The Producer-Consumer Collusion Attack in Content-Centric Networks

A. Nasserala and I. M. Moraes

***Abstract*— This paper evaluates a denial-of-service attack in information-centric networks based on the Content Centric Networking (CCN) architecture. This attack aims at increasing the content retrieval time. In this attack, both malicious consumers and producers collude, by generating, publishing, and changing content popularity. Malicious contents are stored by intermediate nodes and occupy the cache space that should be occupied by legitimate content. Thus, the probability of a legitimate consumer retrieves content directly from the producer increases as well as the content retrieval time. We evaluate the impact of the attack by varying the number of consumers and producers in collusion, the interest packets rate, and the way malicious contents are requested. Results show if 20% of consumers are malicious and send 500 interests/s each, the content retrieval time experienced by legitimate users increases by 20 times, which shows the effectiveness of the attack.**

***Keywords*— Future Internet, CCN, Security, Denial of Service.**

I. INTRODUÇÃO

AS REDES Orientadas a Conteúdo são um novo paradigma de comunicação para a Internet [3]. O principal objetivo dessas redes é a entrega de conteúdo para os usuários independentemente da localização desse conteúdo, ao contrário da arquitetura TCP/IP, cujo objetivo é a comunicação entre sistemas finais. Diversas arquiteturas foram propostas para esse novo paradigma de comunicação e uma das arquiteturas com maior destaque na literatura é a *Content Centric Networking* (CCN) [8,13]. Entre as principais características da CCN estão o roteamento através de nomes de conteúdo, o armazenamento de conteúdo em nós intermediários da rede e a capacidade de auto-certificar o conteúdo, aplicando a segurança diretamente aos pacotes de dados [8].

Uma das mais vantajosas características da CCN é o consumo indireto de conteúdo, ou seja, qualquer nó da rede que ao receber uma solicitação de conteúdo e possua esse conteúdo em *cache* pode enviar tal conteúdo para o nó solicitante. Na CCN, o nó que solicita o conteúdo é chamado de consumidor e o nó que disponibiliza o conteúdo é chamado de produtor. O produtor é a fonte de um conteúdo. Na CCN, é possível que um determinado nó, que esteja mais próximo do consumidor, consiga responder à solicitação de um conteúdo sem que o consumidor seja obrigado a recuperar esse conteúdo diretamente do produtor, que pode estar mais distante. Assim, o tempo de recuperação de conteúdos pode ser reduzido. Além disso, o armazenamento de conteúdo em *cache* aumenta a disponibilidade de conteúdos e pode reduzir o consumo de banda, uma vez que o conteúdo é encaminhado por menos saltos.

Outra característica da CCN é que a segurança é aplicada diretamente aos conteúdos, diferentemente da arquitetura TCP/IP, na qual a segurança é aplicada ao canal de comunicação entre os sistemas finais [13]. Um pacote de dados CCN é auto-certificado, isto é, ele contém a assinatura digital do pacote e a chave pública do produtor [8]. Portanto, é possível verificar a integridade do pacote e se ele foi gerado pelo produtor que possui tal chave pública. O uso de assinaturas gera sobrecarga tanto para o produtor assinar o conteúdo quanto para os consumidores verificarem a assinatura. Além disso, a CCN é mais robusta a ataques de negação de serviço (*Denial of Service* - DoS) comuns na Internet atual, como o de esgotamento de banda e o de reflexão, em virtude do uso de *cache* pelos nós intermediários e da agregação de solicitações de conteúdo [5], como será discutido na Seção II.

Um ataque de negação de serviço particular, chamado de conluio produtor-consumidor, entretanto, pode ser efetivo porque os mecanismos nativos empregados pela CCN não são suficientes para inibi-lo. Não foi encontrado na literatura uma avaliação do ataque no qual produtores e consumidores agem em conluio na CCN. Este trabalho avalia o impacto desse ataque em redes CCN, cujo objetivo é aumentar o tempo de recuperação de conteúdos. Nesse ataque, consumidores maliciosos solicitam conteúdos que são disponibilizados apenas por produtores maliciosos a uma alta taxa. Isso aumenta o tempo de recuperação de conteúdos legítimos, em virtude do aumento da taxa de erro do *cache* (*cache miss*) para esses conteúdos e, consequentemente, da necessidade de nós legítimos terem que recuperar o conteúdo diretamente do produtor. Os mecanismos de segurança da CCN padrão são ineficazes na detecção do ataque em conluio, pois, do ponto de vista da rede, as solicitações e os conteúdos são legítimos. São enviados pacotes de interesse para conteúdos que existem e que são disponibilizados por produtores. O conteúdo é malicioso porque torna popular um conteúdo que não é de interesse de usuários legítimos. Como o produtor malicioso assina os conteúdo de acordo com a política definida pela CCN, os consumidores maliciosos podem solicitá-los sem risco de que esses conteúdos sejam descartados por mecanismos de verificação de assinaturas e chaves. Esse ataque é possível, porque a CCN emprega políticas de substituição do *cache* baseadas, em sua maioria, na popularidade dos conteúdos. Assim, se um determinado conteúdo não é solicitado com frequência ou não foi solicitado

A. Nasserala, Universidade Federal do Acre (UFAC), Rio Branco, AC, Brasil, anasserala@ic.uff.br

I. M. Moraes, Universidade Federal Fluminense (UFF), Niterói, RJ, Brasil, igor@ic.uff.br

recentemente pelos consumidores, ele é considerado menos popular. Dessa forma, esse conteúdo terá prioridade de descarte quando houver necessidade de armazenar novos conteúdos. Vários nós maliciosos podem, então, solicitar um conjunto específico de conteúdos produzidos maliciosamente e em taxas altas de envio de interesse para manipular a política de *cache*. Assim, dependendo da forma como os conteúdos maliciosos são solicitados, é possível até remover conteúdos legítimos do *cache*.

A avaliação do ataque de conluio consumidor-produtor é feita através de simulações para diferentes configurações, nas quais se variam o número de consumidores e produtores em conluio, a taxa de pacotes de interesse e o padrão de solicitações de conteúdos maliciosos. As métricas empregadas são o tempo de recuperação de conteúdos legítimos, o percentual de ocupação maliciosa do *cache*, o percentual da taxa de erros de *cache* de conteúdos legítimos e o percentual de conteúdos legítimos recuperados do produtor. Os resultados mostram que o ataque compromete uma das maiores vantagens das CCN que é a redução do tempo de recuperação de conteúdos pelo uso do *cache* nos nós intermediários. Conclui-se que se 20% dos nós consumidores são maliciosos e enviam 500 solicitações de interesses por segundo cada um, o tempo de recuperação de conteúdos por usuários legítimos aumenta em cerca de 20 vezes na topologia avaliada. Além disso, observa-se que até 67% dos conteúdos legítimos são recuperados diretamente do produtor nas configurações analisadas.

