Poluição de conteúdo em sistemas P2P Live Streaming

João Oliveira
Dep. of Computer Science
Fed. Univ. of Minas Gerais
holiver@dcc.ufmg.br

Alex Borges
Dep. of Computer Science
Fed. Univ. of Minas Gerais
borges@dcc.ufmg.br

Sérgio Campos
Dep. of Computer Science
Fed. Univ. of Minas Gerais
scampos@dcc.ufmg.br

ABSTRACT

Sistemas P2P de Live Streaming estão sujeitos a ataques de poluição de conteúdo. Essa tecnologia tem alto potencial para o consumo contínuo e estendido, no entanto, a degradação intencional pode se tornar fatal para a adesão e manutenção de usuários. Esse trabalho trata da caracterização do tráfego do SopCast exibindo comportamentos da rede com poluição e os impactos do ataque. Foi observado o efeito degradante de um nó que, em diversas amostras, foi capaz de comprometer mais de 30% da banda de *download* da rede e mais de 50% dos pares. Os resultados apontam a susceptabilidade do sistema a ataques com objetivo de degradação de qualidade.

Categories and Subject Descriptors

C.2 [**Computer-Communication Networks**]

General Terms

Caracterização, Poluição

Keywords

P2P Live Streaming, Poluição

1. INTRODUÇÃO

Redes P2P de distribuição de mídia contínua ao vivo, ou P2PTV, vêem se destacando ao longo dos últimos quatro anos pela sua ascensão e pelos números que movimenta sobre a Internet [7]. Apesar dos benefícios oferecidos pelo uso de parcerias, essa solução traz junto consigo deficiências, como a poluição de conteúdo, que, se não tratada corretamente, atrapalha a popularização do serviço.

Esse tipo de ataque degrada a disponibilidade de dados alterando maliciosamente o conteúdo original do sistema alvo [3, 9]. Sua ocorrência é representada pelo aparecimento de conteúdo estranho, ausência de conteúdo, propagandas ou pornografia no meio de uma transmissão normal, o que provavelmente induz os usuários comuns a mudarem de canal ou desligarem-se do sistema. Em caso de degradação prolongada ou conteúdo psicologicamente nocivo, como pornografia ou imagens chocantes, o consumidor pode inclusive alimentar uma aversão pelo sistema deixando de utilizá-lo por completo. Por isso, conhecendo essa ameaça, sua aplicabilidade e seu poder destrutivo, é importante gerar soluções e estudos que impeçam a sua propagação.

Em geral, os sistemas P2P são altamente sujeitos a poluição por uma característica essencial, os clientes participam ativamente do sistema. Como eles são, individualmente, pequenos servidores de conteúdo, podem modificar as requisições respondidas sem que o sistema na sua totalidade reconheça a ação maliciosa. Isso ocorre não só com distribuição de mídia, mas em qualquer tipo de dado, basta que seja conhecida a forma com a qual os dados são manipulados na comunicação, o seu protocolo.

De fato, atualmente, é menos comum encontrar ataques de poluição de conteúdo nas redes de mídia contínua que em redes de compartilhamento de arquivo, especialmente por conta do atributo comercial dos grandes sistemas de Live Streaming que não disponibilizam códigos-fonte ou especificações do protocolo. Contudo com a popularização da tecnologia e um número ascendente de usuários espera-se concomitantemente um crescimento do interesse e do empenho de usuários maliciosos a fim de descobrir meios de adulterar os sistemas para atender suas necessidades.

Esse trabalho realiza um estudo de caracterização e análise do tráfego em um dos mais famosos e utilizados ambiente real de distribuição de mídias ao vivo par-a-par, o SopCast[1], focando ataques de poluição de conteúdo. Os objetivos são descrever brevemente metodologias de ataque e, principalmente, ao realizar experimentos num canal fechado, encontrar relações entre o tempo de permanência de poluidores, o tempo de degradação e recuperação da rede, alcance da poluição, através de medições das taxas de *download*, *upload*, quantidade de parcerias a fim de auxiliar na proposição de alternativas que eliminem ou diminuam seus efeitos.

