Um serviço para prover autenticação e revogação de nós na rede DHT

Jean T. Garcia[1], **Lucas V. Dias**[1], **Tiago A. Rizzetti** [1]

[1]Curso Superior de Tecnologia em Redes De Computadores
– Universidade Federal de Santa Maria (UFSM)
Av. Roraima nº 1000 Cidade Universitária Bairro Camobi
Santa Maria - RS CEP: 97105-900 +55 (55) 3220-8000

{jeangarcia}@redes.ufsm.br, lucas_dias@redes.ufsm.br, rizzetti@gmail.com

Abstract. *In this paper we propose a service that provides authentication and revocation of nodes at runtime communications on a DHT network. Authentication is provided through digital certificates coupled with a public key infrastructure. Search speed tests and data success rate were performed. Considering the verification of authentication and signing of messages before obtaining the information from the network. The tests were performed using computational means simulating a network environment. Thus, the results showed that the proposed service is advantageous to provide secure communication in the DHT network, maintaining overlap performance without the implemented service.*

Resumo. *Neste artigo propomos um serviço que fornece autenticação e revogação de nós em tempo de execução das comunicações em uma rede DHT. A autenticação é provida através de certificados digitais com o acoplamento de uma infraestrutura de chave pública. Foram realizados testes de velocidade de busca e taxa de sucesso de obtenção de dados. Considerando a verificação de autenticação e assinatura de mensagens antes de obter a informação da rede. Os testes foram realizados utilizando meios computacionais simulando um ambiente de rede. Dessa forma, os resultados obtidos demonstraram que o serviço proposto é vantajoso para prover uma comunicação segura na rede DHT, mantendo o desempenho da sobreposição sem o serviço implementado.*

1. Introdução

As redes baseadas em *distributed hash table* (DHT) estão cada vez mais em evidência para implementação de diversos serviços [Rahimi et al. 2016]. Esta atenção especial se deve as suas características como tolerância a falhas, alta disponibilidade, escalabilidade, resiliência entre outros. Em uma rede DHT, cada nó armazena uma tabela de *hash*, nesta tabela é armazenado um par (chave, valor) qualquer nó pode recuperar ou alocar qualquer valor na DHT [Tang et al. 2008]. Estas mesmas propriedades da rede DHT fornecem um terreno fértil para que agentes maliciosos explorem diversas vulnerabilidades [Srinivasan and Aldharrab 2019]. Neste sentido, garantir uma comunicação segura entre os participantes da rede DHT torna-se um grande desafio [Ismail et al. 2016]. Este artigo concentra-se em propor e implementar um serviço reativo que fornece autenticação e revogação de nós. No que se refere a autenticação o serviço proposto utiliza certificados digitais para verificar a identidade de nós. As questões de emissão e revogação dos certificados digitais são tratadas pela infraestrutura de chave pública (ICP).

Dessa forma, as principais contribuições do trabalho são, oferecer uma rede segura, resiliente na presença de nós maliciosos. Além disso, um serviço de revogação de nós reativo, isto permitirá que um nó imediatamente seja revogado. O restante deste artigo esta organizado da seguinte forma. Na seção, 2 é feita uma breve descrição da bibliografia elencando os trabalhos relacionados. Na seção 3, é abordado o serviço proposto. Após, na seção 4, é descrito o cenário de execução dos testes. Além disso, são apresentados os resultados obtidos. Após, é feita uma análise contrapondo os resultados obtidos no presente artigo com os trabalhos relacionados. E por fim na seção, 5 a conclusão deste trabalho elencando os pontos negativos e positivos da abordagem proposta.

2. Trabalhos Relacionados

No trabalho proposto por [Pecori 2015] é apresentado um mecanismo de confiança aplicado ao protocolo do Kademlia. Este mecanismo proposto baseia-se em uma pontuação de confiança em cada operação realizada obtenção ou alocação de dados na rede DHT. Esta abordagem, é desvantajosa, visto que um nó pode ser malicioso, mas se comportar como um nó confiável na rede [Rahimi et al. 2016] e em determinado momento denegrir o comportamento da rede. Diante disso, até que este nó se torne não confiável poderá ainda se comunicar com os demais participantes.

Na arquitetura proposta por [Kohnen et al. 2011] é utilizada a autenticação baseada em certificados digitais também aplicado ao protocolo do Kademlia. O trabalho de [Kohnen et al. 2011] assume que cada nó contém um certificado autenticado por uma autoridade de certificação e que cada nó confia nos nós que possuem certificados autenticados. Em cada operação do protocolo do Kademlia é aplicada uma pontuação de confiança. Um nó decide se aceita ou não a operação de outro nó com base nessa confiança. A abordagem dos certificados digitais é interessante, visto que atribui um certificado a cada nó por uma CA (autoridade de certificação) e dessa forma, um nó pode ser revogado a qualquer momento da rede.