O restante do artigo está organizado da seguinte forma. Na Seção II uma revisão sobre o funcionamento e aspectos de segurança da CCN é apresentada. Na Seção III os trabalhos relacionados são discutidos. Na Seção IV, o ataque de negação de serviço em conluio consumidor-produtor é descrito. Na Seção V, é definido o cenário de avaliação usado nas simulações. Na Seção VI, os resultados dos experimentos são analisados e discutidos. Por fim, na Seção VII, as conclusões são apresentadas.

II. A ARQUITETURA CCN: FUNCIONAMENTO E ASPECTOS DE SEGURANÇA

A arquitetura CCN tem como objetivos aumentar a disponibilidade e reduzir o tempo de recuperação de conteúdos. Na CCN, os nós da rede possuem um *cache* para armazenar conteúdos recebidos previamente. Consequentemente, qualquer nó pode responder a um pedido, se o conteúdo solicitado está disponível em seu *cache*, conhecido como *Content Store* (CS). Os consumidores são os nós que solicitam um conteúdo. Quanto mais nós armazenam um conteúdo na rede, maior a disponibilidade desse conteúdo e maior a probabilidade de consumidores recuperarem esse conteúdo de um nó mais próximo. Essa é uma das vantagens da CCN em comparação com a arquitetura atual da Internet.

A CCN emprega dois tipos de pacotes: interesse e dados. Consumidores enviam pacotes de interesse para solicitar um conteúdo. Os produtores ou qualquer outro nó que possua o conteúdo em seu CS respondem aos interesses com pacotes de dados, que carregam o conteúdo em si ou pedaços de conteúdo [4]. Os nós encaminham tanto pacotes de interesse quanto pacotes de dados com base no próprio nome do conteúdo, ao invés do endereço de destino do nó que possui o conteúdo. Para realizar o encaminhamento de pacotes, cada nó CCN tem duas estruturas de dados: a *Pending Interest Table* (PIT) e a *Forwarding Information Base* (FIB). A PIT guarda o estado de cada pacote de interesse encaminhado por um nó que ainda não recebeu uma resposta, ou seja, os interesses que esperam por um pacote de dados. Cada entrada da PIT também armazena a interface de recepção de um pacote de interesse. É importante ressaltar que o tamanho da PIT é limitado e, dessa forma, novos interesses que chegam enquanto a tabela está cheia não são encaminhados. Esse fato é explorado por usuários maliciosos, como detalhado nos próximos parágrafos.

A FIB atua como uma tabela de encaminhamento para pacotes de interesse. Essa tabela contém uma lista de entradas, cada uma contendo o prefixo de um nome e uma lista de interfaces de saída para as quais os pacotes de interesse com nomes de mesmo prefixo devem ser encaminhados.

Quando um nó CCN recebe um pacote de interesse, ele verifica seu CS para encontrar uma cópia do conteúdo solicitado, cujo nome está no cabeçalho do pacote de interesse. Se o conteúdo está armazenado em *cache*, o nó envia um pacote de dados para o consumidor. Caso contrário, o nó verifica a sua PIT. Se houver uma entrada na PIT para o mesmo conteúdo, o nó atualiza a lista de interfaces de entrada e descarta o pacote de interesse. Esse procedimento é chamado de agregação de pacotes de interesse e torna a CCN mais robusta contra ataques de DoS, como discutido a seguir. Caso contrário, o nó cria uma nova entrada na PIT e, então, consulta a FIB para determinar a interface de saída para encaminhar o pacote de interesse. Se não houver nenhuma entrada na FIB relacionada com o nome do conteúdo, o pacote de interesse é descartado. Os nós repetem este processo de encaminhamento para cada pacote de interesse recebido. Os pacotes de dados seguem o caminho reverso percorrido pelos pacotes de interesse porque a PIT armazena a lista de interfaces com interesses a serem atendidos [15].

Ataques de negação de serviço (DoS) são uma ameaça na Internet atual. A arquitetura CCN, entretanto, é mais robusta a esse tipo de ataque do que a pilha TCP/IP em virtude de duas características: o armazenamento de conteúdo pelos nós intermediários e a agregação de pacotes de interesse [6]. Ataques de esgotamento de banda e de reflexão, por exemplo, são pouco eficientes na CCN.

Ataques de esgotamento de banda inundam a vítima com requisições de serviço para esgotar seus recursos. Neste ataque, os pacotes devem chegar à vítima para que o ataque seja efetivo. Na CCN, no entanto, os pacotes não possuem o endereço de destino e os consumidores não podem garantir que os pacotes de interesse alcancem a origem do conteúdo, ou seja, o produtor e nesse a caso a vítima, porque qualquer nó pode responder ao interesse. Portanto, os consumidores maliciosos podem gerar uma quantidade enorme de pacotes de interesse para um dado conjunto de conteúdos, mas nenhum ou poucos desses pacotes alcançarão o produtor.

É importante ressaltar também que nós no caminho reverso entre o consumidor e o produtor armazenam conteúdos em *cache*. Assim, nós intermediários entre consumidor e produtor provavelmente irão satisfazer novos pacotes de interesse, que dificilmente alcançarão o produtor, dependendo da popularidade destes conteúdos. Além disso, a CCN reduz o número de pacotes de interesse transmitidos. Um nó só envia um pacote de interesse que não corresponde a uma entrada PIT. Caso contrário, o nó atualiza a lista de interfaces e descarta o pacote, como descrito nos parágrafos anteriores.