Nesse artigo serão abordados primeiramente a poluição de conteúdo e seus aspectos e em seguida apresentados os trabalhos relacionados. Na seção 4 será tratado a configuração dos experimentos, na seção 5 discutidos os resultados obtidos da caracterização e por último serão exploradas as conclusões e trabalhos futuros.

2. POLUIÇÃO DE CONTEÚDO

[1]http://www.sopcast.com/

WebMedia'09, October 5–7, 2009, Fortaleza, CE, Brazil.

Para o poluidor de conteúdo, atacar um sistema P2P Live Streaming equivale a manipular a informação que será enviada aos seus parceiros. Dentro dessa premissa um agressor pode ser bastante criativo, existe mais de uma forma de obter êxito em tais ataques e também mais de um objetivo ao realizá-lo, ainda, alvos da modificação podem ser *chunks* ou pacotes de rede. Poluição que tem como meta estragar o fluxo de dado pode ser feita alterando o cabeçalho dos pacotes enviados aos demais pares na parte do protocolo de aplicação, inserindo novos pacotes com cabeçalho adequado mas sem conteúdo válido ou até mesmo alterando um *byte* da parte de dados de um bom pacote, fazendo com que os pares recebam dados antigos, repetidos, inexistentes, corrompidos ou inválidos. Da mesma forma que é possível alterar um único *byte* para danificar o conteúdo codificado da mídia, torna-se factível alterar todos os dados de um pacote transmitindo assim um *stream* completamente diferente que, se integralmente baixado, será reproduzido no próximo nó como uma mídia normal. Outra maneira de pensar esse mesmo ataque envolve alterar os *chunks* da mídia enquanto eles estão armazenados no *buffer* do poluidor, ou seja, alterando-os na memória local antes de serem reenviados. Modificação de dados é a forma básica de diversos ataques e, apesar de ter objetivos diferentes, está presente em outros cenários como em jogos *online* através de ferramentas como MemHack[2] e Winsock Packet Editor[3].

A dificuldade na detecção de qualquer desses métodos é reflexo da exploração de situações inusitadas. Conhecer amplamente os usos impróprios dos sistemas é uma atividade não trivial, além de métodos novos serem criados sempre. A efetiva detecção perpassa o conhecimento prévio do que será recebido, fazê-la de maneira exata implica em ter o dado completo antes mesmo dele ser transmitido, o que seria impossível. No cenário de P2PTV, a complexidade das soluções é ainda maior, a parte de dados pode ter qualquer valor já que é um *stream* codificado. Ainda, a geração e consumo de conteúdo é em tempo real o que implica em fortes restrições aos requisitos de espaço e tempo de execução de algoritmos de segurança. Segundo [3, 4, 9], existem técnicas para a resolução do problema, indicando como mais aceito e correto o método de *chunk signing*, todavia, para funcionar ele deve estar imbutido no sistema.

A forma de ataque protagonizada nesse trabalho envolve a alteração de um conjunto de *bytes* no pacote correspondente ao que seria parte da mídia codificada. Pacotes enviado de tamanho superior a um dado limiar são considerados pacotes de dados e em uma região, supostamente contendo o *stream*, *bytes* são modificados de forma a assinar o pacote com uma sequência específica. Ainda, ao lado da sequência é colocado um identificador de criação garantindo que cada pacote poluído transmitido terá uma assinatura única composta por uma parte constante e um número. Para interceptar um pacote de rede que está saindo do poluidor utiliza-se uma característica do *framework* de filtragem de pacote presente no kernel do Linux 2.6.x, o Netfilter[4]. Ao aplicar uma regra capaz de filtrar todos os pacotes saindo da máquina local relativos ao SopCast e enviá-los a uma fila no espaço de usuário, uma aplicação poluidora pode concretizar as alterações e devolve-los à rede.

