidade, referindo-se à habilidade de um nó em negar conhecer o conteúdo de objetos por ele armazenados.

i) Reputação

O estabelecimento de um mecanismo de reputação entre entidades de um sistema tem por objetivo permitir a um usuário determinar um grau de confiança relativo aos demais nós da rede. Com isso, é possível associar um "risco" ao uso ou prestação de um serviço baseado na reputação da outra parte envolvida. Em redes P2P que não possuam requisitos fortes de autenticação, um esquema de reputação pode atuar como substituto a um sistema baseado em certificados digitais: pode-se confiar em um par simplesmente por seu comportamento no sistema, sem necessidade de mapear o par com a entidade física responsável. Esta

característica permite ainda a coexistência de um mecanismo de anonimicidade.

Um dos problemas fundamentais de sistemas de reputação é garantir a validade das informações prestadas por outros nós. Portanto, é natural que na determinação do escore de reputação de um estranho, experiências anteriores do próprio nó sejam valorizadas em relação à opinião de outros nós. Uma abordagem comum nesse sentido é a aplicação de pesos: a reputação de uma informação dada por um nó é proporcional à reputação desse nó. Informações coletadas através de confiança transitiva podem ser pesadas de acordo com a reputação do nó de menor reputação na cadeia de confiança; alternativamente, se os valores de confiança situam-se em [0,1], então o valor resultaria da multiplicação dos escores de reputação de cada um dos nós (Barcellos & Gaspary, 2006).

Existem diferentes formas de ataque a sistemas de reputação, sendo os principais discutidos a seguir.

No ataque de nó *traidor,* conforme Marti & Garcia-Molina (2006), um nó se comporta adequadamente por um tempo de forma a construir uma boa reputação, e então explora o sistema valendo-se da mesma. Este ataque é especialmente efetivo quando os nós ganham privilégios a medida que conquistam reputação. Em termos sistêmicos, um nó traidor pode surgir não de uma mudança comportamental de um usuário, mas de uma mudança no ambiente: por exemplo, uma máquina cliente perfeitamente correta pode ser infectada com um vírus estilo Cavalo de

Tróia, que então poderia aleatoriamente abusar da boa reputação do nó. Segundo Feldman et al. (2004), a resistência a esse tipo de ataque pode ser aumentada usando a análise da história recente de um nó.

Outro tipo de ataque é o de conspiração contra sistemas de reputação. Este tipo de ataque é freqüentemente efetivo porque em sistemas de reputação típicos um nó deve consultar outros nós sobre a reputação de um terceiro. Se muitos nós estão comprometidos, então nós podem prover falso testemunho, no sentido de aumentar a reputação de um nó malicioso, ou de atacar um nó correto diminuindo a sua reputação. Em princípio, o atacante deveria dispor de recursos em massa, fazendo com que boa parte dos nós do overlay fossem seus; entretanto, conforme Cheng & Friedman (2005), em muitos sistemas P2P não existe um esquema seguro de autenticação, possibilitando que nós adquiram múltiplas identidades falsas (criando nós *Sybil*) com um único nó físico.

O *whitewashing* é outro ataque conhecido, ocorrendo apenas quando nós podem trocar sua identidade facilmente (o que é o caso de muitos sistemas P2P). Um nó pode deixar o sistema e voltar em seguida com uma nova identidade, em uma tentativa de se livrar de qualquer reputação ruim que ele tenha acumulado. Se a política dos nós em relação a estranhos é permissiva, nós podem "usar" a reputação inicial, deixar o sistema e reingressar com nova reputação inicial. Se um nó não consegue distinguir um nó novo correto de um antigo, então *whitewhashers* podem causar o colapso do sistema se nenhuma contramedida for tomada (Feldman et al., 2004).

Conforme Silva (2007), verifica-se claramente que alguns requisitos são base para outros (autenticidade e autorização) e, em contrapartida, outros têm funções antagônicas (autenticidade e anonimicidade). Considera-se que cada aplicação P2P possui seus requisitos próprios de segurança e que deve ter métodos de segurança instanciados conforme a sua necessidade.

3.3.3 Confiança em Redes P2P

Um dos desafios fundamentais em redes e sistemas P2P está relacionado com os riscos de interação e colaboração com entidades desconhecidas e que podem ser potencialmente maliciosas (Duma et al., 2005). Segundo os autores os sistemas atuais de reputação não são capazes de reagir adequadamente a mudanças rápidas de comportamento de entidades pares. Como possível solução, um trabalho de métrica de confiança que satisfaça estas necessidades foi proposto (Duma et. al 2005), sendo capaz de identificar e punir mudanças bruscas no comportamento de entidades pares, além de verificar possíveis oscilações do comportamento, indicando a possibilidade de que uma ação maliciosa possa ocorrer.

Para Suryanarayana & Taylor (2004) tanto a reputação quanto a confiança devem ter características que são consideradas importantes no contexto de sistemas e ambientes P2P. Estas características estão sintetizadas na Tabela 3.2.

Tabela 3.2 – Características de reputação e confiança. Fonte: Suryanarayana &Taylor (2004)

Característica	**Justificativa**
Controle local	Inclui valores de confiança e reputação, bem como outras informações e recursos. É usado para distinguir mecanismos de confiança que trocam valores locais dos que não trocam.
Valores de confiança e reputação	Representam a confiança que um par tem em outro, através de valores que podem ser discretos ou contínuos.
Tipo de reputação	Indica o tipo de reputação utilizado por um modelo de confiança. São utilizados três tipos: reputação positiva, reputação negativa ou uma combinação dos dois.
Verificação de assinatura	Usada para distinguir modelos de confiança que usam explicitamente verificação de credencial para estabelecer a autenticidade do emissor da mensagem.
Anonimato	Utilizada para proteger a identidade dos pares, a fim de protegê-los de ações maliciosas. Essa propriedade ignora as relações de confiança entre os pares.
Custo de largura de banda	A troca de mensagens entre os pares resulta em muito tráfego sendo gerado simultaneamente, que acarreta em um aumento na utilização da banda. Reduzir esta utilização é um objetivo muito importante para qualquer modelo de confiança.
Custo de armazenamento	Utilizado em modelos de confiança que necessitam que os pares armazenem informação acerca de outros pares. Este custo aumenta linearmente com o número de pares.
Tolerância à	Representa a habilidade do modelo de confiança de se

falhas	adaptar à natureza transiente do sistema (mudanças de topologia).
Escalabilidade	Remete à habilidade de um modelo de confiança de se adaptar a um aumento do número de pares.
Confiabilidade	Refere-se à habilidade de um modelo de confiança em determinar corretamente a extensão da confiança nos outros pares baseado em experiências passadas e/ou em informações recebidas de outros pares. A confiabilidade também é determinada pela tolerância à falhas do modelo

Segundo Sousa et al. (2006) , um fato importante a ser considerado em um modelo de segurança em redes está relacionado com a certificação digital. O principal problema, de acordo com a análise realizada, é que não se pode ter uma autoridade certificadora centralizada em redes distribuídas, porque todos os aspectos de gerenciamento não são alcançáveis em redes descentralizadas.