Ataques de reflexão são baseados na técnica de falsificação de endereços IP (IP *spoofing*) e visam atacar vítimas diferentes simultaneamente. Na CCN, esses ataques são menos eficazes porque os pacotes de dados são sempre encaminhados para o consumidor através do caminho reverso percorrido pelo pacote de interesse. Consumidores também não podem garantir que os pacotes de interesse cheguem às vítimas intermediárias ou finais devido ao *cache* nos nós intermediários. Nós CCN, porém, enviam pacotes de interesse em todas as suas interfaces, se não houver nenhuma entrada na FIB para um prefixo de nome solicitado. Portanto, se o atacante e a vítima estão na mesma sub-rede do ataque, a reflexão pode ser eficaz [6]. Neste cenário, o atacante pode enviar pacotes de interesse através de todas as suas interfaces com os endereços da camada MAC falsificados. Assim, múltiplas cópias do conteúdo são enviadas para a vítima. Para evitar isso, nós CCN não transmitem o mesmo conteúdo mais de uma vez no mesmo domínio de difusão (*broadcast*) [6].

Apesar de ser mais robusta do que a arquitetura TCP/IP aos ataques de DoS atuais, a arquitetura CCN possui ataques e vulnerabilidades identificados em trabalhos recentes [6,12], que são discutidos na Seção III.

III. TRABALHOS RELACIONADOS

Os ataques de negação de serviço em CCN são classificados em dois tipos: ataques por inundação de interesses ou envenenamento de *cache* [12].

O objetivo dos ataques de inundação de interesses é sobrecarregar a PIT com solicitações de conteúdo enviadas por um nó malicioso a uma alta taxa [7]. Os pacotes de interesse maliciosos, em geral, solicitam conteúdos inexistentes, o que mantém por mais tempo a informação sobre esses interesses na PIT de um nó. A informação sobre um interesse pendente só é removida após o estouro de um temporizador. Enquanto aguarda pelo pacote de dados, o nó receberá novos interesses para outros conteúdos inexistentes. No pior caso, com a PIT cheia, um nó afetado não atenderá interesses legítimos, o que leva à queda de desempenho da rede.

Gasti *et al.* [6] definem o ataque de inundação de interesses e propõem um mecanismo de *push-back* como contramedida. Esse mecanismo monitora a ocupação da PIT e identifica quando uma determinada interface está próxima de atingir seu número máximo de entradas na PIT. Assim, o mecanismo controla o fluxo de pacotes de interesse que contém os mesmos prefixos de nome. Além disso, a contramedida envia uma notificação na interface supostamente atacada que será recebida por um nó vizinho. Esse nó, por sua vez, deve propagar tal informação no sentido das interfaces atacadas e, ao mesmo tempo, limitar a taxa de interesses encaminhados que contenham o prefixo sob ataque. Portanto, o objetivo da contramedida é empurrar o ataque para o caminho de volta até o atacante, ou pelo menos para um nó no qual seja detectado [6]. A principal característica dessa contramedida é não modificar a arquitetura padrão proposta para a CCN. O ponto fraco do trabalho de Gasti *et al.* é que nem o impacto do ataque e nem a contramedida proposta são avaliados por simulação ou experimentos práticos.

Choi *et al.* [5], por outro lado, avaliam através de simulações a efetividade do ataque de inundação de interesses. Os autores mostram que em uma rede com poucos nós, o desempenho é comprometido. Conclui-se que a vazão de dados total de consumidores legítimos diminuiu cerca de 65%. Da mesma forma, observa-se que o tempo médio de recuperação de conteúdos aumenta rapidamente, logo após o início do ataque.

Afanasyev *et al.* [1] também avaliam o ataque de inundação de interesses através de simulações, porém consideram diferentes cenários e uma rede de maior escala do que a usada no trabalho anterior. Os autores também avaliam a contramedida baseada em um mecanismo de *push-back* proposta por Gasti *et al*. Os resultados mostram que essa contramedida é eficiente, pois isola por completo os atacantes de modo que eles causem pouco ou nenhum impacto no desempenho percebido por usuários legítimos.

Diferentemente dos ataques de inundação de interesses, o objetivo do ataque de envenenamento de *cache* é ocupar o *cache* dos nós com conteúdo poluído. Esse conteúdo é enviado por consumidores maliciosos para fazer com que nós armazenem um conteúdo que possua uma assinatura válida, porém corrompido ou aumentem a popularidade de conteúdos menos populares. No primeiro caso, o objetivo é reduzir o espaço disponível em *cache* para armazenar conteúdos legítimos e fazer com que consumidores recebam conteúdos corrompidos. No segundo, o objetivo é remover do *cache* conteúdos legítimos assumindo que uma política de substituição baseada na popularidade dos conteúdos é usada. Uma contramedida ao ataque de envenenamento de *cache* é a verificação da assinatura contida nos pacotes de dados [13]. Por padrão, a assinatura dos conteúdos é verificada apenas pelos nós de borda, ou seja, os consumidores, e não pelos nós intermediários da rede. Essa característica garante que os consumidores não recebam pacotes de dados contendo conteúdo malicioso. Nesse caso, o serviço da CCN é negado se os consumidores sempre receberem conteúdos inválidos. A solução de obrigar a verificação da assinatura de todos os conteúdos em todos os nós implica sobrecarga de processamento e, por isso, é de difícil adoção prática [6].

Ribeiro *et al.* [10,11] propõem um mecanismo de verificação probabilística de assinaturas. O mecanismo proposto é eficiente, porém se mostrou dependente da topologia de rede utilizada. Quanto maior o número de saltos, maior a probabilidade do conteúdo poluído ser descartado ao longo do caminho. Outra proposta similar, chamada

CacheShield [14], também usa dados estatísticos para verificar se um conteúdo é poluído ou não, porém tem as mesmas limitações do trabalho de Ribeiro *et al.*

Kim *et al.* [9] investigam o impacto de fluxos de conteúdo de longa duração na CCN. A presença de fluxos de longa duração pode ter efeito similar ao ataque de envenenamento de *cache*. Se fluxos de longa duração ocuparem temporariamente um *cache* de um nó por determinado conteúdo, eles podem expulsar pedaços de conteúdos populares do *cache*. Consequentemente, reduz-se a taxa de acertos do *cache* (*cache hit*). Os resultados das simulações mostram que há degradação da taxa de acertos do *cache* quanto maior é o número de fluxos de longa duração.

Todos os trabalhos descritos anteriormente que avaliam e/ou propõem contramedidas para os ataques de inundação de interesses e envenenamento de *cache* não consideram a possibilidade de ataques em que consumidores e produtores maliciosos agem em conluio para gerar, disponibilizar e manipular a popularidade de conteúdos. Avaliar tal ataque, detalhado na Seção IV, é o principal objetivo deste trabalho.