[2]http://www.memhack.com/
[3]http://www.wpepro.net/
[4]http://www.netfilter.org/

3. TRABALHOS RELACIONADOS

Poluição em redes de P2PTV é um tema recente e pouco abordado até onde se tem ciência. Caracterizações de cargas comuns de sistemas de Live Streaming podem ser encontradas em [7, 1, 9, 6, 5, 8, 3] e revelam traços importantes sobre a carga, no SopCast[1] por exemplo, o limiar de tamanho intuitivo entre pacotes de controle e de dados. Em [3], ataques de poluição ao PPLive são observados na ótica particular de dois pares da rede, um nó distante do poluidor e um ligado diretamente, o ataque é realizado de maneira que o poluidor cria pacotes falsos e de alta demanda numa taxa maior do que a própria taxa de apresentação da mídia o que força uma disseminação rápida. Já em [9] a poluição é modelada e simulada considerando características do AnySee, outro sistema P2P Live Streaming. Ainda, a questão é discutida e opções de solução são apresentadas em [3, 9, 2]. Esse trabalho se diferencia pela forma como realiza o ataque, ao procurar se aproximar do que seria uma tentativa real, na quantidade de nós observadores envolvida e nas análises relativas à rede como um todo.

4. EXPERIMENTO

Para realizar o experimento transmitiu-se em um canal privado do SopCast uma mídia codificada através do Windows Media Encoder 9 a uma taxa de aproximadamente 120kbps. Por canal privado entende-se que a transmissão só seria acessada por usuários que conhecessem seu número de identificação, ou seja, ele não é divulgado na lista normal de distribuição de canais do software e, portanto, supõe-se que não houve nenhum usuário real assistindo a mídia, somente *crawlers* ou *bots*. Essa decisão teve como base não causar nenhum prejuízo real ao sistema dada a independência entre canais, além de não ser o foco do trabalho identificar o comportamento de usuários reais sob ataque. Ao optar por experimentos fechados imagina-se que a ausência de *churn* influencia os resultados. De fato, há o isolamento do fator *churn* externo, a saída e entrada de pares do canal, todavia, não é um fator crucial dado que o SopCast apresenta um alto *churn* interno, observado na forma de parcerias bastante promíscuas, e, dessa forma, a visão de cada par é semelhante para cenários fechados e abertos.

Para garantir a corretude e coerência dos resultados, o experimento foi repetido 22 vezes. A cada rodada de experimento foram conectados aproximadamente 400 pares no canal, cada um capturando o *log* sobre a atividade de rede específica do SopCast. Dentre eles um par é poluidor e tem comportamento agressivo, poluindo todos os pacotes que envia num dado intervalo de tempo.

5. RESULTADOS

O primeiro, mais simples e imprescindível resultado obtido é que apesar de existirem técnicas conhecidas de detecção na literatura suas utilizações pelo sistema não são claras. Em experimentos iniciais menores (com quatro pares), sendo um poluidor e um servidor foram identificados pacotes poluídos sendo trocados entre os outros dois. Entretanto, como dito anteriormente os experimentos principais foram feitos repetidas vezes e com um número elevado de nós. É importante salientar todavia que nem todas as capturas foram bem sucedidas. Nesse artigo observaremos os dados relativos a uma coleta representativa.

