Beeapeer: Uma ferramenta para localização e monitoramento de material de abuso sexual infantil na rede Gnutella

Luciano Porto Barreto, Leandro Nunes dos Santos, Daniel Coelho Cunha

***Abstract*—Production and dissemination of sexually abused children content is unfortunately on the rise, especially in peer-to-peer networks. Despite the efforts of law enforcement and national security agencies, there is still a need for specialized tools to help analysts and legal actors in order to discover, gather information and trace such activities. This paper presents our experience in the development and evaluation of BeeaPeer - a tool for automatic discovery and logging of child abuse sexual content exchanged in the Gnutella network.**

Keywords*: *Cybercrime, child sexual abuse content, peer-to-peer networks, Gnutella.

1. Introdução

A produção e distribuição de material referente ao abuso sexual de crianças e adolescentes constitui prática criminosa anterior à popularização da Internet e à invenção dos computadores [1] apud [2]. Na atualidade, infelizmente, autores [2-4] compartilham a idéia de que a Internet está se transformando no principal meio de distribuição e comercialização desse material.

Inicialmente, a disseminação dessa sorte de material, por meios computacionais era realizada através dos *Bulletin Board Systems* (BBS) e grupos de notícias da Usenet. Mais recentemente, estão sendo utilizados canais do Internet Relay Chat (IRC), Web sites da World Wide Web, mensagens de correio eletrônico, bem como a troca direta de arquivos possibilitada pelas redes sociais (*e.g.*, Orkut, Facebook) e redes ponto-a-ponto (*peer-to-peer*). Nesse particular, a forte adesão de usuários às redes sociais e o interesse por serviços de compartilhamento providos por redes ponto-a-ponto tem motivado a intensificação do monitoramento e do combate à distribuição deste tipo de conteúdo ofensivo por parte da sociedade organizada e governos.

Nos EUA, destacam-se a organização sem fins lucrativos NCMEC [5], a subdivisão *Criminal Division, Child Exploitation and Obscenity Section* do Departamento de Justiça e as unidades *U.S. Customs Service CyberSmuggling Center* e do *Serviço Secreto* [6] ligadas ao Departamento do Tesouro.

A polícia espanhola tem utilizado efetivamente o software *Híspalis*[7], desenvolvido por Albert Gabás, para identificação de conteúdo ofensivo em redes ponto-a-ponto. Por meio de sua utilização, a polícia localizou, por exemplo, um usuário que distribuía 132 imagens com cenas de abuso sexual infanto-juvenil [7].

Desde 2008, agências nacionais francesas e organizações não-governamentais participam do projeto *Measurement and Analysis of Peer-To-Peer Activity Against Paedophile Content* [8], cujo objetivo primordial consiste no auxílio à proteção de usuários das redes ponto-a-ponto, em particular, crianças. Os principais resultados residem nas análises derivadas dos estudos conduzidos e na implementação de ferramentas e bancos de dados especializados.

No Brasil, a Polícia Federal tem atuado com rigidez no combate aos crimes perpetrados nas redes *Peer-to-Peer*. Exemplos notórios de atividades policiais incluem as operações Carrossel I [9], Carrossel II [10] e Ossorico [11]. Esta última foi realizada em conjunto com a Polícia Federal da Espanha e as demais se valeram de dados obtidos através do software EspiaMule, o qual efetuava o armazenamento das informações necessárias às operações[12].

Este artigo tem por objetivo somar esforços ao combate à distribuição de material contendo indícios ou evidências de abusos de crianças e adolescentes na rede *Peer-to-Peer* Gnutella. Para tanto, foi desenvolvida a ferramenta *Beeapeer*que possibilita a identificação e a coleta da materialidade destes conteúdos ofensivos visando análise e investigação posteriores. Nosso intuito é colaborar com instituições governamentais na descoberta, análise e fornecimento de informações precisas sobre a distribuição e armazenamento de conteúdo referente, principalmente, à exploração sexual infanto-juvenil.

O restante desse artigo está estruturado da seguinte forma. A seção 2 apresenta os conceitos fundamentais e o funcionamento básico acerca das redes e sistemas *Peer-to-Peer*. Em seguida, a seção 3 descreve o protocolo *Gnutella*, foco principal desse artigo. A seção 4 apresenta as decisões de projeto e detalhes sobre a implementação da ferramenta *Beeapeer*. Os resultados obtidos através da análise experimental dessa ferramenta são descritos na seção 5. Por fim, a seção 6 encerra o artigo destacando trabalhos futuros e tecendo considerações finais acerca do trabalho realizado.

2. Redes e sistemas Peer-to-Peer

Sistemas ou redes *Peer-to-Peer* podem ser definidos como sistemas distribuídos formados por nós interconectados capazes de se auto-organizarem em determinada topologia

de rede a fim de compartilhar recursos tais como conteúdo, capacidade de processamento, espaço de armazenamento e banda passante[13]. Tais sistemas são capazes de se adaptar dinamicamente a falhas, acomodar populações transitórias de nós mantendo níveis de conectividade e desempenho aceitáveis, dispensando a intermediação ou suporte de servidor ou autoridade global centralizada.

As redes ponto-a-ponto podem ser classificadas quanto a sua estrutura e forma de centralização. Em termos de estrutura, ou forma de criação, as redes são caracterizadas em estruturadas e desestruturadas. Na primeira, a topologia da rede é estritamente controlada e seus arquivos são organizados de maneira específica, o que facilita a localização de arquivos resultantes de buscas na rede [14]. Exemplos de tais redes incluem as redes Freenet [15] e Kademlia [16]. As redes desestruturadas, por sua vez, caracterizam-se pela ausência de controle sobre a topologia ou localização de arquivos. Nesse modelo, os nós ingressam na rede sem o atendimento à regras rígidas, o que requer localizar os arquivos sem a utilização de qualquer conhecimento prévio sobre a organização da rede.