IV. ATAQUE DE NEGAÇÃO DE SERVIÇO POR CONLUIO CONSUMIDOR-PRODUTOR

No ataque de conluio existem pelo menos dois atores: o produtor malicioso e o consumidor malicioso, como ilustrados na Fig. 1. O produtor malicioso é responsável por produzir conteúdo malicioso conforme a demanda do consumidor malicioso. Esse conteúdo tem as mesmas características do conteúdo legítimo e, portanto, não terá tratamento diferenciado por nenhum nó CCN. Isso quer dizer que esse tipo de conteúdo ocupará o *cache* dos nós com o mesmo tratamento dos demais conteúdos que trafegam na rede. Os nomes dos conteúdos maliciosos também seguem as especificações da CCN. O consumidor malicioso, por sua vez, solicita conteúdos maliciosos em altas taxas. Na Fig. 1, os nós R1 e R2 armazenam conteúdos maliciosos em *cache*, uma vez que estão no caminho entre o consumidor e o produtor malicioso.

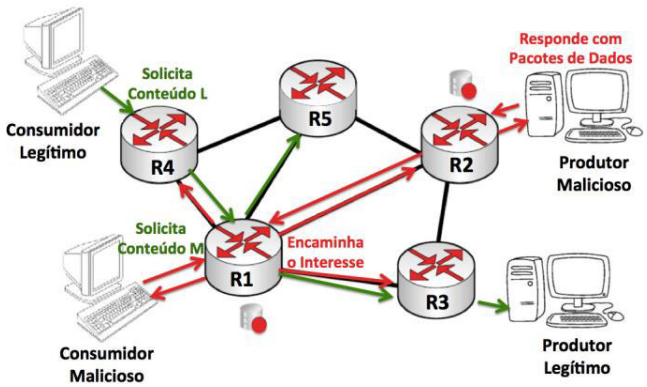


Figura 1. O ataque em conluio consumidor-produtor: nós legítimos e maliciosos em ação.

Com o ataque em conluio, o objetivo é prejudicar o consumo indireto de conteúdos, isto é, obrigar um consumidor legítimo a recuperar o conteúdo desejado diretamente do produtor. Esse objetivo é alcançado através da manipulação da popularidade dos conteúdos armazenados em *cache*.

Consumidores maliciosos enviam pacotes de interesse para um grupo de conteúdos que existem e que são respondidos pelo produtor malicioso. Assim, se solicitado com frequência, um conteúdo se torna popular, apesar de não ter sido solicitado por usuários legítimos. Por isso, o conteúdo é dito malicioso. Esse ataque é possível, porque a CCN emprega políticas de substituição do *cache* baseadas, em sua maioria, na popularidade dos conteúdos. Assim, se um determinado conteúdo não é solicitado com frequência ou não foi solicitado recentemente pelos consumidores, ele é considerado menos popular. Dessa forma, terá prioridade de descarte quando houver necessidade de armazenar novos conteúdos. Ao solicitar um conjunto específico de conteúdos e em taxas altas, os nós maliciosos manipulam a política de *cache*. Com mais conteúdos maliciosos em *cache*, maior a taxa de erro para os conteúdos legítimos e, consequentemente, a necessidade de nós legítimos terem que recuperar o conteúdo diretamente do seu produtor.

Mesmo que os consumidores legítimos não tenham que consumir diretamente do produtor, eles terão seus interesses encaminhados por mais saltos até conseguir o conteúdo desejado. No exemplo da Fig. 1, o consumidor legítimo pode ter que recuperar o conteúdo solicitado diretamente do produtor legítimo, uma vez que o *cache* do nó R1 que está no caminho entre os dois, pode estar sobrecarregado com conteúdo malicioso.

Uma das principais razões para que o ataque em conluio produtor-consumidor seja bem-sucedido é o fato de que os pacotes de interesse e de dados usados no ataque são legítimos para a rede e, portanto, não são detectados por mecanismos de verificação de assinaturas. O pacote com o conteúdo malicioso possui uma assinatura válida, carrega a chave do publicador e, assim, passa no teste de verificação de integridade e autenticidade. Logo, não é identificado como malicioso e nem descartado.

Outro objetivo do ataque em conluio é reduzir a eficiência da PIT, ao enviar pedidos de interesses para diferentes conteúdos maliciosos disponibilizados por produtores maliciosos a uma alta taxa. Dessa forma, é possível burlar o mecanismo de *push-back* proposto por Gasti *et al.* [6]. Esse mecanismo é eficiente contra a inundação de pacotes de interesse porque consegue identificar prefixos de nomes de conteúdo que frequentemente estão pendentes na PIT, uma vez que o conteúdo solicitado é inexistente.

Porém, se o consumidor e produtor estiverem agindo em conluio, os pacotes de interesse terão uma entrada na PIT de um nó somente até o conteúdo malicioso, que existe, retornar. Portanto, o mecanismo *push-back* não terá sucesso ao tentar identificar o ataque, pois os pacotes de interesse receberão uma resposta legítima e suas entradas serão removidas da PIT. Nesse caso, o ataque em conluio não provoca o esgotamento de recursos de armazenamento da PIT em um nó. O objetivo do ataque é gerar uma grande quantidade de pacotes de interesses, fazendo com que um nó tenha que manipular muitas solicitações de conteúdo maliciosas em detrimento a

interesses legítimos, o que pode levar a negação de serviço nesse nó.

V. CENÁRIO DE AVALIAÇÃO

A topologia da rede usada na avaliação do impacto do ataque em conluio é composta por 32 nós dispostos em forma de árvore, como mostra a Fig. 2. Os 24 nós folha são consumidores. O número de consumidores legítimos (CL) é fixo em todas as configurações e igual a 16. O número de consumidores maliciosos (CA) varia de 0 a 8. A posição dos CLs e dos CAs é definida aleatoriamente em cada rodada de simulação. O produtor legítimo (PL) é sempre o nó raiz. O produtor malicioso (PA) é o nó filho do nó raiz. Os demais 6 nós que compõem a topologia são os roteadores da rede (RTR). Os enlaces que interconectam os nós possuem taxa de transmissão de 100 Mb/s e atraso de 1 ms.