Uma característica importante desse tipo de rede e que foi

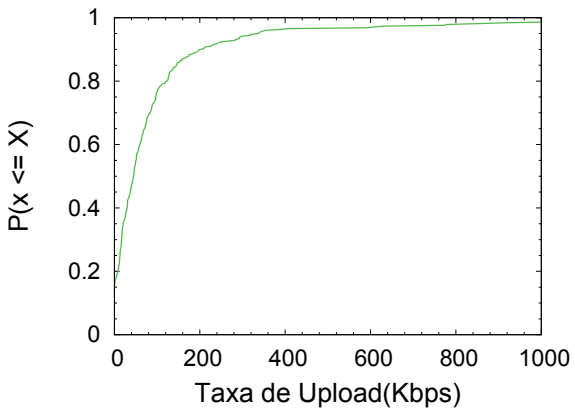

Figure 1: Distribuição da Taxa Média de Upload dos Pares

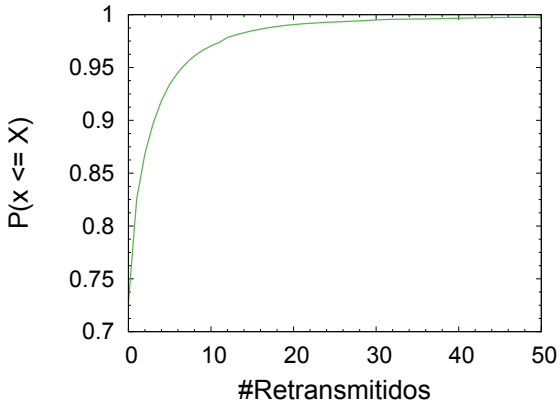

Figure 2: Distribuição da Quantidade de Retransmissões Poluídas por Pacote

observado durante os experimentos é que nem todos os nós contribuem da mesma forma para a transmissão. Na maioria dos experimentos entre 70 e 80% dos nós tiveram taxas médias de *upload* inferior a taxa original de transmissão do vídeo. Uma distribuição comum, obtida no cenário real, de taxas de *upload* média dos pares é apresentado na Figura 1. Somente em parte dos experimentos o poluidor teve uma taxa média de *upload* significativa, capaz de distribuir poluição suficiente para gerar os resultados aqui apresentados.

Nesse cenário, apresenta-se no gráfico da Figura 2 uma distribuição de retransmissões de pacotes poluídos, isto é, quantas vezes cada um dos pacotes modificados trafegou pela rede. Como descrito no sessão 2, cada qual tem identificação única, só é emitido uma vez pelo atacante. Dessa forma o gráfico revela o alcance e a capacidade de pervasão de um nó com alta oferta de *upload* e, por consequência, a possibilidade de impacto negativo do poluidor. Pode-se perceber que a maioria dos pacotes transmitidos por um nó atacante são de baixa demanda, em aproximadamente 97% das vezes eles foram retransmitidos pelos demais nós menos do que 15 vezes, enquanto um pequeno percentual é altamente pulverizado pela rede sendo retransmitido até 82 vezes.

Já na Figura 3 se evidencia um gráfico de *download* médio de uma rede típica sob ataque. O ataque é iniciado no instante t=100 e cessa em t=240 segundos. Como é possível observar a contaminação dos pares é praticamente instantânea, em poucos segundos após o início do ataque a rede está tomada de pacotes poluídos, o que demonstra a suscetibilidade do sistema. Acrescentando à análise, a descontaminação foi similarmente rápida, depois do fim do ataque a rede retransmite os últimos pacotes poluídos produzidos pelo atacante, durante as capturas a descontaminação completa foi praticamente instantânea na maioria das vezes. Acredita-se que o tempo máximo de descontaminação corresponde ao tamanho do *buffer* de vídeo do cliente SopCast.

