

(junto com a mensagem de resposta de requisitos) as exigências desse nó. Novamente, o novo nó terá um gargalo no link de saída para responder a todos esses pedidos e alto custo computacional para atender a todas as requisições.

Se a rede P2P apresentar alta transiência de nós em um ambiente com uma quantidade de nós substancial, o protocolo de troca de requisitos irá esgotar a capacidade de comunicação e processamento dos nós da rede, pois cada nó que entra na rede irá gerar, no mínimo,  $3N$  mensagens, onde  $N$  é o número de nós e o 3 são as mensagens *Requeriments\_request*, *Requeriments\_request\_and\_reply* e *Reply*. Deve-se somar a esse número as requisições de chaves públicas feitas a autoridade.

## 5.3 Segurança

Nesta Seção serão discutidos os tipos de ataques conhecidos na literatura e como estes podem impactar sobre a P2PSL e as aplicações desenvolvidas com sua utilização. Mais especificamente, os principais tipos de ataques expostos serão os apresentados em [34]. É importante salientar que a P2PSL foi pensada como uma camada para incorporação de segurança de maneira a deixar as aplicações *Peer-to-Peer* livres para construírem sua solução final. Essa liberdade permite que vários tipos de aplicações sejam construídas e o desenvolvedor da aplicação deverá inserir as restrições necessárias ao sistema satisfazendo suas necessidades. A P2PSL auxiliará a criação fornecendo os requisitos de segurança providos por seus módulos.

### 5.3.1 Negação de Serviço

A negação de serviço afeta a disponibilidade do sistema e pode ser executada de várias formas sendo que cada forma terá uma eficácia e um consumo diferente. A maneira tradicional de efetuar um ataque desta natureza é o esgotamento das capacidades de comunicação de um nó, ou seja, um nó atacante ou conjunto de nós atacantes enviam uma grande quantidade de dados para o nó atacado. Se atacantes tiverem a sua disposição uma grande quantidade de recursos, será praticamente impossível evitar esse tipo de ataque. Do ponto de vista da P2PSL não há nada que se possa fazer para contornar esse problema, pois o mesmo é um ataque do nível de rede.

Além dos ataques de negação de serviço do nível de rede, existem os ataques que atuam no nível de aplicação onde a intenção é esgotar as capacidades de processamento de uma máquina ou sistema para que não haja respostas a requisições legítimas do sistema. Algumas vezes, os ataques que atuam nesse nível tentam fazer com que a aplicação de um nó atacado se comporte arbitrariamente e produza dados inválidos na rede introduzindo alta sobrecarga. A P2PSL não oferece proteção direta contra esse tipo de ataque, pois ela somente suporta a garantia da origem da mensagem e um nó conhecido pode fazer esse tipo de ataque.

A P2PSL não oferece suporte para situações onde nós respondem corretamente as buscas de informações mas se negam a entregar o conteúdo solicitado, pois não existe módulos que façam controle de reputação. Em contrapartida, o módulo de autenticação provido pela P2PSL evita ataques de injeção de mensagens não solicitadas na rede, pois mensagens não poderão ser forjadas e então serão ignoradas pelos outros nós que estejam utilizando a P2PSL.

Outro ataque descrito é o ataque de nó lento (*slow node attack*). Esse ataque consiste em um nó malicioso interceptar mensagens de nós não-maliciosos modificando-as de forma a falsificar informações sobre capacidade de banda e processamento. A falsificação faz com que os receptores destas mensagens pensem que um nó tem uma capacidade maior

do que a real, com isso o sistema pode tentar usar esse nó para desafogar certa parte da rede causando um prejuízo tanto ao nó quanto aos interessados em seus serviços. O módulo de autenticidade resolve facilmente esse problema, pois as mensagens, depois de assinadas, não podem ser modificadas.