No que concerne ao aspecto de centralização, as redes podem ser divididas em [13]:

Puramente descentralizadas: Nestas redes inexiste coordenação central das atividades na rede. Assim, os nós atuam tanto como clientes quanto servidores, executando as mesmas tarefas. Free Haven[17] representa um exemplo de rede dessa categoria.

Parcialmente centralizadas: Estas redes diferenciam-se das redes puramente descentralizadas por possuírem o conceito de supernós; nós eleitos dinamicamente por disporem largura de banda e poder de processamento suficiente para servirem um pequeno subconjunto de tarefas da rede. A rede Gnutella pode ser incluída neste rol.

Híbridas: Nas arquiteturas de redes híbridas existe uma entidade central que provê alguns serviços para facilitar a interação entre os nós. Os servidores geralmente mantém uma estrutura de índices com metadados sobre os arquivos compartilhados por cada usuário. Napster [18], BitTorrent [19] e eDonkey [20] são redes fundamentadas nesse estilo de organização.

A rede Gnutella, objeto de estudo deste trabalho, nas suas versões anteriores era classificada como uma rede desestruturada e puramente descentralizada. Em sua versão atual, é considerada como uma rede parcialmente centralizada devido à utilização de supernós, denominados de *Ultrapeers*.

3. A rede e o protocolo Gnutella

O elemento central da rede Gnutella consiste no protocolo de comunicação subjacente, descrito resumidamente nessa seção. As funcionalidades do protocolo Gnutella [21] podem ser sintetizadas na busca por nós ativos, no estabelecimento de conexões e troca de mensagens com os nós ativos. A troca de mensagens envolve, essencialmente, a descoberta de outros nós conectados, e a procura de documentos na rede e suas respostas correspondentes.

3.1. Características

A rede Gnutella é qualificada como parcialmente centralizada e desestruturada, ou seja, inexiste coordenação central para os serviços oferecidos. As conexões são feitas diretamente entre os nós, os quais assumem ambos os papéis de cliente e servidor. Quando atuam como clientes, os nós conectam-se a outros nós e podem efetuar buscas e *downloads* de arquivos. No papel de servidores, os nós aceitam conexões, pesquisam sua base de dados local, respondem às buscas recebidas e perfazem *uploads* de arquivos. Por tais razões, os nós são igualmente conhecidos por *servents* - contração dos termos *servers* e *clients*.

No intuito de aprimorar a escalabilidade da rede, o protocolo emprega um sistema de supernós, no qual os nós funcionam em dois modos: *Leaf* e *Ultrapeer*. Um nó em modo *Leaf* mantém apenas algumas conexões com nós no modo *Ultrapeer*. Estes últimos atuam como intermediários para nós *Leaf* conectados, possibilitando uma redução na quantidade de nós envolvidos no roteamento e troca de mensagens.

3.2. Bootstrapping

O ingresso na rede Gnutella inicia-se com a descoberta de um IP pertencente a um nó ativo na rede. Esta etapa, conhecida por *Bootstrapping* (inicialização), é de importância significativa, visto que os nós vizinhos determinam a localização do novo *servent* na topologia da rede e, além disso, influem diretamente no desempenho das pesquisas e transferências de arquivos.

Segundo [21], há quatro maneiras para obtenção dos endereços de outros nós; sendo a principal delas através da requisição a um *Gnutella Web Cache* ou *GWebCache*. Um *GWebCache* provê um ponto de partida para os nós que desejam adentrar a rede fornecendo *a)* endereços IP's de outros clientes que estão na rede Gnutella e *b)* URL's de outros caches. Cabe salientar que a descrição do funcionamento desse serviço não integra o protocolo Gnutella.

3.3. Handshaking

A entrada na rede Gnutella requer o estabelecimento de uma conexão com, ao menos, um outro *servent* previamente conectado. O estabelecimento da conexão dá-se através de troca de mensagens entre os *servents*; etapa denominada de *Handshake*. Após a obtenção do endereço IP e número da porta de um nó, uma conexão TCP/IP é criada e o *Handshake* é iniciado. Um *Handshake* bem sucedido envolve a troca de três mensagens entre os nós. No decorrer do processo, os dois *servents* participantes acordam sobre a versão do protocolo e recursos a serem utilizados, por exemplo.

3.4. Mensagens do protocolo

Após a conclusão do *Handshake* e estabelecimento da conexão, a comunicação entre os *servents* ocorre mediante troca de mensagens. Cada mensagem consiste em um cabeçalho e um corpo. A Figura 1 ilustra o envio de mensagens de *Ping* e *Query* (explicadas a seguir) entre nós na rede.