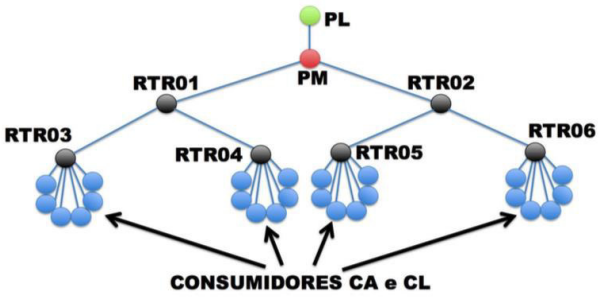


Figura 2. A topologia da rede usada nas simulações.

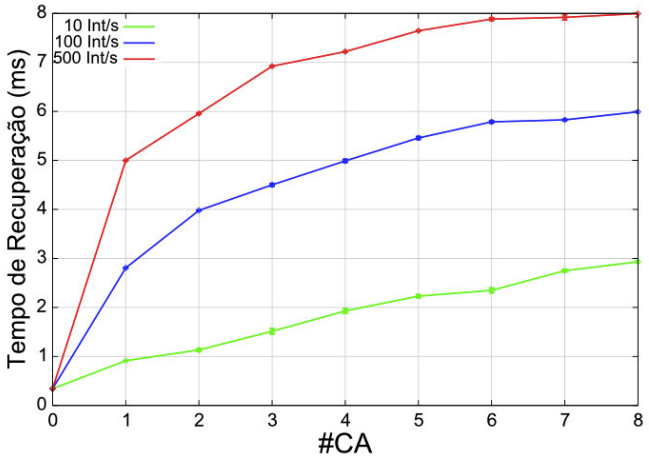
Os conteúdos são solicitados da seguinte forma. Os consumidores maliciosos enviam interesses para 12 conteúdos que são disponibilizados pelo único produtor malicioso a taxas de 10, 100 e 500 interesses por segundo. Cada conteúdo malicioso possui 100 pedaços (*chunks*) e prefixos de nome diferentes. Os pedaços de conteúdo são solicitados de duas formas diferentes: o consumo segundo a popularidade do conteúdo malicioso, seguindo uma distribuição *Zipf* com parâmetro α = 0,7[2], e o consumo sequencial, dito CBR (*Constant Bit Rate*), no qual um consumidor envia pacotes de interesse ordenados pelo nome do conteúdo e de forma cíclica. Os consumidores legítimos sempre enviam 10 interesses/s para outros 12 conteúdos disponibilizados pelo produtor legítimo. Cada conteúdo malicioso possui 100 pedaços (*chunks*) e prefixos de nome diferentes. Os pedaços são solicitados seguindo uma distribuição *Zipf* com parâmetro α = 0,7.

O *cache* dos consumidores legítimos e dos roteadores tem capacidade para armazenar até 1000 pedaços de conteúdo e cada pedaço possui 1024 bytes. Os consumidores maliciosos não possuem *cache* para potencializar o ataque, isto é, sempre enviam interesses independentemente se já receberam o conteúdo anteriormente ou não. A PIT tem tamanho ilimitado para que seja possível avaliar apenas o efeito do aumento da ocupação maliciosa no *cache* dos nós. A política de substituição de *cache* é a *Least Recently Used* (LRU).

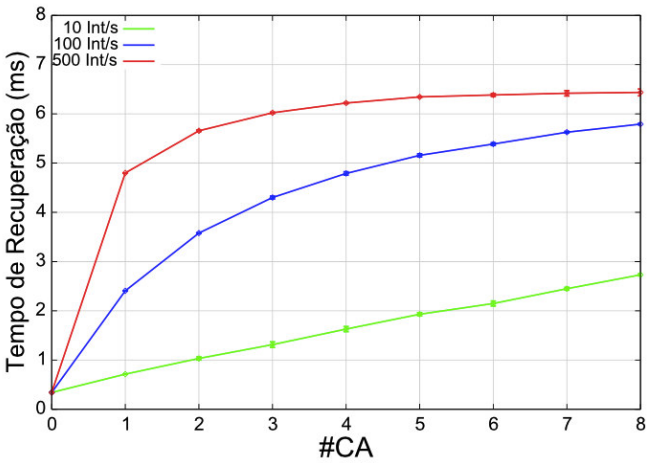
O módulo ndnSIM do simulador NS-3 é usado na avaliação. Para cada configuração, são realizadas 50 rodadas de simulação, cada uma com duração de 180 s. Para os pontos dos gráficos obtidos, são calculados intervalos de confiança representados por barras verticais para um nível de confiabilidade de 95%.

VI. RESULTADOS

Os resultados apresentados têm como objetivo avaliar o impacto do ataque de negação de serviço por conluio produtor-consumidor no desempenho da CCN. As métricas de desempenho são o tempo médio de recuperação de conteúdos legítimos, a ocupação maliciosa média do *cache* dos roteadores, a taxa média de erros de *cache* dos conteúdos legítimos e o percentual de conteúdos legítimos recuperados do produtor.



(a) CBR.

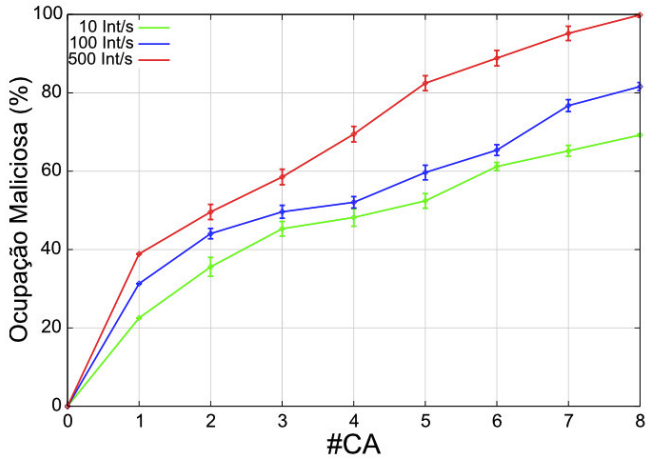


(b) Zipf.