Quando pares não colaboram ou não executam corretamente o protocolo especificado é necessário responsabilizá-los de alguma forma, para inibir este tipo de comportamento. Conforme Marti & Garcia-Molina (2006) citado por Silva (2007), pares com comportamento incorreto podem ser divididos em egoístas ou maliciosos. Pares egoístas (também chamados de pares-carona ou “free-riders”) são empregados para obter o máximo possível do sistema, contribuindo com o mínimo possível de recursos. Pares maliciosos, em geral, objetivam prejudicar determinados pares ou mesmo a rede P2P como um todo.

4. CONCLUSÕES

A tendência é que se utilize cada vez mais aplicações relativas às redes P2P. Elas proporcionam aos seus usuários uma gama incrível de recursos, sendo que esses recursos podem ser divididos entre os vários pares componentes desta rede. São redes dinâmicas e geralmente disponibilizam serviços que na maioria das vezes teria um custo computacional muito grande em redes tradicionais. Por isso, a tecnologia P2P tende a se fortalecer junto aos usuários da internet e de ambientes corporativos. Hoje, P2P é uma grande chance para empresas que pretendem construir um novo tipo de negócio.

Uma vantagem dos sistemas de redes P2P é que eles são muito atrativos e difundidos mundialmente. São sistemas escaláveis, ou seja, não possuem um ponto central de falhas ou gargalo na forma de um servidor central; são redes que resistem melhor a ataques intencionais como os de negação de serviço assim como tem o poder de atrair um grande número de usuários em função dos benefícios oferecidos pela coletividade sem , no entanto, abrir mão da autonomia de seus participantes.

Entretanto, apesar de resistir melhor a ataques intencionais, são suscetíveis a falhas de segurança ou apresentam algum tipo de vulnerabilidade. Nesse sentido, é necessário que as aplicações desenvolvidas sejam confiáveis e seguras, considerando que cada

aplicação P2P possui seus requisitos próprios de segurança e que devem ter métodos de segurança instanciados conforme as suas necessidades.

Com relação ao “crash” da rede mundial P2P devido ao altíssimo fluxo de informações, uma possível solução seria a utilização do chamado P2P *caching*, onde um *ISP* armazena parte dos arquivos mais acessados pelos clientes P2P para poupar o acesso à Internet. Agindo desse modo o sistema permanecerá viável.

5. REFERÊNCIAS BIBLIOGRÁFICAS

ANDERSEN; D., BALAKRISHNAN, H., KAASHOEK, M., AND MORRIS, R. Resilient Overlay Networks. In: **Proc. ACM SOSP**, Oct. 2001.

ANDROUTSELLIS-THEOTOKIS, S.; SPINELLIS, D. A Survey of Peer-to-Peer Content Distribution Technologies. **ACM Computing Surveys**, New York, v.36 n.4, p. 335-371, 2004.

BARCELLOS, M. P.; GASPARY, L. P. **Fundamentos, Tecnologias e Tendências: rumo a Redes P2P Seguras**. 2006. Disponível em: https://www.sbc.org.br/bibliotecadigital/download.php?P2Per=64. Acesso em: 12 de Fev de 2009.

CAMPISTA, M. E. M.; DUARTE, O. C. M. B. **Segurança em Redes de Sensores**. Grupo de Teleinformática e Automação – Universidade Federal do Rio de Janeiro (UFRJ).

CARDOSO, A.R. **Sistemas Peer-to-Peer** (apostila)**.** Universidade Estadual do Ceara – UECE. Disponível em: http://www.larces.uece.br/~andrec/SD20082/SD_Capitulo02b.pdf. Acesso em: 10 de out. de 2009.

CHENG, A. AND FRIEDMAN, E. Sybil proof reputation mechanisms. **In P2PECON '05: Proceeding of the 2005 ACM SIGCOMM**

workshop on Economics of peer-to-peer systems, pages 128–132, New York, NY, USA. ACM Press. 2005.

COULOURIS, G., DOLLIMORE, J., KINDBERG, T. **Distributed Systems** : Concepts and Design. 4th Edition (2005)

DETSCH, ANDRÉ. **Uma Arquitetura para Incorporação Modular de Aspectos de Segurança em Aplicações Peer-to-Peer**. Dissertação (mestrado) - Universidade do Vale do Rio dos Sinos. Ciências Exatas e Tecnológicas Curso de Pós-Graduação em Computação Aplicada, São Leopoldo, BR/RS, 2005.

DOUCEUR, J. R. The sybil attack. In **1st International Workshop on Peer-to-Peer Systems**, pages 251–260. (2002).

DUMA, C.; SHAHMEHRI, N.; CARONNI, G. Dynamic trust metrics for peer-to-peer systems. Proceedings on **Sixteenth International Workshop on Database and Expert Systems Applications**. Pag. 776 – 781, Volume, Issue, August 2005.

FELDMAN, M., LAI, K., STOICA, I., AND CHUANG, J. Robust incentive techniques for peer-to-peer networks. **In EC '04: Proceedings of the 5th ACM conference on Electronic commerce**, pages 102–111, New York, NY, USA. ACM Press. 2004.

GALVÃO, RICARDO KLÉBER MARTINS. **SurRFE - Sub-rede de filtragens.** 81 pág. Dissertação (mestrado) - Universidade Federal do Rio Grande do Norte, Natal- RN, 2006. Centro de Tecnologia/ Programa de Pós-Graduação em Engenharia Elétrica.

GE, Z.; FIGUEIREDO, D. R.; JAISWAL, S.; KUROSE, J.; TOWSLEY; D. “Modeling Peer-to-Peer File Sharing Systems”. **In:** Proceedings of INFOCOMM 2003.