O cabeçalho está presente em todas as mensagens e possui os seguintes campos:

- **Globally Unique ID (GUID)** - Identificador único da mensagem na rede. Necessário para rotear as mensagens de resposta de volta ao remetente;
- **Tipo do Payload** - Identifica o tipo da mensagem: *Ping*, *Pong*, *Bye*, *Push*, *Query* e *QueryHit*;
- **Time To Live (TTL)** - Representa o número de vezes que a mensagem será encaminhada na rede[1]. O valor é decrementado pelo *servent* antes da mensagem ser reenviada. Ao atingir o valor zero, a mensagem é retirada da rede. Seu valor máximo é 7;
- **Hops** - Contador do número de vezes (saltos) que a mensagem já foi encaminhada por um nó. Útil para saber por quantos nós a mensagem viajou até atingir o nó atual;

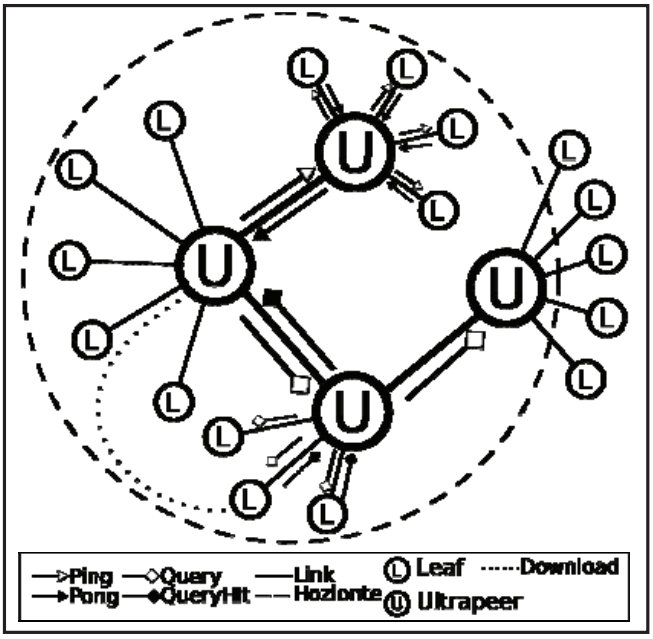

Figura 1: Comunicação na rede Gnutella. As mensagens Ping e Query são encaminhadas de nó a nó até o término de seu tempo de vida (TTL). Assim, as mensagens trafegam somente no escopo do horizonte do nó originário.

- **Tamanho do payload** - Determina o tamanho do corpo da mensagem, a qual segue imediamente ao campo de cabeçalho. O corpo da mensagem varia em função do seu tipo. O protocolo define seis tipos principais:

[1] Na rede Gnutella existe o conceito de *horizonte*. Cada nó enxerga somente até uma certa distância, pois suas mensagens tem alcance limitado na rede. Assim, um nó não dispõe da visão global da rede.

3.4.1. Ping

São encaminhadas de nó-a-nó até que o TTL atinja o valor zero. Estas mensagens não carregam informação útil. São utilizadas para verificar se o cliente está conectado e determinar o horizonte da rede.

3.4.2. Pong

Cada nó que recebe uma mensagem de *Ping* deve respondê-la com uma mensagem de *Pong*, a qual contém informações adicionais sobre o nó. Os campos dessa mensagem são os seguintes:

- **Porta** - Número da porta na qual o *servent* aceita conexões;
- **IP** - Endereço IP do nó;
- **Quantidade de arquivos** - Número de arquivos compartilhados pelo nó;
- **Total em kilobytes** - Número em kilobytes de arquivos compartilhados;
- **Bloco** - Bloco opcional destinado a extensões do protocolo.

3.4.3. Bye

Esta mensagem (de uso opcional) indica que, em breve, o cliente deixará a rede.

3.4.4. Query

Tal qual as mensagens de *Ping*, as mensagens *Query* são repassadas de *servent*-a-*servent* até que seu TTL expire. Essa mensagem serve para localizar um arquivo específico nos nós conectados. Os campos dessa mensagem são:

- **Velocidade mínima** - Velocidade mínima de transmissão que os *servents* devem possuir para responder a esta mensagem. Entretanto, essa utilização está obsoleta. A semântica atual considera cada bit deste campo como um sinalizador (*flag*) que indica condições nas quais a *Query* deve ser respondida ou compatibilidade com extensões do protocolo;
- **Critério de busca** - É um campo de texto em codificação ASCII. As buscas podem ser realizadas por palavra-chave ou por *hash*. Inexiste especificação precisa da interpretação que os *servents* receptores dessa mensagem devem realizar.
- **Bloco GGEP** - Campo opcional destinado a extensões do protocolo.

3.4.5. QueryHit

As mensagens *QueryHit* são respostas às mensagens *Query*. Um nó envia uma mensagem de *QueryHit* se, de acordo com seu critério, considerar que possui um ou mais arquivos que satisfaçam à mensagem *Query* recebida. É importante frisar

que um nó pode fingir ter a guarda de determinado arquivo, pois o protocolo não dispõe de verificação específica a esse fim. De maneira similar às mensagens de *Pong*, as *QueryHit*'s são roteadas de volta ao *servent* que originou a *Query*. Os campos da mensagem *QueryHit* estão descritos a seguir:

- **Hits** - Quantidade de elementos listados no campo Resultados (descrito a seguir);
- **Porta** - Número da porta na qual o *servent* que respondeu a mensagem *Query* aceita requisições HTTP para o download de arquivos;
- **IP** - Endereço IP do nó;
- **Velocidade** - Velocidade de transmissão em kilobytes por segundo do nó;
- **Resultados** - Conjunto de respostas da mensagem *Query* correspondente. Os resultados contém o número de elementos indicados no campo Hits (a seguir). Cada elemento da resposta é representado dessa forma:
- **Índice** - Número que identifica unicamente o arquivo no nó remoto;
- **Tamanho** - O tamanho em bytes do arquivo no nó remoto;
- **Nome** - Nome do arquivo no nó remoto;
- **Extensões** - Campo opcional destinado a extensões;
- **Extended Query Hit Descriptor (EQHD)** - Campo opcional, porém recomendado. Possui os seguintes itens:
- **Nome da aplicação** - Quatro caracteres representando o nome da aplicação.
- **Tamanho do campo de dados** - Tamanho em bytes do campo de dados.
- **Campo de dados** - Contém duas *flags* de um byte cada. As *flags* contém bits que sinalizam: *a)* se o bloco de Dados é do tipo GGEP[2]; *b)* se o bloco Velocidade é uma média dos últimos uploads ou é definido pelo usuário; *c)* se o *servent* já realizou, com sucesso, o upload de pelo menos um arquivo; *d)* se o *servent* está ocupado ou pode responder a uma eventual requisição e *e)* se o *servent* está situado atrás de um *firewall* ou não pode aceitar requisições HTTP diretas.
- **Dados** - Área de dados específica de cada aplicação. Não documentada.
- **Identifier (ID)** - Cadeia de caracteres de 16 bytes que identifica unicamente o *servent* na rede Gnutella.