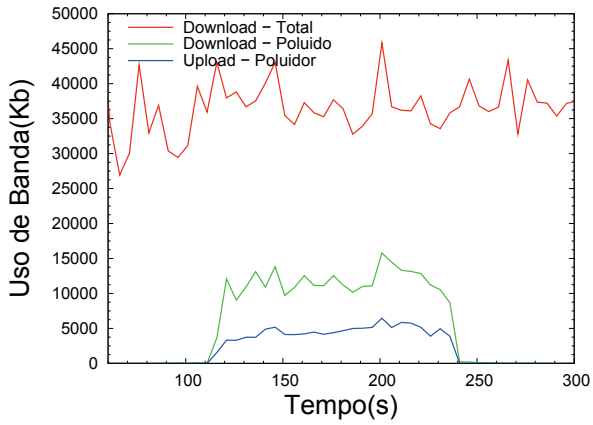

Figure 3: Uso de Banda da Rede

O percentual de contaminação da banda total consumida pela rede é também observado na mesma figura. Os dados mostram que a rede transmitiu a poluição numa taxa entre duas e três vezes maior que a taxa de geração, ou de *upload* do poluidor. De acordo os dados obtidos das capturas a taxa de *download* na rede não é influenciada pela quantidade de poluição inserida ou transmitida fortalecendo a conclusão de que os nós não solicitam a retransmissão desses pacotes. Ainda, em todos os experimentos a taxa de poluição foi praticamente a mesma, não dependeu do tamanho da rede ou da taxa total de *download*. Neles a poluição variou sua ocupação aproximadamente entre 30 e 70% do total transmitido no período do ataque. Coincidentemente, o grau de parcerias no poluidor foi similar em todas as capturas e aparentaram saturar em 50 nós. Podemos observar o poder de transmissão de um único nó, de forma análoga, lembrando que a diferença entre pacotes normais e poluídos é mínima e não influencia na difusão do vídeo.

Na Figura 4 temos a quantidade atual de sessões de parcerias do poluidor, assim como as parcerias com pares poluídos. As sessões de parceria são consideradas desfeitas quando se passa mais do que três minutos sem comunicação entre os pares. Um nó é considerado poluído quando recebe um pacote modificado e deveria ficar nesse estado até o pacote não servir mais para ser retransmitido. Por observação esse tempo foi assumido como 30 segundos, se nesse intervalo um novo pacote for recebido o temporizador é reiniciado. Ainda, em alguns experimentos notou-se que apesar do grau de poluição ser próximo de 50 pares houveram situações onde o poluidor mantinha mais parceiros e um alto *churn* nos pares atacados, ou seja, o conjunto de parceiros possíveis era maior que o conjunto instantâneo

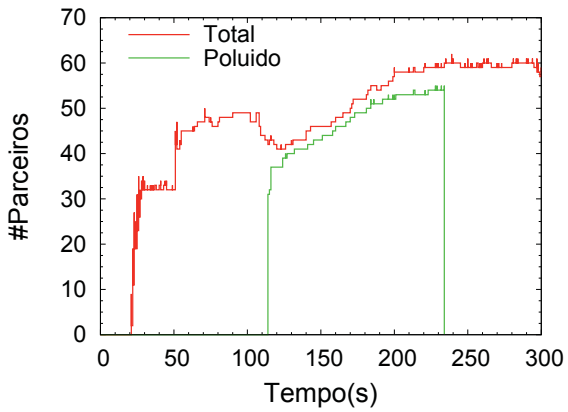

Figure 4: Parcerias do Poluidor

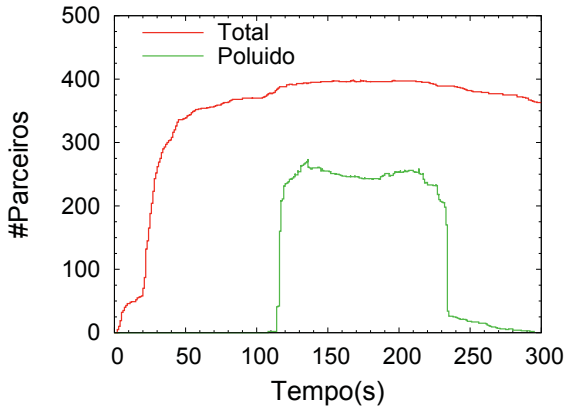

Figure 5: Parceiros na Rede

de poluídos que varia a cada instante apesar de manter um tamanho fixo. Essa observação foi possível graças aos limites temporais diferentes entre uma sessão de parceria e de poluição.

O percentual de contaminação de pares a partir da visão da rede variou entre aproximadamente 50 e 75%, enquanto o número de pares contaminados na rede ficou entre 3 e 5 vezes maior que o número de pares contaminados diretamente pelo atacante. Na Figura 5 mostra quantos nós da rede foram atingidos pelo mesmo ataque mostrado na figura anterior.