GIL, A. C. **Como elaborar projetos de pesquisa.** São Paulo: Atlas, 1991.

GIL, A. C. **Métodos e Técnicas de Pesquisa Social**. São Paulo: Atlas, 1999.

GRANVILLE, L.; DA ROSA, D.; PANISSON, A.; MELCHIORS, C.; ALMEIDA, M.J.; TAROUCO, L. Managing Computer Networks Using Peer-to-Peer Tecnologies. IEEE **Communications Magazine**, New York, v.43 n.10, p. 62-68, Oct. 2005.

HAARTSEN, J. et al. Bluetooth: Vision, Goals, and Architecture. **Mobile Computing and Communications Review**, [S.l.], v.2 n.4, p. 38-45, Oct. 1998.

INTERNET EXPERIMENT NOTE – IEN 111. (1979). Disponível em: http://www.postel.org/ien/txt/ien111.txt. Acesso em: 10 de out. de 2009.

KONRATH, MARLOM ALVES. **Estudo das vulnerabilidades da arquitetura BitTorrent, ataques e contramedidas possíveis**. Dissertação (mestrado) -- Universidade do Vale do Rio dos Sinos, Programa de Pós-Graduação em Computação Aplicada, 2007.

LAWTON, G. Is Peer-to-Peer Secure Enough for Corporate Use? **IEEE Computer**, 37(1):22-25, January 2004.

MARTI, S. AND GARCIA-MOLINA, H. **Taxonomy of trust: Categorizing P2P reputation systems.** Computer Networks, 50(4):472–484. 2006.

MATTAR, Fauze Najib. **Pesquisa de marketing v.1: metodologia, planejamento.** 5. ed São Paulo: Atlas, 1999.

MICHEL, M. H. **Metodologia e pesquisa científica em ciências sociais:** um guia prático para acompanhamento da disciplina e elaboração de trabalhos monográficos. São Paulo: Atlas, 2005.

MILOJICIC, S.; KALOGERAKI, V.; LUKOSE, R. Peer-to-peer Computing. **Palo Alto: HP Laboratories**, 2002. Technical Report.

MOREIRA, N. S. **Segurança mínima - uma visão corporativa da segurança de informações**. Vol. 1, Rio de Janeiro- RJ, 2001.

PANISSON, ANDRÉ. **Aplicação de Técnicas de Distribuição de Carga em Sistemasde Gerenciamento de Redes Baseados em P2P** . 81 pag. Dissertação (mestrado) – Universidade Federal do Rio Grandedo Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2007. Orientadora: Maria Janilce B. Almeida.

PEERMETRICS. **PeerMetrics Peer System**. Draft. Disponível em: <http://www.peermetrics.com>. Acesso em: ago. 2003.

PINHEIRO. M.C.M.A. **Uma Arquitetura P2P Baseada na Hierarquia do ndereçamento IP com Roteamento Unificado**. Universidade Federal

do Rio Grande do Norte, Natal, RN – 2006. Tese de Doutorado/ Eng. Elétrica, 137 pag.

RFC760 – **DoD standard Internet Protocol.** Defense Advanced Research Projects Agency. Information Processing Techniques Office, 1400 Wilson Boulevard Arlington, Virginia 22209. 1990.

RICHARDSON, Roberto Jarry; PERES, Jose Augusto de Souza. . **Pesquisa social:** metodos e tecnicas. 3. ed. rev. ampl São Paulo: Atlas, 1999.

RIPEANU, M.; IAMNITCHI, A.; FOSTER, I. Mapping the Gnutella network: Properties of large- scale peer-to-peer systems and implications for system design. **IEEE Internet Computing Journal**, New York, v.6, n.1, p. 50-57, 2002.

RIVEST, R., SHAMIR, A., ADLEMAN, L. **A Method for Obtaining Digital Signatures and Public Key Cryptosystems. Communications of the ACM**, 21:120, 126, February 1978.

ROCHA, J., DOMINGUES, M. A., CALLADO, A., SOUTO, E., SILVESTRE G., KAMIENSKI, C. A., AND SADOK, D. **Peer-to-Peer: Computação Colaborativa na Internet.** Minicursos SBRC2004 (capítulo de livro), pp. 3-46, Maio 2004.

ROCHA, R. R. da. **Redes *Peer-to-Peer* para Compartilhamento de Arquivos na Internet**. Grupo de Teleinformática e Automação PEE/COPPE - DEL/POLI. UFRJ, Rio de Janeiro, Brasil. Disponível em < http://www.gta.ufrj.br>. Acesso em 07/08/2006.

ROETTGERS, JANKO. **5 Ways to Test Whether your ISP throttles P2P**. Disponível em: http://newteevee.com/2008/04/02/5-ways-to-test-if-your-isp-throttles-p2p/. Acesso em: 10 de out. de 2009.

SCHOLLMEIER, R. A Definition of Peer-to-Peer Networking for the Classification of Peer-to-Peer Architectures and Applications. Proceedings of the First International Conference on Peer-to-Peer Computing, **IEEE (2002).**

SCHOLLMEIER; R. **A Definition of Peer-to-Peer Networking for the Classification of Peer-to-Peer Architectures and Applications**. Institute of Communication Networks, Technische Universität München. Germany. Disponível em: <http://csdl2.computer.org/comp/proceedings/P2P/2001/1503/00/15030101.pdf>. Acessado em 02/05/2006.

SILVA, JULIANO FREITAS DA. **Métodos para Contenção de Poluição em Redes**. 71 pg. Dissertação (mestrado) - Universidade do Vale do Rio dos Sinos. Ciências Exatas e Tecnológicas Programa Interdisciplinar de Pós-Graduação em Computação Aplicada, São Leopoldo, BR/RS, 2007.

SOUSA JR, R. T. DE; ALBUQUERQUE, R. DE O.; HANASHIRO, M.; SILVA, Y. A. DA; GONDIM, P. R. DE L. Towards Establishing Trust in MANET: an Integrated Approach for Auto-configuration, Authentication and Certification. **The International Journal of Forensic Computer Science**, Vol. 1, N° 1, 2006.

STOICA, I.; MORRIS, R.; LIBEN-NOWELLY, D.; KARGER, D.; KAASHOEK, M. F.; DABEK, F.; BALAKRISHNAN, H.. Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications. In: **Proceedings of SIGCOMM,** 2001.