### 5.3.2 Roteamento

Esse tipo de ataque afeta o substrato de rede das aplicações P2P e tem o intuito de fazer com que o sistema fique lento ou não responda as requisições. Esse ataque acontece quando nós maliciosos “envenenam” as tabelas de roteamento de nós corretos ou quando nós maliciosos roteiam as mensagens por caminhos errados ou para nós inválidos, fazendo com que as mensagens se percam na rede.

Como a P2PSL foi proposta para atuar na camada de aplicação, a maioria dos ataques dessa natureza não poderá ser impedido ou detectado por ela. Dessa forma, outros métodos precisam ser desenvolvido para contornar os problemas gerados por esse tipo de ataque, tanto em nível de aplicação quanto em nível de rede.

A implementação atual da P2PSL está baseada no substrato de rede JXTA e portanto os ataques que podem comprometer o JXTA também serão capazes de comprometer a P2PSL. Porém, pode-se incluir mecanismos de controle no nível da aplicação na tentativa de barrar alguns dos ataques. A seguir, os ataques de roteamento serão descritos analisando o impacto para a P2PSL.

[35] descreve três tipos de ataques desta natureza em redes DHT. O primeiro acontece quando um nó atacante que faz parte do *overlay*, ao receber mensagens, encaminha as mesmas para nós inexistentes na rede ou as envia por rotas incorretas. Com isso, as mensagens se perdem pela rede, mas os nós não irão desconfiar do nó atacante pois ele está apenas encaminhando as mensagens. O segundo tipo acontece quando nós maliciosos inserem mensagens de atualização de rotas no *overlay* de forma a enganar os nós corretos na rede sobre a rota de outros nós. Com isso, nós sádios irão enviar mensagens por caminhos incorretos pois foram enganados e tiveram suas tabelas de roteamento poluídas.

O terceiro trata de partições na rede que ocorrem quando nós maliciosos formam uma rede paralela e fazem com que, erroneamente, um nó sadio entre nessa rede. Com o controle da rede, os nós maliciosos podem enganar nós sádios negando-lhes serviços ou ainda distribuindo conteúdo poluído. Outra consequência desse ataque, também conhecido como Eclipse, é que a rede paralela pode conter nós que fazem a ligação entre a rede falsa e a rede normal monitorando os nós sádios para verificar que tipo de buscas eles estão fazendo. Isso termina com a anonimidade de uma rede.

Para efetuar um ataque do tipo Eclipse pode-se utilizar duas abordagens. A primeira diz respeito a manipulação do algoritmo de manutenção do *overlay* onde nós maliciosos irão envenenar tabelas de roteamento ou tabelas de nós disponíveis para conexão enganando parte da rede. A segunda forma diz respeito a utilização de ataque Sybil [36] para conseguir grande quantidade de identificadores para controlar uma porção da rede.

P2PSL não fornece suporte direto para resolver esses problemas, pois os mesmos atuam diretamente no substrato de rede. Sybil é um problema de alocação de identificadores e isto não é coberto pela P2PSL, assim como roteamento. Porém, com a utilização do módulo de autenticidade e a possibilidade de descobrir se alguma mensagem está envenenada é possível determinar quem está gerando essas mensagens e colocá-lo em uma lista negra.

### 5.3.3 Autenticidade

Assegurar identidade de um nó na rede é uma tarefa difícil de ser resolvida definitivamente, pois sempre haverá a necessidade de confiarmos em alguém ou alguma entidade e neste momento a vulnerabilidade aparece. Um nó pode assegurar digitalmente a autenticidade de outro nó, basicamente, de três maneiras distintas: agência de confiança, próprio nó ou outros nós.

Teoricamente, é fácil assegurar que uma mensagem não é forjada ou garantir o remetente de uma mensagem, pois para isso basta usar assinaturas digitais. O problema se encontra em como verificar a quem essa “identidade digital” pertence. Fazendo uso deste problema, o ataque Sybil acontece quando um único nó pode criar várias identidades digitais dentro da rede P2P para enganar outro nó.