3.4.6. Push

Mecanismo do protocolo que permite aos clientes situados atrás de um *firewall* compartilhar seus arquivos.

[2] Gnutella Generic Extension Protocol, campo destinado a extensões do protocolo.

3.5. Transferência de arquivos

Em posse das mensagens de *QueryHit*, o *servent* pode iniciar o download dos arquivos. Vale ressaltar que as transferências não são feitas através da rede Gnutella. Para tanto, deve ser estabelecida uma conexão direta entre o nó que disponibiliza o arquivo e aquele que efetuará o download.

O protocolo utilizado para o download de arquivos é o HTTP. Com as informações adquiridas por meio das mensagens *QueryHit*, o *servent* envia uma requisição GET ao nó portador do arquivo, o qual, subseqüentemente, envia o arquivo.

Caso *servent* esteja atrás de um *firewall*, o arquivo pode ser enviado mediante o emprego das mensagens de *Push*. Para tal, um nó *D*, que fará o download, deve possuir uma porta acessível externamente, a qual será utilizada por um nó *U*, que fará o upload. Neste cenário, inicialmente, o nó *D* envia uma mensagem de *Push* que será roteada pela rede Gnutella até o nó de destino *U*. Ao recebê-la, o nó *U* abre uma conexão dedicada com *D* para efetuar a comunicação diretamente através do protocolo HTTP, por exemplo, sem necessidade de uso da rede Gnutella.

Esta abordagem funciona na maioria dos casos, pois se o nó *U* está conectado à rede Gnutella, este consegue enviar requisições através do *firewall*. Entretanto, devido às ações deste último, pode ser incapaz de recebê-las. Assim, o nó *U*, por conseguir enviar requisições não retidas pelo *firewall*, pode tentar estabelecer outro tipo conexão com *D*. Todavia, dependendo das limitações impostas pelo *firewall*, este método pode não funcionar em todas as circunstâncias.

3.6. Pseudo-anonimato na rede Gnutella

Dois fatores contribuem para que os usuários tenham nítida sensação de anonimato na rede Gnutella. A arquitetura parcialmente centralizada é um deles, frente à ausência de entidade central mantenedora de um índice de arquivos compartilhados pelos nós. O sistema de roteamento baseado nas mensagens é o segundo fator, pois torna quase impossível determinar a origem e destino de uma mensagem, já que as tabelas de roteamento são dinâmicas e estão armazenadas em diversos nós da rede.

Dito isto, uma forma adequada de associar um arquivo a um usuário é efetuar esse registro durante o upload ou download do arquivo, visto que o protocolo normalmente utilizado é o HTTP. Essa abordagem foi utilizada, por exemplo, pelo site Zeropaid.com's Wall of Shame [22], o qual continha endereços IP de alguns usuários que tentavam obter conteúdo com abuso sexual infanto-juvenil através da rede Gnutella. Este site utilizava um cliente que mantinha uma lista de arquivos com nomes referentes àquele tipo de conteúdo. Ao receber uma requisição para download de um destes arquivos,

o cliente modificado, por meio das informações obtidas da conexão HTTP, armazenava o IP do requisitante.

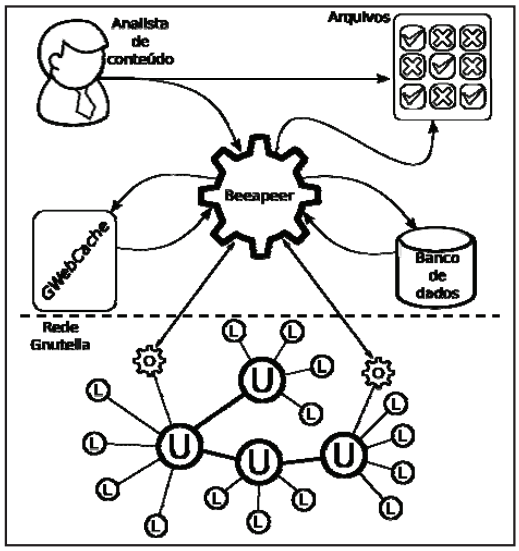

Figura 2: Visão geral da ferramenta BeeaPeer. O analista insere uma lista de palavras-chave ou de códigos de hash na base de dados. Durante a inicialização, o nó BeeaPeer efetua o Bootstrapping consultando um GWebCache. Em seguida, conecta-se à rede Gnutella inserindo vários nós para dispor de um horizonte abrangente. Os nós BeeaPeer comunicam-se com a rede enviando mensagens de Ping, Pong, Query e Push e respondendo às mensagens de Ping, Pong e QueryHit. Para realizar download, requisições são realizadas com as informações obtidas nas mensagens de QueryHit. Ao final, o analista de conteúdo classifica os arquivos adquiridos e toma as providências cabíveis.