STOICA, I., ROBERT I., MORRIS, R., KARGER, D., KAASHOEK, M. F. AND BALAKRISHNAN, H. Chord: A scalable peer-to-peer lookup service for internet applications. In **Proceedings of the ACM SIGCOMM**, pp. 149-160, 2001.

SURYANARAYANA, G.; TAYLOR, RICHARD N. **A Survey of Trust Management and Resource Discovery Technologies in Peer-to-Peer Applications**. Institute for Software Research, University of California, ISR Technical Report . UCI-ISR-04-6, July 2004.

THE ANONYMIZER. Disponível em: http://anonymizer.com. Acesso em: 08 de fev de 2009.

THEOTOKIS, S. A. AND SPINELLIS, D. A survey of peer-to-peer content distribution technologies, 2004. **ACM Computing Surveys**, 36(4):335–371.

TRAN, H., HITCHENS, M., VARADHARAJAN, V., AND WATTERS, P. A trust based access control framework for P2P file-sharing systems. **In HICSS '05: Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05)** - Track 9, Washington, DC, USA. IEEE Computer Society. . (2005).

TRIVINOS, Augusto Nibaldo Silva. **Introdução a pesquisa em ciências sociais:** a pesquisa qualitativa em educação. São Paulo: Atlas, 2006.

TRUELOVE, K. **Gnutella and the Transient Web**. O'Reilly P2P, 2001. Disponível em: <http://www.openP2P.com/lpt/a/705>. Acesso em: jul. 2005.

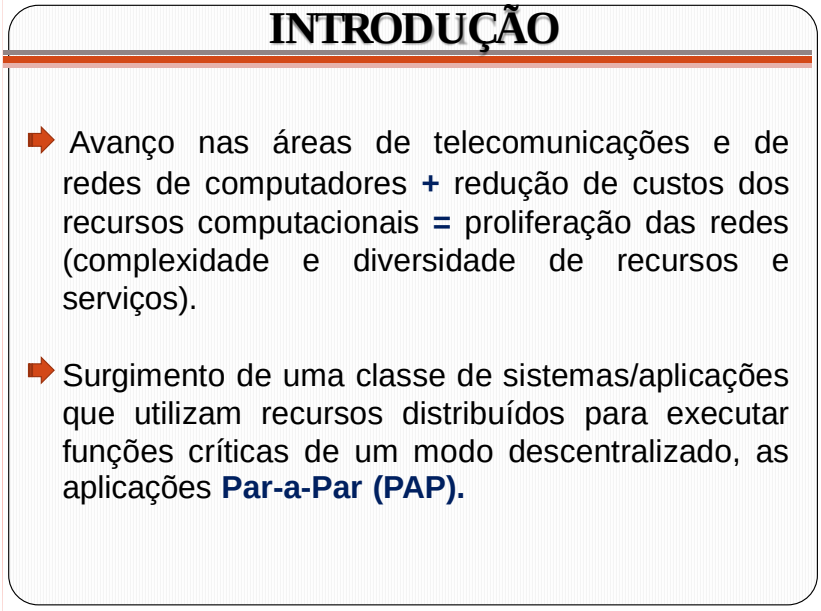
VILANOVA, FELIPE JUNG. **Uma Ferramenta Peer-to-Peer para Gerenciamento Cooperativo de Redes**. 63 f.:il. Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação. Porto Alegre, BR – RS, 2005.

WIKIPÉDIA. **Criptografia de chave pública**. 2009. Disponível em: <http://pt.wikipedia.org/wiki/Criptografia_de_chave_p%C3%BAblica>. Acesso em 18 de out de 2009.

ZIHUI, G., FIGUEIREDO, D. R., JAISWAL. S., KUROSE, J., TOWSLEY, D."Modeling peer-peer file sharing systems," **in IEEE Infocom 2003**, vol. 3, 2003, pp. 2188 – 2198.

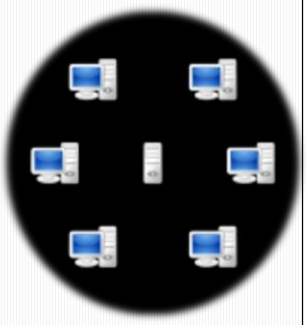
6. ANEXOS


Universidade Federal de Lavras
Departamento de Ciência da Computação
REDES PAR-A-PAR (PAP) E SEGURANÇA DA INFORMAÇÃO
Marcelo Messora Miranda
Orientador
Luiz Henrique Andrade Correia


INTRODUÇÃO
Avanço nas áreas de telecomunicações e de redes de computadores + redução de custos dos recursos computacionais = proliferação das redes (complexidade e diversidade de recursos e serviços).
Surgimento de uma classe de sistemas/aplicações que utilizam recursos distribuídos para executar funções críticas de um modo descentralizado, as aplicações Par-a-Par (PAP).

- **Aplicações da PAP:** compartilhamento de arquivos, serviços de mensagens instantâneas, processamento distribuído, *webcaching*, jogos, disseminação de conteúdo, backup distribuído, telefonia IP, gerência de redes (Granville et al., 2005).

- **Objetivo da PAP:** visa a **descentralização, a escalabilidade e a tolerância à falhas** (Vilanova, 2006).



MOTIVAÇÃO

- A motivação está no intuito de que as redes PAP sejam amplamente adotadas, sendo que para isso elas precisam estar protegidas da ação maliciosa de diversos pares.

Diante dos freqüentes ataques sofridos pelo sistema PAP, o objetivo da segurança, no que tange à informação, é a busca da disponibilidade, confidencialidade e integridade dos seus recursos e da própria informação.

OBJETIVO

Discorrer sobre dois temas que tem recebido a atenção tanto da comunidade científica como da indústria: **Redes Par-a-Par (PAP)** e **Segurança da Informação.**

METODOLOGIA

Pesquisa exploratória (abordagem detalhada sobre PAP) **e descritiva** (estratégias de subversão e de ataque que possam ser empregados para avaliar a segurança das redes).