Apesar da P2PSL garantir a autenticidade dos remetentes das mensagens, ela não é capaz de detectar se um ataque do tipo Sybil está presente na rede. Isso acontece, pois um nó pode gerar vários pares de chave pública/privada e depositar essas chaves na autoridade de chave pública. Depois disso, o nó pode instanciar várias vezes a aplicação P2P localmente e utilizar para cada uma delas um dos pares de chaves criados. Esse nó de posse das várias identidades pode comprometer algoritmos de replicação, pois várias das réplicas podem acabar sendo alocadas na mesma entidade física que possui várias entidades digitais.

A P2PSL também não oferece suporte para impedir que nós gerem suas identidades. Essa restrição fica a cargo da aplicação, mas vale ressaltar que é difícil impedir que um nó instancie várias vezes a aplicação para conseguir várias identidades.

### 5.3.4 Reputação e Confiança

Aparentemente, não existe, hoje, um sistema de reputação que resolva os vários problemas provenientes de nós que não querem cooperar (agindo de má fé) ou de nós que estão atacando a rede ativamente. Existem na literatura muitas propostas tentando endereçar alguns dos problemas de reputação, porém eles apresentam fraquezas e não resolvem o problema como um todo.

O problema de reputação não acontece apenas no mundo digital, mas também no mundo real onde pessoas podem enganar outras, agir de má fé, passar informações errôneas ou caluniosas. O grande dilema está em quem confiar quando se trata de uma interação com outro nó. A única informação realmente segura que se tem é quando existe conhecimento e interação prévia com o nó, pois não podemos garantir que outros nós enviarão informações confiáveis sobre o nó em questão. Mas e quando se interage com um nó pela primeira vez? Qual a abordagem deve ser tomada?

Quando se pensa nesses sistemas é necessário avaliar qual *tradeoff* entre a imposição de mecanismos para controlar todas as variáveis em um sistema de reputação e a utilização do sistema, pois a utilização pode ficar bastante deteriorada dependendo da quantidade de restrições feitas.

Existem muitos casos em que sistemas de reputação são interessantes para garantir alguns requisitos ou a melhor utilização do sistema como um todo. Um exemplo disso é utilizar reputação para beneficiar ou punir um usuário dependendo da sua colaboração com a rede. Assim um nó que tentasse atuar como um *free-rider* apenas consumindo recursos começaria a ser barrado com o tempo até que contribuísse com os outros nós.

Outro exemplo é o de comércio eletrônico como o eBay e o Mercado Livre onde os usuários do sistema classificam uns aos outros. Essa classificação ocorre a partir da

experiência que um vendedor teve com um comprador. Geralmente, somente usuários com boa qualificação conseguirão vender seus produtos sem problemas ou desconfiança. Nesse modelo, o eBay ou Mercado Livre atuam como gerentes de reputação dando uma certa garantia aos outros nós sobre as informações que ele distribui. Mesmo com esse tipo de sistema, ainda existe o problema de traição, onde um nó tem comportamento correto durante um período de tempo e, quando ganha bastante confiança, aplica um golpe prejudicando outro usuário.

Além da traição, existe outro ataque que tem uma eficácia elevada. Este ataque utiliza o conluio entre nós para prejudicar um usuário. Como exemplo, pode-se citar um grupo de 200 usuários que resolve prejudicar o nó X. Os duzentos usuários escolhem um nodo Y que será o nó em que eles irão assegurar confiança provendo falso testemunho ele dizendo que Y é um nó correto. Quando X fizer uma transação com Y ele irá considerar que Y é um nó correto pela quantidade elevada de recomendações, porém Y é um nó malicioso do grupo e irá prejudicar X.

O ataque de Sybil pode ser usado juntamente com esse citado no parágrafo anterior de forma a não existir a necessidade de vários nós entrarem em conluio. Para isso, um único nó aplicando Sybil seria capaz de criar todas as recomendações falsas e elevar falsamente a confiabilidade de um nó. Assim, mesmo utilizando um gerente de reputação, é complicado o gerente garantir quais mensagens são verdadeiras e quais são forjadas (conluio).