4. A Ferramenta BeeAPeer

BeeaPeer é uma ferramenta computacional com o propósito de identificar a distribuição de material contendo cenas de abuso sexual de crianças e adolescentes na rede Gnutella. Entretanto, cabe salientar que a utilização e o propósito dessa ferramenta se insere num escopo mais amplo. Em verdade, a BeeaPeer faz parte de um processo que compreende etapas de busca, análise e classificação dos arquivos suspeitos obtidos em redes ponto-a-ponto. Tal ferramenta compõe atualmente o rol de produtos tecnológicos desenvolvidos pela SaferNet Brasil [23], uma associação civil sem fins lucrativos, voltada essencialmente à defesa dos direitos humanos na Internet.

O objetivo principal da ferramenta BeeaPeer consiste em permitir a consulta e descoberta automatizada de arquivos na rede e o armazenamento dos dados obtidos nas trocas de mensagens com nós suspeitos. O estado atual da ferramenta requer o mínimo de interação humana na atividade de busca e monitoramento, além de não concretizar o compartilhamento de arquivos ilícitos. A Figura 2 apresenta uma visão geral e uma breve descrição do funcionamento da ferramenta no contexto da rede Gnutella.

A ferramenta está estruturada em diversos módulos. O módulo de banco de dados provê facilidades de acesso a um banco de dados relacional. O módulo de comunicação efetua a comunicação com os *GWebCaches* e os demais nós da rede Gnutella. Por fim, dois módulos realizam o download dos arquivos: um para a transferência comum de arquivos e outro para transferências utilizando o método de *Push*.

4.1. Bootstrapping e Handshake

A ferramenta possui uma lista com URL's de *GWebCache* e seleciona aleatoriamente um desses URL's para obter os endereços IP's necessários à realização do *Handshake*. É importante manter esta lista atualizada porque esta determina a quais nós o cliente estará conectado.

Após adquirir os endereços, é instanciado um objeto *GnuEvent* - que possui a lógica do protocolo Gnutella. Todas as instâncias, independentemente, estabelecem uma conexão com seu respectivo nó, iniciando o *Handshake* e, posteriormente, a troca de mensagens. A Figura 3 ilustra a efetivação destas etapas.

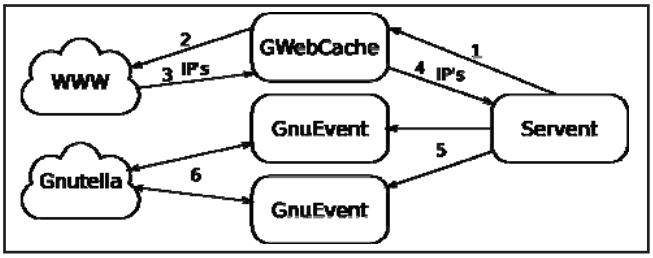

Figura 3: Início da interação com a rede Gnutella. O servent envia uma requisição ao GWebCache (1) para obter endereços IP de nós ativos na rede. O GWebCache faz essa requisição a um sistema web (2). Ao receber a resposta (3), ele a trata e envia uma lista de endereços IP ao servent. Este último instancia objetos GnuEvents (5) que efetuam a comunicação com os nós encontrados (6).

4.2 Troca de mensagens

As mensagens trocadas entre os nós dentro da rede Gnutella em formato binário, o que torna complexa sua criação e interpretação. Dessa forma, foi necessária a construção de uma classe específica para a codificação e decodificação das mensagens, denominada *GnuPack*. A interação entre objetos da classe *GnuPack* e da classe *GnuEvent* é apresentada na Figura 4.

Figura 4: Criação de mensagens pelo objeto GnuPack. Uma mensagem recebida da rede pelo objeto GnuEvent (1) é repassada e tratada pelo objeto GnuPack (2). Este decodifica a mensagem, a identifica como um Ping, e a devolve para o objeto GnuEvent (3). Em resposta, o objeto GnuEvent envia um Pong para a codificação (4), recebe o resultado (5) e, finalmente, envia a resposta (6).

4.2. Busca e transferência de arquivos

Para iniciar a busca de arquivos, a *GnuEvent* consulta a lista de palavras-chave e seleciona aleatoriamente uma destas. Em seguida, constrói a consulta (*Query*) com o auxílio da *GnuPack* e a envia para rede. Ao receber as respostas (*QueryHit*), é instanciado um objeto *Download* para cada arquivo listado no campo Resultados. O objeto *Download* lida com o protocolo HTTP realizando a requisição, tratando as respostas e obtendo os arquivos. Esse processo é reiniciado em um tempo preestabelecido. A Figura 5 apresenta um cenário-exemplo de busca e transferência de arquivos.

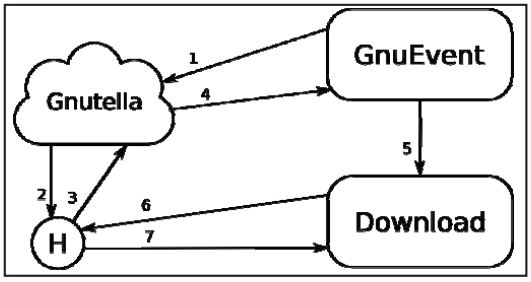

Figure 5: Busca e transferência de arquivos entre nós da rede. Uma Query é enviada à rede (1), a mensagem atinge um nó (2) que responde com uma QueryHit (3). A resposta é roteada de volta (4) e um objeto Download é criado (5), o qual envia uma requisição diretamente para nó (6). Por fim, o arquivo é transferido (7).