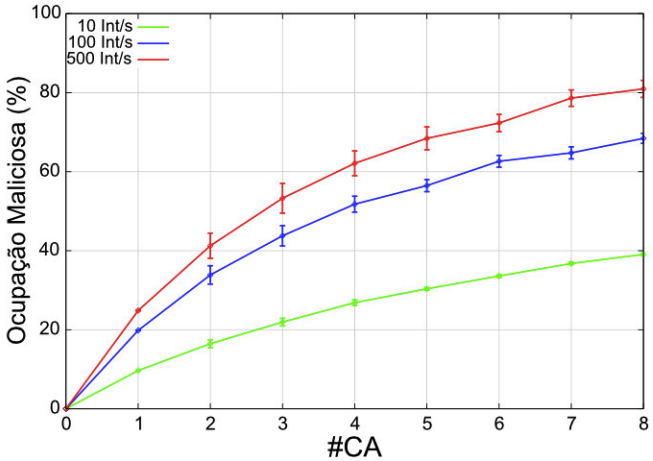
Figura 3. O tempo de recuperação de conteúdos legítimos.

A Fig. 3 mostra o comportamento do tempo médio de recuperação de conteúdos legítimos em função do número de consumidores maliciosos. Nas duas configurações, quando consumidores maliciosos solicitam conteúdos segundo o padrão CBR (Fig. 3(a)) ou quando solicitam conteúdos segundo a distribuição *Zipf* (Fig. 3(b)), o comportamento observado é o mesmo: quanto mais consumidores maliciosos, maior o tempo médio de recuperação de conteúdos. Da mesma forma, quanto maior a taxa de interesses maliciosos, maior o tempo médio de recuperação de conteúdos legítimos. Para a

configuração da Fig. 3(a), por exemplo, quando somente consumidores legítimos solicitam conteúdos, o tempo médio de recuperação de conteúdos legítimos é igual a 0,34 ms. Por outro lado, quando 4 consumidores maliciosos solicitam conteúdos esse tempo é igual a 1,92 ms e 7,21 ms, quando enviam 10 e 500 interesses/s, respectivamente. Quando há 8 consumidores maliciosos, o tempo médio de recuperação de conteúdos legítimos é igual a 2,93 ms e 7,99 ms para as taxas de 10 e 500 interesses/s, respectivamente. Isso mostra que o tempo de recuperação aumentou 23,5 vezes no pior caso para as configurações avaliadas. É importante ressaltar que como os consumidores legítimos possuem *cache* e como eles sempre consomem de acordo com a popularidade (*Zipf*), pode-se observar que sem ataque o consumo é, muitas vezes, feito do próprio *cache* do nó, o que resulta em um tempo de recuperação inferior a 1 ms.



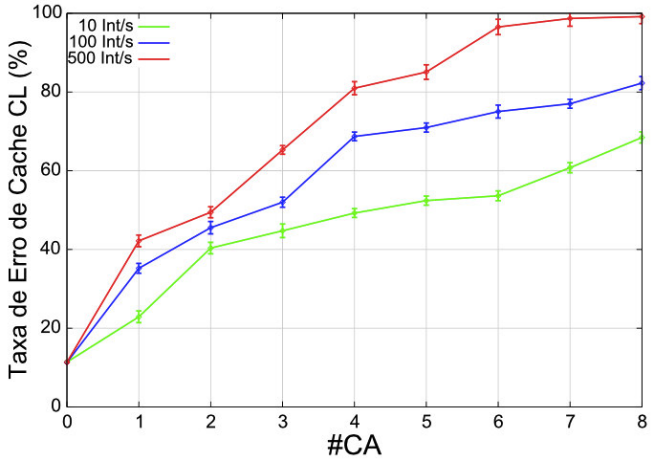
(a) CBR.



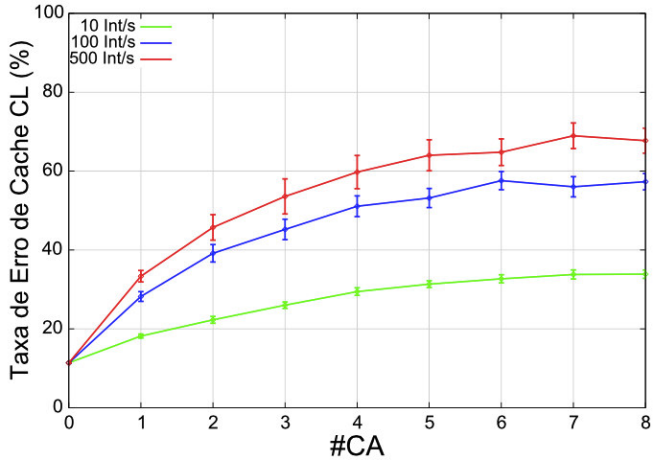
(b) Zipf.

Figura 4. O percentual de ocupação do *cache* dos roteadores por conteúdos maliciosos.

O aumento do tempo de recuperação nas duas configurações é explicado pelo aumento da ocupação maliciosa no *cache* dos nós intermediários e, consequentemente, do aumento da taxa de erros de *cache*, como mostram as Fig. 4 e 5, respectivamente. Quanto maior a ocupação maliciosa, maior a probabilidade do conteúdo solicitado não estar armazenado em *cache*. Para a configuração da Fig. 3(a) e curva para a taxa de 100 interesses/s, por exemplo, nota-se que com 8 consumidores maliciosos, o tempo de recuperação é da ordem de 6 ms. Como o atraso de cada enlace é de 1 ms, conclui-se que os conteúdos legítimos são recuperados mais frequentemente de nós que estão a mais saltos do consumidor do que os nós de borda. Isso indica que os roteadores de borda estão com uma alta ocupação de conteúdos maliciosos em seu *cache*. Nessa situação, os conteúdos maliciosos ocupam em média 80% do espaço total do *cache* dos roteadores, como mostra a Fig. 4(a). Para a configuração na qual os consumidores maliciosos solicitam conteúdos com base na sua popularidade (Fig. 3(b)), o tempo médio de recuperação também aumenta, mas esse aumento é menor do que o observado para a configuração baseada no consumo CBR (Fig. 3(a)).



(a) CBR.

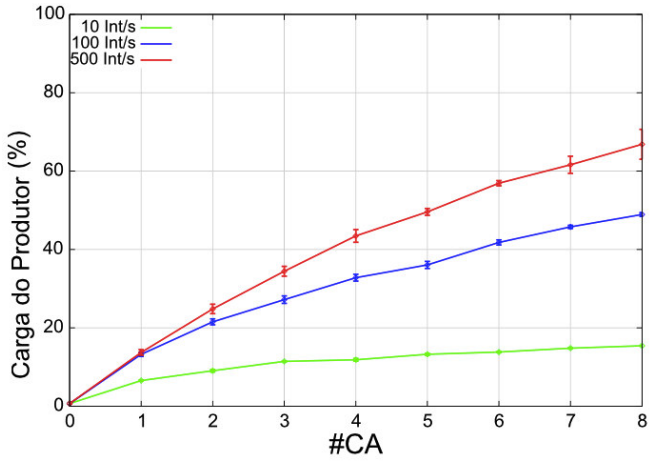


(b) Zipf.