REFERENCIAL TEÓRICO

Sistemas PAP

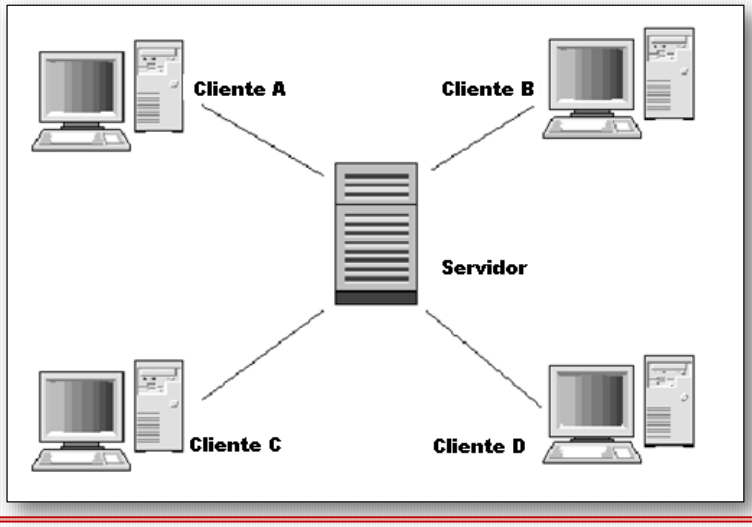
- Classe de sistemas e aplicações que utilizam recursos distribuídos para executar funções críticas (processamento distribuído, troca de arquivos, comunicação e colaboração, ou serviços de plataforma) de um modo descentralizado. Os recursos incluem poder de processamento, dados, banda, e presença.

  Milojicic et al. (2002)

- **Modelo cliente/servidor:** pode efetuar pedidos e serviços mas não pode disponibilizá-los. O modelo ***Par-a-Par*** **(PAP)**, dá a máquinas individuais a capacidade de fornecer serviços umas às outras.

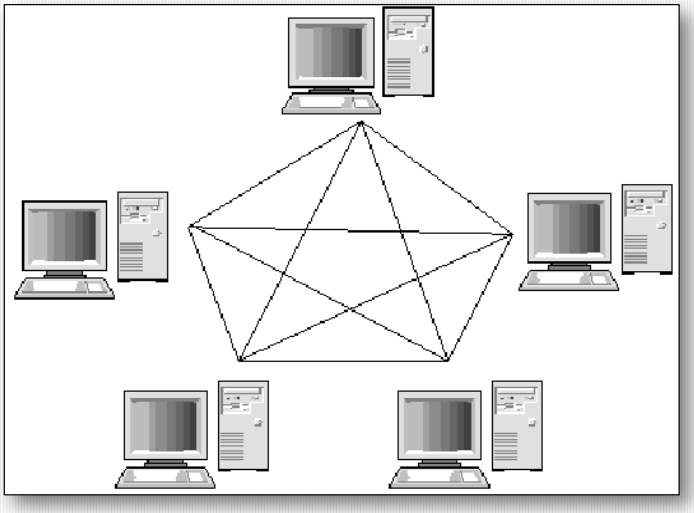
  (Vilanova, 2006)

Modelo Cliente / Sevidor



Fonte: Vilanova, 2006

Modelo PAP



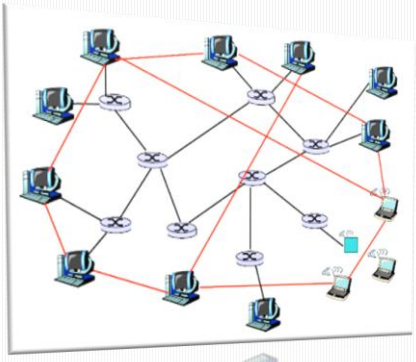
Fonte: Vilanova, 2006

Classificação	Descrição (Ge et al., 2003)
CIA	Contêm um servidor central ou um cluster de servidores (responder pelos pedidos de busca e realização de todas as tarefas de manutenção da infra-estrutura. O principal exemplo e precursor desta arquitetura foi o *NAPSTER*.
DIFA	Completa descentralização, onde os mecanismos de busca e manutenção da infra-estrutura estão distribuídos pela rede, onde cada nó é responsável por manter a listagem dos seus próprios arquivos. Ex: *Gnutella*.
DIHA	Descentralizado, onde cada nó é responsável por um subconjunto do espaço total de índices, onde o nó que entra na rede recebe um espaço do conjunto dos índices dos arquivos. Ao sair da rede, esta deverá designar estes índices para outro nó. Ex: *Chord*.

- **Classificação de redes PAP por Schollmeier (2006):**
  - **Redes puras:** os nós são responsáveis por todas as transações entre si, além de gerenciarem todas as informações que sejam relevantes para a aplicação que se utilize dessa rede.
  - **Redes híbridas:** existem servidores centrais responsáveis pela execução de tarefas consideradas como críticas.

PAP representa uma considerável fração de tráfego na Internet (Panisson, 2007) ➡ ***"crash"*** da rede mundial (Roettgers, 2009).

Redes Overlays



(Cardoso, 2009)

Rede *overlay* (ou rede sobreposta) é uma rede de computadores construída em cima de outra rede. Os nós na sobreposição podem ser concebidos como sendo conectados por ligações virtuais ou lógicas. A rede PAP também configura um tipo de rede *overlay*.

(Andersen et al, 2001)

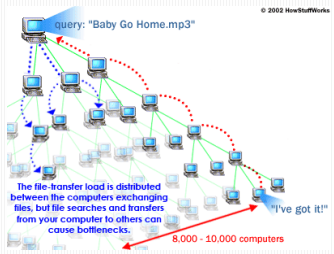
Arquiteturas PAP

- Classificações das redes *overlay* (Rocha et al (2004):

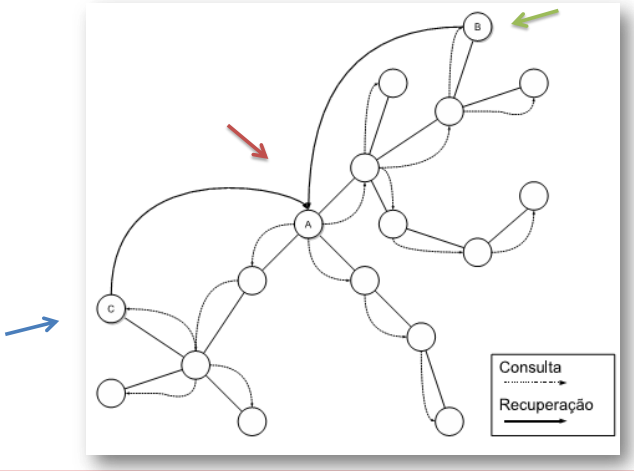
- **Centralizadas:** um nó central mantém informações sobre todos os nós e recursos da rede. *Napster;* sistemas de mensagens instantâneas.
- **Descentralizadas e não estruturadas:** as consultas são propagadas de nó em nó até que encontrem o destino ou que algum mecanismo de *timeout* encerre o processo. *Gnutella* e *Kazaa.*
- **Descentralizadas e estruturadas.** Utilizam regras para distribuir os dados na rede de modo que facilite a posterior localização dos mesmos. DHT (*Tabela Hash Distribuída*), como Chord, CAN e Pastry.