A dinamicidade da rede Gnutella (*e.g.*, entrada e saída de nós, mudança de topologia) é mascarada pela automatização no processo de busca, o qual reduz sobremaneira a necessidade de intervenção humana no uso da ferramenta. Por exemplo, é necessário que uma mesma busca seja realizada reiteradas vezes e em diferentes momentos de modo a inquerir uma maior gama de nós. De fato, a intervenção humana se faz necessária somente ao término dos *downloads*.

Um servidor associado a uma porta específica é mantido ativo caso seja necessário responder a mensagens de *Push*. Denominado de *GnuPush*, o servidor aceita conexões e, ao identificar o nó que estabeleceu a conexão, inicia o *download* de arquivos através do canal criado. Este procedimento está ilustrado na Figura 6.

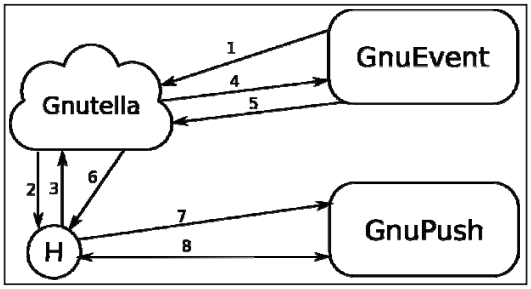

Figura 6: Transferência de arquivos por meio de Push. Uma Query é enviada à rede (1), a mensagem atinge um nó (2) que responde com uma QueryHit (3). A resposta é roteada de volta (4) e tratada por um objeto GnuEvent, o qual envia uma mensagem de Push a fim de obter o arquivo (5). A mensagem de Push é roteada até atingir o nó emissor da QueryHit (6). Este último estabelece uma conexão com o objeto GnuPush e se identifica (7). Através do canal criado, o objeto GnuPush requisita os arquivos (8).

4.3. Salvaguarda dos dados

As mensagens trocadas com nós pertencentes à rede Gnutella são armazenadas em um banco de dados relacional. Dessa forma, é possível efetuar análises sobre o horizonte da rede, as mensagens *Query* e *QueryHit*, e os downloads que podem ser efetuados. Para salvaguardar tais informações, uma interface de mapeamento foi utilizada para lidar com o sistema gerenciador de banco de dados. A Figura 8 exemplifica um dos fluxos de persistência da informação nesse contexto.

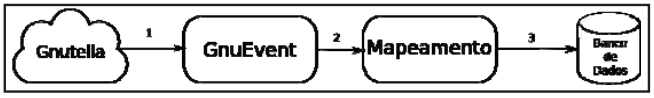

Figure 8: Persistência no banco de dados. A informação é captada da rede (1), os dados relevantes são enviados à interface de mapeamento (2) e salvaguardados no banco de dados (3).

4.4. Monitoramento de nós

Outra característica relevante da ferramenta consiste na possibilidade de monitoramento periódico de nós. Isto permite averiguar se um cliente determinado está distribuindo arquivos ofensivos. Tal funcionalidade é relevante devido à constante mudança da topologia da rede, o que dificulta reencontrar nós.

A ferramenta utiliza uma relação de endereços IP de interesse e, periodicamente, tenta realizar o *Handshake* com cada elemento da lista. Deste modo, em caso de sucesso no *Handshake*, os nós estarão diretamente ligados à ferramenta - dentro do seu horizonte - permitindo a realização de buscas automáticas em seus nós.

4.5. Limitações da implementação

Visando reduzir a complexidade e tempo de desenvolvimento do protótipo, o protocolo *Gnutella* não foi implementado integralmente. A saber, as funcionalidades não implementadas envolvem as operações para tratamento de *Query* recebidas e envio de mensagens de *QueryHit*, compartilhamento de arquivos (*i.e.*, *upload*) e as operações específicas do modo *Ultrapeer*. Em nossa implementação atual, o nó cliente atua apenas em modo *Leaf* e comunica-se na rede através de um subconjunto de mensagens. Entretanto, tais limitações não acarretam prejuízos às funcionalidades providas pela ferramenta.

5. Avaliação experimental

Esta seção apresenta os resultados obtidos com o uso da ferramenta BeeaPeer na busca e monitoramento da disseminação de arquivos ilícitos na rede Gnutella. Para tanto, os dados apresentados nessa seção referem-se à utilização da ferramenta durante cinco dias na rede Gnutella.

A busca por arquivos na rede Gnutella foi efetuada unicamente através de palavras-chave. Por não utilizar códigos de *hash*, como um meio para localizar arquivos

ilícitos, pode-se avaliar a facilidade na obtenção de tais conteúdos por um usuário comum. A simplicidade para encontrar arquivos com cenas de abuso sexual de crianças e adolescentes dá-se, principalmente, pela nomenclatura genérica e extensa atribuída aos arquivos - que possuíam entre 40 e 216 caracteres - e pelas palavras utilizadas. Grande parte das palavras-chave é de conhecimento popular (em inglês) no tange referências à pornografia adulta. Entretanto, outras palavras e termos foram identificados durante a pesquisa com o objetivo de realizar buscas mais precisas. Algumas palavras são jargões ou abreviaturas que formam um dialeto específico sobre conteúdo sexual infanto-juvenil. A Figura 8 apresenta as 150 palavras com maior número de ocorrência nos nomes arquivos classificados (letras maiores indicam palavras mais freqüentes).