Analisando os resultados percebe-se que um ataque que visa a degradação da qualidade do sistema como um todo é possível de ser obtido a partir de um único nó poluidor visto que o percentual de contaminação de parceiros é bastante alto. Mesmo que a taxa de poluição seja menor que à taxa do vídeo, nesse caso, a chegada de alguns pacotes danificados resultaria num vídeo mal decodificado afetando a percepção do usuário. Ainda, supõe-se que essa degradação, mesmo que num conjunto pequeno de nós, geraria um efeito em cadeia, cada usuário descontente que saísse do sistema daria seu lugar a um usuário ainda não poluído.

6. CONCLUSÕES

Trata-se nesse trabalho da caracterização de uma rede P2PTV sob ataque de poluição de conteúdo com base em aspectos como quantidade e rotatividade de parceiros, taxa de *upload* e *download*, e pervasividade do tráfego de pacotes. Para isto, foram coletados, em dezenas de experimentos, *logs* de atividade de rede de todos os nós envolvidos na difusão de um canal fechado no SopCast.

Os estudos revelaram o comportamento e a influência de um nó com alta oferta de *upload* no papel de atacante. Em um canal fechado um nó por repetidas vezes foi capaz de comprometer mais de 30% da banda de *download* da rede com poluição e mais de 50% dos pares. Foram também traçadas relações entre a taxa de transmissão do poluidor, a pervasividade dos seus pacotes e a velocidade de contaminação e descontaminação com os impactos na totalidade do canal.

À continuidade desse trabalho espera-se realizar pesquisas referentes a variações nos parâmetros do ataque, em especial, no número de poluidores e na forma com a qual um grupo de poluidores pode influenciar uma parcela significativa ou a totalidade da rede com um ataque conjunto. Espera-se com isso gerar análises sobre ataques com objetivo de eclipsar o conteúdo original de vídeo. Ainda, pretende-se produzir um estudo mais específico sobre nós de alta demanda de *upload*, como identificá-los e quais fatores influenciam na eleição dos mesmos.

7. REFERENCES

[1] S. Ali, A. Mathur, and H. Zhang. Measurement of commercial peer-to-peer live video streaming. In *Proc. of Workshop in Recent Advances in Peer-to-Peer Streaming*, 2006.

[2] A. Borges, J. Almeida, and S. Campos. Fighting pollution in p2p live streaming systems. *Multimedia and Expo, 2008 IEEE International Conference on*, pages 481–484, 23 2008-April 26 2008.

[3] P. Dhungel, X. Hei, K. Ross, and N. Saxena. The pollution attack in P2P live video streaming: measurement results and defenses. In *Proceedings of the 2007 workshop on Peer-to-peer streaming and IP-TV*, pages 323–328. ACM New York, NY, USA, 2007.

[4] M. Haridasan and R. van Renesse. Defense against intrusion in a live streaming multicast system. In *Proceedings of the Sixth IEEE International Conference on Peer-to-Peer Computing (P2P'06)*, pages 185–192, 2006.

[5] X. Hei, C. Liang, J. Liang, Y. Liu, and K. Ross. A Measurement Study of a Large-Scale P2P IPTV System. *Multimedia, IEEE Transactions on*, 9(8):1672–1687, 2007.

[6] Y. Huang, T. Z. Fu, D.-M. Chiu, J. C. Lui, and C. Huang. Challenges, design and analysis of a large-scale p2p-vod system. *SIGCOMM Comput. Commun. Rev.*, 38(4):375–388, 2008.

[7] A. Sentinelli, G. Marfia, M. Gerla, S. Tewari, and L. Kleinrock. Will IPTV Ride the Peer-to-Peer Stream? *IEEE COMMUNICATIONS MAGAZINE*, 45(6):86, 2007.

[8] C. Wu, B. Li, and S. Zhao. Characterizing Peer-to-Peer Streaming Flows. *Selected Areas in Communications, IEEE Journal on*, 25(9):1612–1626, 2007.

[9] S. Yang, H. Jin, B. Li, X. Liao, H. Yao, and X. Tu. The Content Pollution in Peer-to-Peer Live Streaming Systems: Analysis and Implications. In *Parallel Processing, 2008. ICPP'08. 37th International Conference on*, pages 652–659, 2008.