Ao contrário dos ataques citados anteriormente, que aumentam falsamente a reputação de um nó, um ataque pode visar prejudicar a reputação de um nó correto no sistema para que este fique incomunicável ou seja ignorado. Esse ataque pode ser realizado tanto por um conluio quanto por um atacante usando Sybil. O ataque consiste em informar, falsamente, para outros nós que um nó em questão (atacado) não é confiável anulando a confiabilidade dele perante os outros nós.

Como pode ser visto, existe um grande desafio a ser vencido na área de reputação de entidades. As várias propostas da literatura resolvem apenas alguns pontos isolados dos problemas ainda deixando brechas para atacantes. A P2PSL não fornece mecanismos explícitos para lidar com reputação, logo aplicações desenvolvidas utilizando-a poderão sofrer os ataques descritos aqui a menos que implementem mecanismos próprios para controlar esses requisitos.

### **5.3.5 Autorização**

Quando deseja-se criar sistemas que são mais que simples redes de compartilhamento de arquivos sem nenhum controle é necessário empregarmos mecanismos que permitam estabelecer controle de acesso aos recursos [34]. Sistemas de controle de acesso provém uma forma de controlar quais entidades da rede terão acesso a quais recursos e em qual momento. Assim, recursos não ficarão disponíveis a qualquer entidade presente na rede.

A natureza autônoma das redes P2P fazem com que sistemas que usem listas globais de controle de acesso centralizadas em servidores não sejam factíveis, pois cada nó precisa informar quais são as suas restrições para acessar os recursos que ele disponibiliza [37]. Além de listas de controle individuais, os nós precisam de políticas diferentes das tradicionais baseadas em sistemas fortemente acoplados, pois a comunicação em sistemas P2P baseia-se em uma interação com muitos nós não conhecidos ou anônimos.

Outro ponto importante é encorajar os participantes da rede a utilizá-la, pois políticas de controle de acesso diminuirão a disponibilidade de recursos ou arquivos na rede de maneira a reduzir a possibilidade de encontrar o que se deseja. É importante que o

controle de acesso enderece esse problema de maneira a incentivar os participantes a compartilharem recursos provendo um bônus que ele possa utilizar no futuro.

A P2PSL possui um módulo baseado em RBAC para políticas de controle de acesso. Na política se define quais nós desempenham quais papéis na rede e cada papel terá associado os recursos que serão liberados e bloqueados. A política de acesso atual é baseada no nome dos nós, por isso pode ser atacada se um nó obtiver um dos nomes contido na lista de acesso. Na P2PSL é simples obter um nome na rede, pois é a própria implementação que defini o nome. Dessa forma, um nó pode acessar qualquer recurso de outro nó se souber o nome de nó que tenha esse recurso liberado. O módulo RBAC não trabalha em conjunto com módulos de autenticidade para evitar isso.

### 5.3.6 Integridade

As aplicações mais comuns no ambiente P2P são as de compartilhamento de arquivos e por esse motivo é importante que exista um meio de garantir a integridade dos arquivos de maneira que usuários não obtenham conteúdo corrompido. A adulteração do conteúdo pode ocorrer tanto no seu armazenamento quanto no trânsito do mesmo pela rede.

Para assegurar que um conteúdo não seja modificado no trânsito pela rede basta utilizar o módulo de autenticidade, pois caso o conteúdo seja modificado a assinatura não irá corresponder ao conteúdo e isso invalidará o dado. A garantia ocorre, pois o módulo de autenticidade (assinatura digital) aplica o algoritmo de criptografia sobre o *hash* do conteúdo. Dessa forma, se o conteúdo for alterado, seu *hash* não será o mesmo afetando o valor da assinatura.