Exemplos de Redes Par-a-Par

Gnutella: compartilhamento de arquivos na Internet (topologia *ad hoc)* que permite a busca de arquivos através de seus nomes, ou partes dele, e a posterior obtenção do arquivo diretamente da máquina de outros usuários. Não existe um diretório centralizado, controle sobre a topologia da rede e dos locais onde os dados são armazenados. (Pinheiro, 2006)



Gnutella: nodo A faz uma busca por inundação, encontra recurso procurado em B e C, e após interage com os mesmos para obter tal recurso.

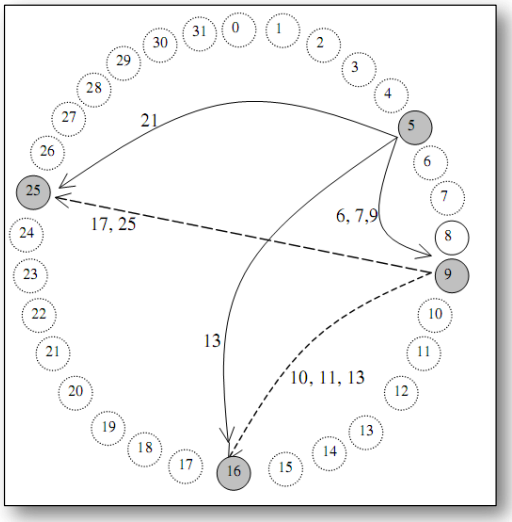


(Barcelos & Gaspari, 2006)

Chord: é normalmente utilizado para armazenar pares contendo uma chave e seu valor associado em nós distribuídos pela rede. O serviço de busca da rede permite que, dada uma chave, seja determinado o nó responsável pela mesma (Stoica et al, 2001).

❖O espaço de identificadores é formado por um anel conectado, onde cada identificador possui m bits. Tanto os nós quanto os dados a serem armazenados são mapeados através de uma função *hash*. O mapeamento dos nós é feito aplicando-se a função *hash* ao seu endereço IP. O número de nós consultados em uma busca é O(log N) (Pinheiro, 2006).

Rede ***Chord*** com 4 nós e 32 identificadores



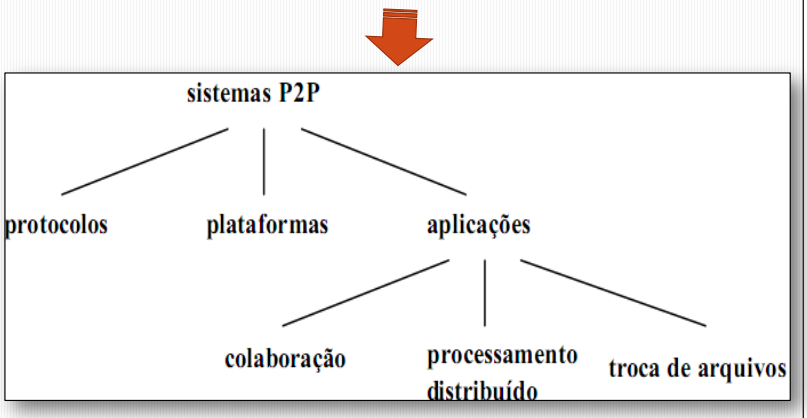
(Pinheiro, 2006)

- ***Kademlia:*** Infraestrutura de roteamento que usa um mecanismo inovador para roteamento de mensagens e busca de objetos segundo uma métrica de distância entre identificadores de nodos baseada em *xor*.

  - A topologia tem a propriedade que toda a mensagem trocada carrega ou reforça informações úteis de contato.

  - Ex: *Overnet*, e*Donkey* e *eMule*, além de *BitTorrent,* que emprega *Kademlia* para permitir o uso de *torrents* sem um *tracker* (Barcelos & Gaspary, 2006).

Categorias de Sistemas PAP



Vilanova (2006)

Protocolos

- Tecnologias que definem padrões para comunicação em redes PAP, utilizados como base para o desenvolvimento das aplicações ou plataformas PAP (Vilanova, 2006).

Ex:*Gnutella* (Ripeanu et al, 2002). Entre os programas que utilizam esse protocolo, estão o *BearShare*, *LimeWire*, *Azureus* e agora o *Shareaza*.

Plataformas

- Sistemas que oferecem os componentes PAP básicos, como descoberta, comunicação, segurança e agregação de recursos. Base para o desenvolvimento e utilização das aplicações (VILANOVA, 2006).

Após a criação dos sistemas PAP houve problema da comunicação entre os diferentes sistemas existentes.

Necessidade de criação de plataformas para o desenvolvimento de redes PAP que permitissem a comunicação entre si. Ex: *JXTA*, o *Windows Peer-to-Peer Networking* (redes PAP do Windows), o *XNap* e o *Bluetooth* (Haartsen et al, 1998).

Aplicações

- São os sistemas com funcionalidades específicas que podem ser subdivididas em outras três categorias: **processamento distribuído**, **colaboração** e **troca de arquivos.**

- As aplicações de processamento distribuído utilizam o poder computacional disponível dos seus usuários para formar supercomputadores, dividindo grandes tarefas em pequenos pedaços (Vilanova, 2006). Exemplos de aplicações bastante conhecidas estão o *ICQ*, o MSN e o *BearShare*.

Fundamentos de Segurança da Informação

- **Segurança em redes PAP** (Detsch, 2005):

  - Segurança de uma rede/instituição: bloqueio do tráfego gerado por aplicações de compartilhamento de arquivos ou prevenção que tais aplicações sirvam de porta de entrada para vírus ou *trojan horses.*

  - Criação de aplicações PAP seguras: autenticação, confidencialidade, integridade e autorização.