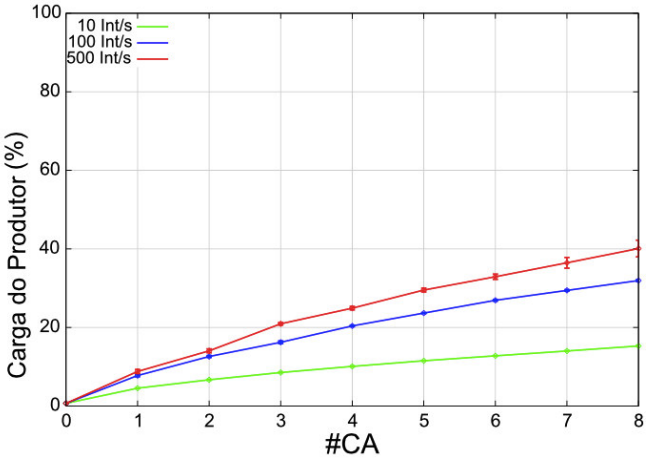
Figura 5. A taxa de erros de *cache* para os conteúdos legítimos.

Isso porque os consumidores maliciosos têm maior probabilidade de encontrarem conteúdos solicitados anteriormente em nós que estão a um ou dois saltos de distância e seus interesses não chegam ao produtor malicioso. Assim, a ocupação maliciosa na rede é menor (Fig. 4(b)), em especial nos nós próximos ao produtor. Assim, tem-se uma taxa de erros de *cache* menor (Fig. 5(b)) e um tempo de recuperação menor para consumidores legítimos.

A Fig. 6 mostra o percentual de conteúdos legítimos recuperados do produtor em função do número de consumidores maliciosos e da taxa de envio de interesses por esses nós. Esses resultados corroboram que o ataque em conluio reduz e eficiência do emprego do *cache* pela CCN. A Fig. 6(a) mostra que se não há ataque cerca de 0,5% dos conteúdos solicitados são recuperados diretamente do produtor. Nesse caso, cada conteúdo legítimo é recuperado do produtor no máximo duas vezes, até que seja armazenado pelos nós RTR1 e RTR2 (Fig. 2). Porém, basta se ter 4 consumidores maliciosos operando a taxa de 10 interesses/s para que esse valor aumente para cerca de 12%. No pior caso, os consumidores legítimos estão recuperando cerca de 67% dos conteúdos legítimos diretamente do produtor.



(a) CBR.



(b) Zipf.

Figura 6. Percentual de carga do produtor legítimo.

Um resultado interessante é que uma ocupação maliciosa de quase 100% quando há 8 consumidores maliciosos enviando 500 interesses/s (Fig. 4(a)) não resulta em 100% de conteúdos recuperados do produtor (Fig. 6(a)). Tal fato é explicado pelo uso de *cache* pelos próprios consumidores. Assim, é possível recuperar o conteúdo do próprio *cache*, sem ter que encaminhar pacotes de interesse para outros nós. No entanto, quando sob ataque, os consumidores legítimos ainda têm solicitações de conteúdo encaminhadas até o produtor, mesmo que eles façam uso de *cache* e solicitem conteúdos de acordo com a popularidade. Portanto, isso comprova que o serviço é negado em virtude da ocupação maliciosa dos *caches* dos roteadores.

Outra observação interessante extraída dos resultados é que a distribuição de consumidores maliciosos é mais efetiva do que o aumento da taxa agregada de envio de interesses maliciosos. Por exemplo, na Figura 3(a), é possível observar que o tempo médio de recuperação de conteúdos legítimos é da ordem de 5 ms quando 4 consumidores maliciosos enviam 100 interesses/s cada um (taxa agregada de 400 interesses/s) ou quando um consumidor malicioso envia sozinho 500 interesses/s. Esse fato é explicado pela ocupação maliciosa dos *caches* ser mais efetiva quando o ataque é distribuído. Para o mesmo exemplo anterior, a Fig. 4(a) mostra que a ocupação maliciosa quando há 4 atacantes enviando 100 interesses/s é da ordem de 50%. Quando há um atacante apenas enviando 500 interesses/s ela é de 40%. Esse fato se repete em outros pontos dos gráficos, considerando também o consumo baseado na popularidade. Por exemplo, na Fig. 3(b), quando 8 consumidores maliciosos enviam 10 interesses/s cada um (taxa agregada de 80 interesses/s) o tempo médio de recuperação de conteúdos legítimos é da ordem de 3 ms. Se um consumidor malicioso envia sozinho 100 interesses/s, esse tempo é menor do que 2,5 ms.

É importante ressaltar que em todos os experimentos realizados, os consumidores legítimos recuperaram todos os conteúdos solicitados. Isso pode ser explicado pelo fato da PIT não ter seu tamanho limitado e por ter sido usado um temporizador para remoção de entradas desta tabela da ordem de 4 s. Nas configurações usadas, esse tempo é muito maior do que o tempo necessário para um pacote de interesse legítimo ser encaminhado até o produtor e o pacote de dados ser encaminhado pelo caminho reverso até o consumidor. No pior caso, como mostra a Fig. 3(a), esse tempo é de aproximadamente 8 ms. Além disso, o produtor nunca remove do seu *cache* o conteúdo produzido por ele próprio. Portanto, nenhum dos nós da rede remove uma entrada da PIT nas configurações usadas antes do consumidor legítimo receber o conteúdo solicitado, mesmo que para isso, os pacotes de interesse tenham que ser encaminhados até o produtor.

VII. CONCLUSÃO

Este trabalho avaliou o ataque de negação de serviço em conluio produtor-consumidor para a arquitetura CCN. Esse ataque visa aumentar o tempo de recuperação de conteúdos aumentando a ocupação do *cache* dos nós intermediários com conteúdos maliciosos. Além disso, o ataque em conluio produtor-consumidor, não é identificado pelo mecanismo padrão de verificação de assinaturas da CCN porque os pacotes maliciosos carregam uma assinatura digital válida.