Figure 8: Nuvem de rótulos das palavras-chave nos arquivos classificados

No decorrer da análise foram utilizados cinco conjuntos de palavras-chave. Um conjunto com palavras-chave compostas por duas ou mais palavras como, por exemplo, *child porn*. Outro conjunto com palavras-chave genéricas que podem, de alguma forma, fazer referência à pornografia infanto-juvenil. Um terceiro e quarto conjuntos de palavras-chave em português e em inglês, respectivamente. E um quinto conjunto com palavras-chave específicas que fazem referência direta ao tipo de conteúdo pesquisado, a exemplo de *pthc*. Destes, o conjunto de palavras-chaves em português foi o único que não retornou resultados.

De acordo com [24], a classificação de imagens contendo cenas com abuso sexual de crianças e adolescentes não é trivial por conta da variabilidade na sua definição face à legislação de cada país ou da perspectiva subjetiva que cada indivíduo possui sobre tema. Portanto, é necessária a definição de um método de análise com critérios os mais objetivos quanto possíveis, a fim de evitar distorções na categorização dos resultados. Em nossa abordagem de análise, os arquivos são classificados segundo a tipologia apresentada em [25], a qual considera cinco níveis de severidade apresentados na Tabela 1.

Nível	Descrição
1	Imagens retratando poses eróticas sem atividade sexual
2	Atividade sexual sem penetração entre crianças ou masturbação por parte da criança
3	Atividade sexual sem penetração entre adultos e crianças
4	Atividade sexual com penetração envolvendo criança ou crianças, ou ambos crianças e adultos
5	Sadismo ou penetração em ou por um animal

Tabela 1: Escala de severidade

Escolhido o método de categorização do conteúdo, efetuamos a conexão da ferramenta na rede Gnutella e realizamos buscas por palavras-chaves. A Tabela 2 apresenta a classificação dos 65 arquivos obtidos durante o monitoramento e validados pelo analista de conteúdo.

Nível	Quantidade
1	42
2	6
3	8
4	9
5	0

Tabela 2: Classificação dos arquivos coletados

A fim de identificar os países nos quais os distribuidores de conteúdo estavam localizados, utilizamos um banco de dados livre para geolocalização [26]. A Tabela 3 apresenta a distribuição de arquivos por país de origem [3] .

5.1 Discussão dos resultados

Um elemento de destaque nos resultados obtidos foi a grande quantidade de falsos positivos, ou seja, arquivos propostos para *download* que não continham relação com os critérios de busca enviados na *Query*. Esse é um problema comum do protocolo Gnutella devido à "inteligência" da rede ser distribuída pelos nós. Portanto, cada cliente tem liberdade

[3] Nos resultados, três arquivos foram excluídos porque foram coletados durante a fase de testes do protótipo, o qual ainda persistia no banco de dados as informações obtidas na rede.

para interpretar as buscas à sua maneira, o que pode resultar na distribuição de *spams* através da rede. Conseqüentemente, é preciso cautela na afirmação de que mensagens de *QueryHit* comprovam o compartilhamento do arquivo buscado. Isto somente pode ser atestado após o download e a análise comprobatória do arquivo pelo analista de conteúdo. O mesmo problema referente aos falsos positivos vale para buscas que utilizam códigos de *hash*.

Portanto, convém ser meticuloso na definição das palavras-chave que serão utilizadas na busca por arquivos. Idealmente, este conjunto de palavras deve ser reavaliado regularmente utilizando-se, por exemplo, técnicas de reconhecimento de padrões e aprendizado de máquina que considerem o nome, o formato e os metadados associados aos arquivos já classificados na base de dados. Assim, deve ser possível melhorar a eficiência das buscas, minorando a quantidade de falsos positivos. Além disso, isso possibilita adaptação contínua do software face aos novos padrões de utilização da rede, típicas das redes *Peer-to-Peer*.

O monitoramento de usuários previamente identificados como distribuidores de conteúdo abusivo, ainda que desejável, enfrenta diversos empecilhos. Em primeiro lugar, os endereços IP dos nós são dinâmicos, o que possibilita a um mesmo usuário entrar na rede com endereços IP diferentes e, conseqüentemente, restringir seu rastreamento pela ferramenta. Em segundo lugar, ainda que a ferramenta consiga localizá-lo, o nó suspeito precisa estar no modo *Ultrapeer*, caso contrário, ele não aceitará a conexão.

País	Quantidade
Estados Unidos	36
Alemanha	8
França	4
Canadá	3
Marrocos	2
México	2
Bélgica	1
Costa Rica	1
Eslovênia	1
Holanda	1
Hungria	1
Itália	1
Polônia	1

Table 3: Distribuição dos arquivos coletados por país

6 Considerações finais

Este artigo apresentou a ferramenta BeeaPeer, cujo objetivo consiste na descoberta e monitoramento de atividades relacionadas à disseminação de conteúdo de abuso sexual infanto-juvenil na rede Gnutella. A ferramenta possui diversas funcionalidades que automatizam o processo de busca e obtenção de arquivos, aliado à produção de registros (*logs*) de operações necessários à identificação dos nós suspeitos. A utilização prática da ferramenta nos permitiu verificar o atendimento aos requisitos necessários do contexto da aplicação e à obtenção concreta de arquivos e identificação de nós disseminadores de conteúdo referente à cenas de abuso sexual infanto-juvenil.

A ferramenta BeeaPeer pode ser estendida de diversas maneiras. É possível, por exemplo, sua adequação para funcionamento em outras redes ponto-a-ponto, a exemplo do que já foi realizado em por outras ferramentas tais como EspiaMule e Híspalis. Outras funcionalidades são igualmente desejáveis. Outra extensão importante consiste na melhoria da interface gráfica do protótipo a fim de facilitar a análise dos resultados, classificação dos arquivos e visualização das informações armazenadas no banco de dados. Além disso, seria útil a disponibilização de uma base de dados centralizada e segura que viabilizasse a utilização da ferramenta em diferentes sítios.