3. Serviço Proposto

O serviço proposto é constituído de duas partes, o método de autenticação e o de revogação de nós. Em primeiro lugar a lista de revogação de certificados é periodicamente publicada na rede DHT pela autoridade certificação. Para ingressar na rede um nó deve contatar outro nó já inserido, chamado de nó de *bootstrap* [Maymounkov and Mazieres 2002]. Cada nó já ingressa com o certificado válido emitido pela CA. Para a autenticação inicial é estabelecida uma chave de sessão com o algoritmo *diffie-hellman*, isto evitará mensagens falsificadas durante a comunicação. Este processo pode ser melhor visualizado na figura 1.

Na etapa 1 o nó de *bootstrap* assina, seu certificado e envia para o nó solicitante. Após receber o certificado o nó solicitante verifica a autenticidade do pacote (2), verificando a assinatura da autoridade de certificação e se o certificado não foi revogado. Se o certificado foi autenticado o nó solicitante assina seu certificado e envia-o ao nó de *bootstrap* (4). No passo 5 então o nó, *bootstrap* válida o certificado do nó solicitante. Se o certificado for autentico, e não revogado (6), é feita a inserção do nó na rede DHT (7). A verificação da integridade e da autenticidade das mensagens trocadas é feita com a chave de sessão do algoritmo *diffie-hellman*. Algumas nomenclaturas importantes usadas neste artigo serão descritos a seguir.

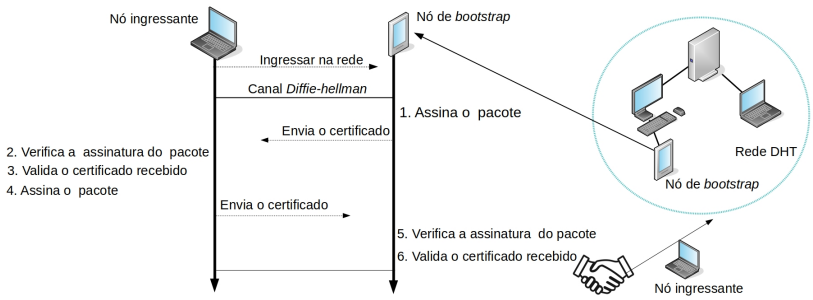

Figura 1. Modelo de autenticação inicial na rede DHT

- *Infohash*: É uma *hash* (chave) de 160 bits gerada pelo protocolo do Kademlia.
- *Get*: É operação de busca de dados da rede DHT.
- *Put*: É operação para alocar uma informação na rede.

No que se refere ao serviço de revogação de nós, cada nó na rede DHT terá um identificador, sendo este uma *infohash* do número de série do certificado. Um exemplo de comunicação pode ser visualizado na figura 2.

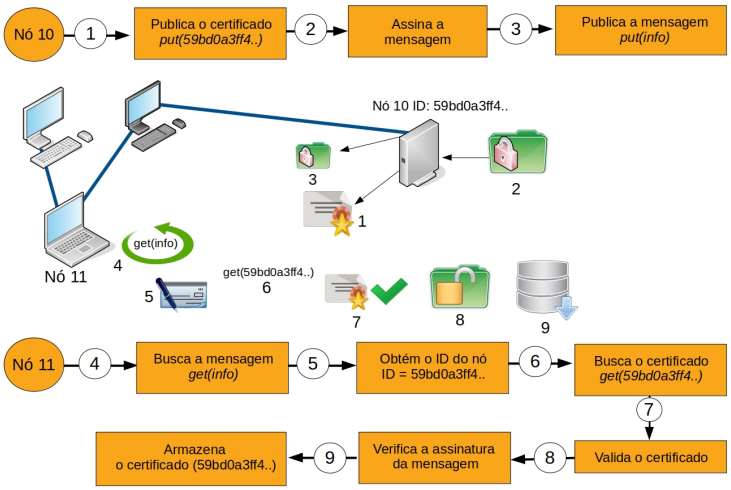

Figura 2. Exemplo do serviço proposto na rede DHT

Basicamente um nó assinará a mensagem, e publicará seu certificado e a mensagem assinada com sua chave privada na rede DHT. Do outro lado o nó realizará a busca pela mensagem, obtendo o identificador do nó e partir disso buscará o certificado do nó na rede DHT. Se o certificado for válido (autentico e não revogado), o nó então verifica a assinatura da mensagem com a chave pública do certificado. Para operações futuras o certificado é armazenado localmente.