Diferentes configurações foram usadas nas simulações, variando-se o número de consumidores maliciosos, a política de consumo desses consumidores e taxa de pacotes de interesse maliciosos. Os resultados mostram que o ataque em conluio é efetivo, o que compromete o emprego do *cache* pela CCN. No pior caso, o tempo de recuperação aumentou 23,5

vezes para as configurações avaliadas. Esse aumento se deve a uma ocupação maliciosa média de 99% e, consequentemente, a uma taxa de erro de *cache* de 99%. Com isso, os consumidores legítimos recuperando 67% dos conteúdos solicitados diretamente do produtor. Mostra-se também que a distribuição de consumidores maliciosos é mais efetiva do que o aumento da taxa agregada de envio de interesses maliciosos. Os trabalhos futuros incluem a avaliação de outras políticas de substituição de *cache* e do emprego do mecanismo de *push-back* proposto por Gasti *et al.* O objetivo é verificar se tal mecanismo é eficiente como contramedida ao ataque em conluio. Caso não seja, o próximo passo é propor uma contramedida. Sobre os cenários, pretende-se empregar topologias reais e com maior escala nas simulações.

AGRADECIMENTOS

Este trabalho é apoiado por Dinter UFF/UFAC, CNPq, CAPES, FAPERJ, Proppi/UFF, TBE/ANEEL e CELESC/ANEEL.

REFERÊNCIAS

[1] A. Afanasyev, P. Mahadevan, I. Moiseenko, E. Uzun, e L. Zhang, "Interest flooding attack and countermeasures in named data networking," in IFIP Networking, May 2013, pp. 1–9.

[2] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: Evidence and implications," in IEEE Conference on Computer Communications - INFOCOM, Mar. 1999, pp. 126–134.

[3] G. M. Brito, P. B. Velloso e I. M. Moraes, "Redes orientadas a conteúdo: Um novo paradigma para a Internet." Em Minicursos do Simpósio Brasileiro de Redes de Computadores - SBRC, Abr. 2012 pp 211–264.

[4] G. M. Brito, P. B. Velloso, and I. M. Moraes, Information-Centric Networks, A New Paradigm for the Internet, 1st ed., ser. FOCUS - Networks and Telecommunications Series. Wiley-ISTE, 2013.

[5] S. Choi, K. Kim, S. Kim, and B. Roh, "Threat of DoS by interest flooding attack in content-centric networking," in Information Networking International Conference, Jan. 2013, pp. 315–319.

[6] P. Gasti, G. Tsudik, E. Uzun, and L. Zhang, "DoS and DDoS in named-data networking," in International Conference on Computer Communications and Networks - ICCCN, Aug. 2013, pp. 1–7.

[7] F. Q. Guimarães, I. C. G. Ribeiro, A. A. de Rocha e C. V. N. Albuquerque. "Nem tanto nem tão pouco: Existe um timeout Ótimo para PIT CCN na mitigação de ataques DoS," Em Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais - SBSeg, Out. 2013.

[8] V. Jacobson, D. Smetters, J. Thornton, M. Plass, N. Briggs, and R. Braynard, "Networking named content," in International Conference on emerging Networking EXperiments and Technologies - CoNEXT, Dec. 2009, pp. 1–12.

[9] Y. Kim, U. Kim, and I. Yeoml, "The impact of large flows in content centric networks," in IEEE International Conference on Network Protocols - ICNP, Oct. 2013, pp. 1–2.

[10] I. C. G. Ribeiro, A. A. de A. Rocha, C. V. N. Albuquerque, and F. Q. Guimarães, "On the possibility of mitigating content pollution in content-centric networking," in Conference on Local Computer Networks (LCN), Sep. 2014, pp. 498–501.

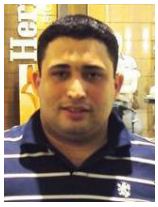
[11] I. C. G. Ribeiro, A. A. de A. Rocha, C. V. N. Albuquerque, and F. Q. Guimarães, "CCNcheck: um mecanismo de mitigação para poluição de conteúdos em redes centradas em conteúdo," Em Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais - SBSeg. Out. 2013.

[12] I. C. G. Ribeiro, F. Q. Guimarães, J. F. Kazienko, A. A. Rocha, P. B. Velloso, I. M. Moraes e C. V. N. Albuquerque, "Segurança em redes centradas em conteúdo: Vulnerabilidades, ataques e contramedidas." Em Minicurso do Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais - SBSeg. Out. 2012, pp 101-150.

[13] D. Smetters and V. Jacobson, "Securing network content," Xerox Palo Alto Research Center - PARC, Tech. Rep. TR-2009-1, 2009.

[14] M. Xie, I. Widjaja, and H. Wang, "Enhancing cache robustness for content-centric networking," in IEEE Conference on Computer Communications - INFOCOM, Mar. 2012, pp. 2426–2434.

[15] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. Thornton, D. K. Smetters, B. Zhang, G. Tsudik, K. Claffy, D. Krioukov, D. Massey, C. Papadopoulos, T. Abdelzaher, L. Wang, P. Crowley, and E. Yeh, "Named Data Networking (NDN) project," Xerox Palo Alto Research Center - PARC, Tech. Rep. NDN-0001, 2010.



André Luiz Nasserala Pires possui Mestrado pela Universidade Federal Fluminense (2010) e faz Doutorado em Computação na mesma instituição, é graduado em Bacharelado em Sistemas de Informação pela Universidade Federal do Acre (2002). Atualmente é professor da Universidade Federal do Acre. Tem experiência na área de Ciência da Computação, com ênfase em redes de computadores, atuando principalmente no seguinte tema: Segurança em Redes de Computadores.

Igor Monteiro Moraes é Professor Adjunto do Departamento de Ciência da Computação do Instituto de Computação da Universidade Federal Fluminense (UFF) desde 2010. Igor recebeu o título *cum laude* de Engenheiro Eletrônico e de Computação em 2003 e os títulos de Mestre e Doutor em Engenharia Elétrica pela Universidade Federal do Rio de Janeiro (UFRJ), respectivamente, em 2006 e 2009. Seus temas de interesse são as redes orientadas a conteúdo, as redes oportunistas, as arquiteturas para a Internet do Futuro, os sistemas par-a-par, as redes sem fio e a segurança em redes de computadores. Igor é membro da SBC.