Camada de Robustez

- Diretriz geral para o desenvolvimento de software que opera ou controla a infra-estrutura da Internet ou outras redes baseadas em Protocolos da Internet.

- Centralizar os mecanismos de segurança é uma solução que anula os benefícios de uma estrutura descentralizada. Outro aspecto relativo à segurança é a utilização de sistemas de criptografia para a transmissão dos dados (Vilanova, 2006).

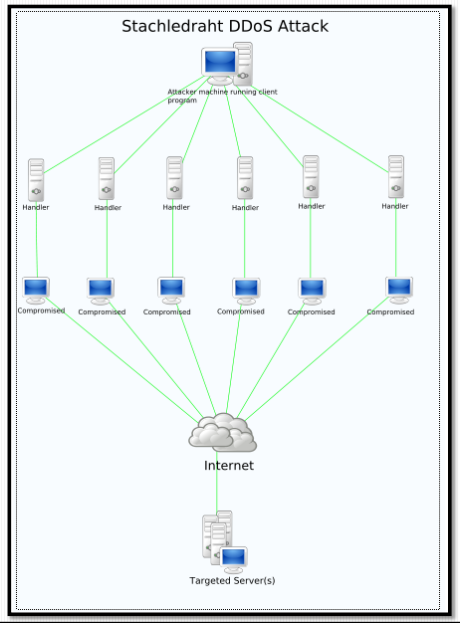
Principais Aspectos de Segurança da Informação

- **Disponibilidade:** Parcela de tempo que um determinado objeto ou serviço está disponível para acesso. Em sistemas PAP para a computação distribuída é a garantia de que os pares estão acessíveis para realizar o processamento esperado (Silva, 2007).

- A principal contramedida para ataques de disponibilidade refere-se à replicação. Os principais ataques à disponibilidade estão baseados em negação de serviço DoS – *Denial of Service* ou “negação de serviço” e ataques de roteamento.

Diagrama de um Ataque *DDoS Stachledraht*

(Wikipédia, 2009)



- **Confidencialidade:** visa manter o sigilo, o segredo ou a privacidade das informações, evitando que pessoas, entidades ou programas não-autorizados tenham acesso às mesmas (Moreira, 2001).

  A garantia de confidencialidade dos dados está relacionada ao uso de um algoritmo de cifragem, ou criptografia (Detsch, 2005). Algoritmos com esta finalidade podem ser divididos em dois grupos: **criptografia simétrica** e **criptografia assimétrica**.

- **Troca de chaves:**

  Criptografia simétrica: é necessário o uso de um canal seguro para transmissão da chave (Detsch, 2005).

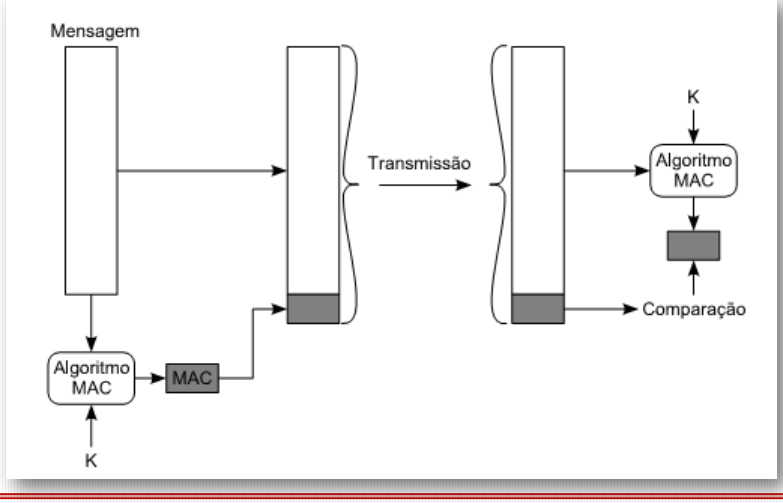
  Criptografia assimétrica: a chave pública é distribuída livremente para todos os correspondentes, enquanto a chave privada deve ser conhecida apenas pelo seu dono (Wikipédia, 2009).

- **Autenticação:** assegura que alguém (ou algo) é de fato quem (ou o que) afirma ser (Silva, 2007).

- **Autenticação Com Criptografia:** é possível prover autenticação simplesmente usando criptografia simétrica (Detsch, 2005).

- **Autenticação Sem Criptografia:** utilização de apenas uma *tag* de autenticação (Detsch, 2005).

Autenticação de Mensagem Utilizando Código de Autenticação (MAC)



Detsch, 2005

- **Integridade:** Consiste em proteger a informação contra qualquer tipo de alteração, sem a autorização explícita do autor da mesma (Moreira, 2001). Pode ser aplicada a um fluxo de mensagens, a uma mensagem individual ou a alguns campos da mensagem.

Ataques à integridade mais típicos (Barcellos & Gaspary, 2006) são: o ataque de poluição a arquivos e o ataque de resposta falsa.

- **Não-repúdio:** compreende os esforços dispendidos para garantir a autoria de determinadas ações (Moreira, 2001). Sua principal aplicação está em transações comerciais ou financeiras (Detsch, 2005)

- **Autorização:** é a capacidade de restrição de acesso aos recursos baseando-se em informações sobre o requisitante, o recurso a ser acessado e os detalhes específicos da requisição. (Silva, 2007)

- **Auditoria:** permite análise do funcionamento dos mecanismos de segurança durante ou após a sua execução. A técnica mais simples e difundida com este objetivo consiste na geração de logs de execução. (Detsch, 2005)

- **Anonimicidade e Negabilidade:** garante que as identidades reais no sistema permanecerão desconhecidas. (Silva, 2007)

- **Reputação:** visa permitir um grau de confiança relativo aos demais nós da rede. Com isso, é possível associar um "risco" ao uso ou prestação de um serviço baseado na reputação da outra parte envolvida.

  Formas de ataque a sistemas de reputação:
  - ❖Ataque de nó *traidor (*Marti & Garcia-Molina, 2006);
  - ❖Conspiração contra sistemas de reputação (Cheng & Friedman (2005);
  - ❖*Whitewashing* (Feldman et al., 2004).

  Observa-se que, no geral, **confidencialidade**, **autenticação** e **integridade** são os aspectos mais importantes, seguidos de **autorização** e **auditoria** (Detsch, 2005).