4. Cenário de Testes

O cenário criado na ferramenta de simulação de redes CoreEmulator [U.S. Naval Research Lab 2015] contém 50 nós, número máximo permitido pelo equipamento computacional disponível. Para implementar o serviço foi desenvolvida uma aplicação na linguagem C++ com a biblioteca do OpenDHT [Savoir-faire Linux Inc 2019]. Com esta biblioteca foi possível colocar a rede DHT em funcionamento. O equipamento computacional utilizado foi um computador com as seguintes características:

- Processador: Intel® Core® I3
- Memória RAM: 8 Gigabytes
- Sistema operacional: Linux mint 19.1 64 bits
- Versão do CoreEmulator: 4.8

No que se refere ao processo de teste, foram executados cinco rodadas de testes em cada configuração. As *infohashes* publicadas pelos nós foram geradas pseudo-aleatoriamente dentro do espaço total de nós (50 nós). Isto significa que os nós escolheram a *infohash* entre (1 - 50). Para garantir a persistência das informações cada nó republicou a informação em um tempo pseudoaleatório gerado entre 1 e 10 segundos, este tempo foi escolhido com o objetivo de não permitir que um nó imunde a rede com suas infohashes. Os certificados revogados pela CA também eram gerados de forma pseudo-aleatória, isto fez com que cada nó tivesse a mesma probabilidade de ser revogado. Com relação à porcentagem de nós maliciosos . No teste de *get* foi gerada uma lista de revogação com 20% de nós não autênticos.

4.1. Testes e Resultados

Com 20% de nós não autênticos no cenário, a taxa de sucesso de *get*, por exemplo com 50 nós e 5000 *infohashes* publicadas foi de 79,4%. Isto significa o restante (20.6%), das informações foram descartadas, visto que não eram válidas. Isto comprova a eficácia da abordagem reativa, em que a autenticação é verificada em tempo de execução das comunicações na rede. Em alguns casos a porcentagem de não sucesso de *get* foi maior que o número de nós maliciosos na rede. Isto se deve ao fato de que os nós publicavam a informação mais de uma vez. O gráfico geral pode ser visualizado na figura 3. Com o intuito de avaliar o desempenho do serviço proposto em relação à rede DHT com o serviço ausente, foi realizado um teste de velocidade de busca das *infohashes* com o serviço proposto implementado. O gráfico da figura 4 demonstra que o serviço não degradou de forma significativa o desempenho da rede DHT.

4.2. Análise com os trabalhos relacionados

No que se refere ao trabalho de [Pecori 2015], o resultado de operações de (*get*) bem sucedidas é cerca de 80%, com nenhum nó falso na rede. Isto significa que possivelmente pela relação de confiança abordada as comunicações estão sendo ignoradas. No que se refere a proposta de [Kohnen et al. 2011] o resultado de busca considerando cerca de 40% de nós maliciosos a taxa de sucesso de *get* é quase 70%. Porém, novamente a questão da confiança leva aos participantes considerarem nós maliciosos como confiáveis. A abordagem proposta no presente artigo, insere algumas contribuições vantajosas neste sentido. Isto porque, mesmo com a presença de uma significativa porcentagem nós falsos

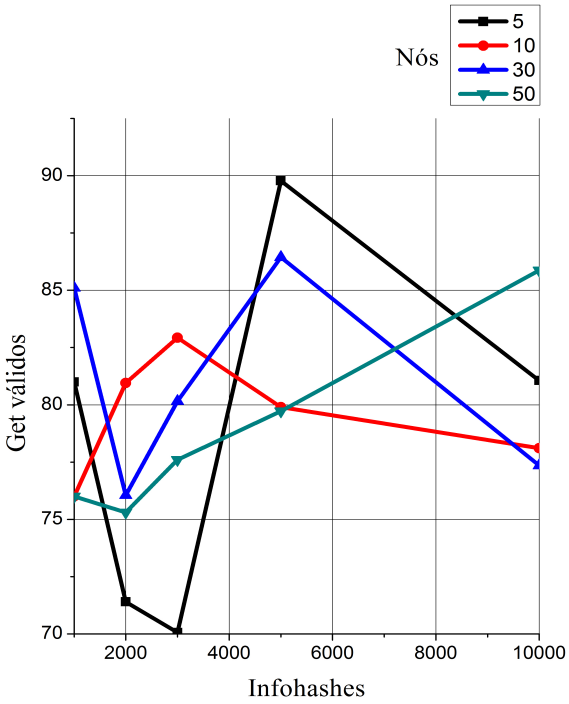

Figura 3. Resultados de *get* válidos com 20% de nós maliciosos.