Referências

[1] S. H. Edwards, "Pretty babies: Art, erotica or kiddie porn? " History of Photography, vol. 18, no. 1, pp. 38–46, 1994.

[2] E. Quayle and M. Taylor, "Child pornography and the internet: Perpetuating a cycle of abuse," Deviant Behavior, vol. 23, no. 4, pp. 331–362, 2002.

[3] A. F. de Saint Maur, "The sexual abuse of children via the internet: A new challenge for interpol," International Conference Combating Child Pornography on the Internet, oct 1999.

[4] M. Taylor, "The nature and dimensions of child pornography on the internet," International Conference Combating Child Pornography on the Internet, oct 1999.

[5] "National center for missing and exploited children," 2010. [Online]. Available: http://www.missingkids.com

[6] File-sharing programs – Peer-to-Peer Networks Provide Ready Access to Child Pornography, United States General Accounting Office, Feb 2003.

[7] "Presentado el "buscador híspalis" dentro de la operación "azahar" contra la pornografía infantil," Guarda Civil - Oficina de relaciones informativas y sociales, Espanha, oct 2005. [Online]. Available: http://www.guardiacivil.org/prensa/notas/noticia.jsp? idnoticia=1828

[8] "Measurement and analysis of p2p activity against paedophile content national center for missing and exploited children," 2009. [Online]. Available: http://antipaedo.lip6.fr/

[9] "Operação Carrossel contribui para reprimir a pedofilia no mundo," Departamento de Polícia Federal - Divisão de Comunicação Social da Polícia Federal, Brasil, aug 2008. [Online]. Available: http://www.dpf.gov.br/DCS/noticias/2008/Setembro/08092008\s\do5(c)arrossel\s\do5(e)strangeiro.html

[10] "Operação Carrossel II combate pornografia infantil pela internet," Departamento de Polícia Federal - Divisão de Comunicação Social da Polícia Federal, Brasil, sep 2008. [Online]. Available: http://www.dpf.gov.br/DCS/noticias/2008/Setembro/03092008\s\do5(c)arrosselIIDF.html

[11] "PF deflagra operação Ossorico," Departamento de Polícia Federal - Divisão de Comunicação Social da Polícia Federal, Brasil, jan 2010. [Online]. Available: http://www.dpf.gov.br/DCS/noticias/2010/Janeiro/13012010\s\do5(O)pOssoricoDF.html

[12] J. R. S. de Oliveira and E. E. da Silva, "Espiamule e wyoming toolkit: Ferramentas de repressão à exploração sexual infanto-juvenil em redes peer-to-peer," in Proceedings of the Fourth International Conference of Forensic Computer Science – ICoFCS, ABEAT, 2009, pp. 108–113.

[13] S. Androutsellis-Theotokis and D. Spinellis, "A survey of peer-to-peer content distribution technologies," ACM Comput. Surv., vol. 36, no. 4, pp. 335–371, 2004.

[14] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim, "A survey and comparison of peer-to-peer overlay network schemes, IEEE Communications Surveys and Tutorials, vol. 7:2, march, pages 72-93, 2006.

[15] I. Clarke, O. Sandberg, M. Toseland, and V. Verendel, "Private communication through a network of trusted connections: The dark freenet," 2010, submitted to PET 2010. [Online]. Available: http://freenetproject.org/papers/freenet-0.7.5-paper.pdf
[16] P. Maymounkov and D. Mazières, "Kademlia: A peer-to-peer information system based on the xor metric," In Peer-to-Peer Systems. Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, October 2002, vol. 2429, ch. 5, pp. 53–65.
[17] R. Dingledine, M. J. Freedman, and D. Molnar, Free Haven. O'Reilly, 2001, ch. 12.
[18] "Napster," 2003. [Online]. Available: http://www.napster.com
[19] "Bittorrent," 2008. [Online]. Available: http://bittorrent.org/
[20] Y. Kulbak and D. Bickson, The eMule Protocol Specification, DANSS Distributed Algorithms, Networking and Secure Systems) Lab, School of Computer Science and Engineering, The Hebrew University of Jerusalem, jan 2005.
[21] Gnutella Protocol Specification v0.6, LimeWire, 23 jun. 2008. [Online]. Available: http://wiki.limewire.org/index.php? title=GDF
[22] G. Kan, Gnutella. O'Reilly, 2001, ch. 8.
[23] "SaferNet Brasil," 2010. [Online]. Available: http://www.safernet.org.br
[24] M. Taylor, E. Quayle, and G. Holland, "Child pornography: the internet and offending," The Canadian Journal of Policy Research (ISUMA), vol. 2, no. 2, pp. 94–100, 2001.
[25] "Sentencing guidelines council's definitive guidelines of the sexual offences act 2003," 2004. [Online]. Available: http://www.iwf.org.uk/police/page.105.htm
[26] "MaxMind's GeoIP," 2010. [Online]. Available: http://www.maxmind.com/app/ip-location

Luciano Porto Barreto Universidade Federal da Bahia, DCC/LaSiD, lportoba@ufba.br, Salvador, Bahia, Brazil

Leandro Nunes dos Santos SaferNet Brasil, leandronunes@safernet.org.br, Salvador, Bahia, Brazil

Daniel Coelho Cunha SaferNet Brasil, danielcunha@safernet.org.br, Salvador, Bahia, Brazil