Em muitos sistemas de compartilhamento de arquivos, uma requisição é respondida por várias fontes aumentando a disponibilidade, dessa maneira o nó requisitor irá contactar um subgrupo dos nós que detém o conteúdo que ele deseja. Caso haja algum nó malicioso no caminho entre o nó requisitor e algum dos nós entregando o conteúdo alterando o conteúdo, o nó requisitor pode solicitar o mesmo conteúdo de outro dos nós que o detém.

### 5.3.7 Anonimidade e Negabilidade

Segundo [38], a anonimidade de um sistema deve cobrir os seguintes aspectos:

- **Anonimidade do Autor:** Impede um atacante de descobrir quem é o autor de um documento;
- **Anonimidade do Publicador:** Impede um atacante de descobrir quem publicou o documento na rede;
- **Anonimidade do Leitor:** Impede que um atacante consiga identificar quais são os leitores de um documento de maneira a proteger sua privacidade;
- **Anonimidade do Servidor:** Dado um documento, impede que um atacante consiga identificar qual ou quais servidores na rede tem posse deste documento;
- **Anonimidade do Documento:** Esse aspecto também é conhecido como negabilidade, pois impede o servidor de identificar o conteúdo dos documentos que ele armazena;
- **Anonimidade de Consulta:** Impede que um servidor consiga identificar o tipo de consulta que um cliente está fazendo. Da mesma forma, impede que o servidor identifique os dados que está enviando para o cliente em uma resposta.

A anonimidade tem a intenção de manter a privacidade dos usuários da rede e, juntamente com isso, evitar censura e problemas legais devido a conteúdo que possa ser considerado legalmente ofensivo em alguns países. Um exemplo de problema legal foi o caso Napster [39] em que a empresa que centralizava o servidor para permitir aos usuários troca de músicas no formato MP3 foi processada e teve que mudar seu modelo de negócio e controlar o que seus usuários estivessem distribuindo.

Como exemplo de censura pode-se citar a proibição ao acesso e distribuição a certos documentos, vídeos e documentários. Um exemplo disso no Brasil é o documentário “Muito Além do Cidadão Kane” e na China pode-se citar uma grande gama de informação que é considerada proibida sendo que o governo chinês possuiu um acordo com a Cisco para controlar o tráfego de Internet no país.

A P2PSL não oferece suporte para mecanismos de anonimidade. Atualmente é necessário implementar soluções para isso totalmente na camada da aplicação e por esse motivo é possível que um atacante obtenha informações sobre quem está fazendo o que em um sistema utilizando como único mecanismo de segurança a P2PSL.

### **5.3.8 Poluição de Conteúdo**

Esse assunto pode ser considerado juntamente com a integridade de arquivos, pois visa fazer com que o conteúdo chegue corrompido para um usuário. A poluição de conteúdo tem um grande apelo comercial, pois entidades como a indústria fonográfica estão tentando barrar a pirataria de músicas. Existem entidades que oferecem serviços de poluição como o Overpeer (<http://www.overpeer.com>).

Basicamente, os motivos que levam a inserção de conteúdo poluído na rede podem ser divididos em três: usuários maliciosos com intuito apenas de prejudicar a rede; usuários com intenção de espalhar vírus ou material próprio sob títulos de alta popularidade; indústria tentando proteger sua propriedade intelectual da disseminação sem os devidos pagamentos.

Existem certos tipos de poluição que não podem ser detectados pelo sistema, ficando a cargo do usuário a classificação manual do conteúdo como poluído. Dessa forma, para melhor funcionamento da rede, o sistema deve permitir e facilitar a classificação do conteúdo pelos usuários para que possam informar uns aos outros a situação de certo conteúdo ou nó como poluidor. Outra boa prática do ponto de vista do usuário é apagar imediatamente um conteúdo que veio poluído para não disseminá-lo entre os outros usuários e não ser taxado como poluidor.

A P2PSL não oferece meios para controlar a distribuição de conteúdo poluído. Da mesma forma, não é capaz de taxar usuários poluentes como maliciosos e fica a mercê desse tipo de ataque.