- **Confiança em Redes PAP:** Um dos desafios em redes e sistemas PAP está relacionado com os riscos de interação e colaboração com entidades desconhecidas e que podem ser potencialmente maliciosas (Duma et al., 2005).

  Segundo Sousa et al. (2006) , um fato importante a ser considerado em um modelo de segurança em redes está relacionado com a certificação digital.

  Conforme Marti & Garcia-Molina (2006), pares com comportamento incorreto podem ser divididos em **egoístas** ou **maliciosos**.

CONCLUSÕES

- **PAP:** são atrativos e difundidos em função dos benefícios oferecidos pela coletividade sem, no entanto, abrir mão da autonomia.

- Apesar de resistir melhor a ataques intencionais, são suscetíveis a falhas de segurança ou vulnerabilidade. Nesse sentido, é necessário que as aplicações desenvolvidas sejam confiáveis e seguras.

- Para o “crash” da rede mundial, uma possível solução para PAB seria a utilização do PAP *caching*.

- Polêmica: criação do Conselho Soberano de Internet x acesso de dados de usuários

REFERÊNCIAS BIBLIOGRÁFICAS

ANDERSEN; D., BALAKRISHNAN, H., KAASHOEK, M., AND MORRIS, R. Resilient Overlay Networks. In: **Proc. ACM SOSP**, Oct. 2001.

ANDROUTSELLIS-THEOTOKIS, S.; SPINELLIS, D. A Survey of Peer-to-Peer Content Distribution Technologies. **ACM Computing Surveys**, New York, v.36 n.4, p. 335-371, 2004.

BARCELLOS, M. P.; GASPARY, L. P. **Fundamentos, Tecnologias e Tendências: rumo a Redes PAP Seguras**. 2006. Disponível em: https://www.sbc.org.br/bibliotecadigital/download.php?paper=64. Acesso em: 12 de Fev de 2009.

CAMPISTA, M. E. M.; DUARTE, O. C. M. B. **Segurança em Redes de Sensores**. Grupo de Teleinformática e Automação – Universidade Federal do Rio de Janeiro (UFRJ).

CARDOSO, A.R. **Sistemas Peer-to-Peer** (apostila)**.** Universidade Estadual do Ceara – UECE. Disponível em: http://www.larces.uece.br/~andrec/SD20082/SD_Capitulo02b.pdf. Acesso em: 10 de out. de 2009.

CHENG, A. AND FRIEDMAN, E. Sybil proof reputation mechanisms. **In PAPECON 05: Proceeding of the 2005 ACM SIGCOMM workshop on Economics of peer-to-peer systems**, pages 128–132, New York, NY, USA. ACM Press. 2005.

COULOURIS, G., DOLLIMORE, J., KINDBERG, T. **Distributed Systems** : Concepts and Design. 4th Edition (2005)

REFERÊNCIAS BIBLIOGRÁFICAS

DETSCH, ANDRÉ. **Uma Arquitetura para Incorporação Modular de Aspectos de Segurança em Aplicações Peer-to-Peer**. Dissertação (mestrado) – Universidade do Vale do Rio dos Sinos. Ciências Exatas e Tecnológicas Curso de Pós-Graduação em Computação Aplicada, São Leopoldo, BR/RS, 2005.

DOUCEUR, J. R. The sybil attack. In **1st International Workshop on Peer-to-Peer Systems**, pages 251–260. (2002).

DUMA, C.; SHAHMEHRI, N.; CARONNI, G. Dynamic trust metrics for peer-to-peer systems. Proceedings on **Sixteenth International Workshop on Database and Expert Systems Applications**. Pag. 776 – 781, Volume, Issue, August 2005.

FELDMAN, M., LAI, K., STOICA, I., AND CHUANG, J. Robust incentive techniques for peer-to-peer networks. **In EC '04: Proceedings of the 5th ACM conference on Electronic commerce**, pages 102–111, New York, NY, USA. ACM Press. 2004.

GALVÃO, RICARDO KLÉBER MARTINS. **SurRFE - Sub-rede de filtragens.** 81 pág. Dissertação (mestrado) - Universidade Federal do Rio Grande do Norte, Natal- RN, 2006. Centro de Tecnologia/ Programa de Pós-Graduação em Engenharia Elétrica.

GIL, A. C. **Como elaborar projetos de pesquisa.** São Paulo: Atlas, 1991.

GIL, A. C. **Métodos e Técnicas de Pesquisa Social**. São Paulo: Atlas, 1999.

REFERÊNCIAS BIBLIOGRÁFICAS

GRANVILLE, L.; DA ROSA, D.; PANISSON, A.; MELCHIORS, C.; ALMEIDA, M.J.; TAROUCO, L. Managing Computer Networks Using Peer-to-Peer Tecnologies. IEEE **Communications Magazine**, New York, v.43 n.10, p. 62-68, Oct. 2005.

HAARTSEN, J. et al. Bluetooth: Vision, Goals, and Architecture. **Mobile Computing and Communications Review**, [S.l.], v.2 n.4, p. 38-45, Oct. 1998.

INTERNET EXPERIMENT NOTE – IEN 111. (1979). Disponível em: http://www.postel.org/ien/txt/ien111.txt. Acesso em: 10 de out. de 2009.

MARTI, S. AND GARCIA-MOLINA, H. **Taxonomy of trust: Categorizing PAP reputation systems.** Computer Networks, 50(4):472–484. 2006.

MICHEL, M. H. **Metodologia e pesquisa científica em ciências sociais:** um guia prático para acompanhamento da disciplina e elaboração de trabalhos monográficos. São Paulo: Atlas, 2005.

VILANOVA, FELIPE JUNG. **Uma Ferramenta Peer-to-Peer para Gerenciamento Cooperativo de Redes**. 63 f.:il. Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação. Porto Alegre, BR – RS, 2005.

WIKIPÉDIA. **Criptografia de chave pública**. 2009. Disponível em: <http://pt.wikipedia.org/wiki/Criptografia_de_chave_p%C3%BAblica>. Acesso em 18 de out de 2009.

ZIHUI, G., FIGUEIREDO, D. R., JAISWAL. S., KUROSE, J., TOWSLEY, D.“Modeling peer-peer file sharing systems,” **in IEEE Infocom 2003**, vol. 3, 2003, pp. 2188 – 2198.