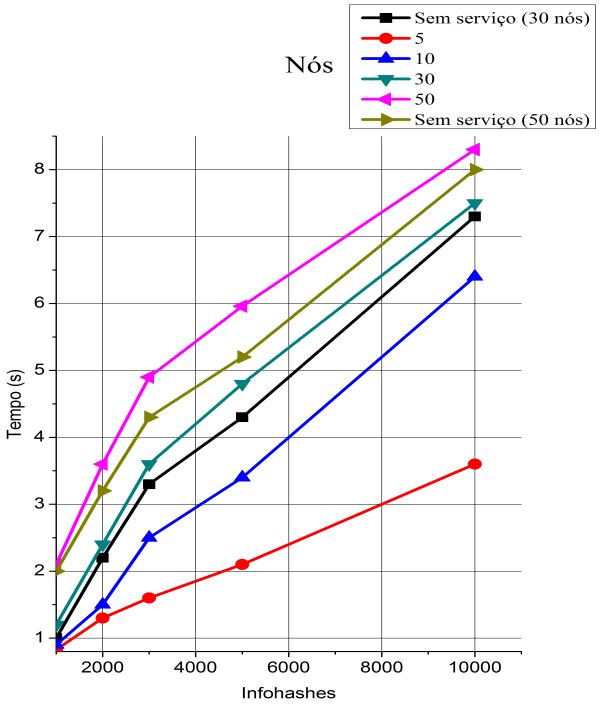

Figura 4. Resultados do teste de velocidade de busca das *infohashes*

tentando se comunicar na rede DHT. O sucesso de busca de uma informação na rede DHT foi confiável considerando os nós não autênticos com base na figura 3. Além disso, um nó é verificado em tempo de comunicação na rede, o que impede a comunicação futura deste nó a partir da revogação deste pela autoridade de certificação. E por fim, a abordagem

não impactou significativamente o desempenho da sobreposição com o serviço proposto.

5. Conclusão

Atualmente (2019), a rede do Mainline é um bom exemplo do uso das vantagens da rede DHT baseadas no Kademlia implementadas na internet [Xinxing et al. 2016] com milhões de usuários ativos. Neste sentido, a segurança da informação especialmente o pilar de autenticação tem sido um grande desafio [Shin et al. 2019]. Diante disso, este artigo trouxe um serviço para prover autenticação e revogação de nós em uma rede DHT. Foi implementado uma aplicação na linguagem C++ com a biblioteca do openDHT para verificar o serviço proposto. Para teste do serviço foi utilizado meios computacionais simulando um ambiente de rede com o CoreEmulator. Dessa forma, os resultados obtidos demonstraram que o mecanismo de revogação e autenticação é muito vantajoso para garantir esses aspectos de segurança.

Referências

Ismail, H., Germanus, D., and Suri, N. (2016). Malicious peers eviction for p2p overlays. In *2016 IEEE Conference on Communications and Network Security (CNS)*, pages 216–224.

Kohnen, M., Gerbecks, J., and P Rathgeb, E. (2011). Applying certificate-based routing to a kademlia- based distributed hash table.

Maymounkov, P. and Mazieres, D. (2002). Kademlia: A peer-to-peer information system based on the xor metric. In *International Workshop on Peer-to-Peer Systems*, pages 53–65. Springer.

Pecori, R. (2015). Trust-based storage in a kademlia network infected by sybils. pages 1–5.

Rahimi, N., Sinha, K., Gupta, B., Rahimi, S., and Debnath, N. C. (2016). Ldepth: A low diameter hierarchical p2p network architecture. pages 832–837.

Savoir-faire Linux Inc (2019). Opendht. Acesso em 20 jul. 2019.

Shin, J., Islam, M. R., Rahim, M. A., and Mun, H.-J. (2019). Arm movement activity based user authentication in p2p systems. *Peer-to-Peer Networking and Applications*, pages 1–12.

Srinivasan, A. and Aldharrab, H. (2019). Xtra—extended bit-torrent protocol for authenticated covert peer communication. *Peer-to-Peer Networking and Applications*, 12(1):143–157.

Tang, X., Xu, J., and Lee, W. (2008). Analysis of ttl-based consistency in unstructured peer-to-peer networks. *IEEE Transactions on Parallel and Distributed Systems*, 19(12):1683–1694.

U.S. Naval Research Lab (2015). Core manual. Acesso em 20 jul. 2019.

Xinxing, Z., Zhihong, T., and Luchen, Z. (2016). A measurement study on mainline dht and magnet link. In *2016 IEEE First International Conference on Data Science in Cyberspace (DSC)*, pages 11–19.