Um Arcabouço para Validação de Conteúdo em Redes de Dados Nomeados baseado em Blockchain

Antonio M. de Sousa[1], **Francisco R. C. Araújo**[1], **Fabíola Greve**[1], **Leobino Sampaio**[1]

[1]Programa de Pós-Graduação em Ciência da Computação (PGCOMP)
Departamento de Ciência da Computação – Universidade Federal da Bahia (UFBA)
Salvador – BA – Brasil

{antonio.mateus, franciscorca, fabiola, leobino}@ufba.br

Abstract. *In recent years, the Named Data Networking (NDN) has been proposed as a promising paradigm to the Future Internet due to its content centric disruptive characteristic. To ensure the content authenticity, NDN uses digital signature, which due its complexity may result in additional overhead. As a result, some NDN implementations are susceptible to several attacks. This paper proposes a framework that aims at allowing the content authenticity and integrity verification by leveraging by the Blockchain's technology attributes. Emulation results show that the proposed framework successfully detect contents poisoning and unauthorized distribution performed by malicious providers.*

Resumo. *Recentemente, as Redes de Dados Nomeados (NDN) foram propostas como um paradigma promissor para a Internet do Futuro devido à sua característica disruptiva centrada no conteúdo. A fim de garantir a autenticidade de conteúdos, a NDN se baseia no uso da assinatura digital que, devido à complexidade envolvida no processo de validação, pode resultar em sobrecargas adicionais de processamento. Como resultado, implementações de NDN acabam sendo suscetíveis a um variado número de ataques. Este artigo propõe um arcabouço de segurança que, a partir dos atributos da tecnologia Blockchain, possibilita a verificação da integridade e autenticidade de conteúdos em NDN de maneira eficaz. Resultados obtidos, a partir de experimentos de emulação, mostram que o arcabouço proposto detecta ataques de adulteração de dados e evita a distribuição indevida de conteúdos feita por provedores maliciosos, com sucesso.*

1. Introdução

Nas Redes de Dados Nomeados (do inglês, *Named Data Networking* - NDN) [Zhang et al. 2014], a adoção do paradigma de comunicação centrada no conteúdo favorece o surgimento de diferentes tipos de ameaças, uma vez que o próprio conteúdo, replicado em diversos pontos da rede, torna-se o principal alvo dos atacantes. Para garantir a autenticidade de conteúdos em NDN, implementações se baseiam no uso da assinatura digital dos produtores para propagação de conteúdos, enquanto que os consumidores verificam a validade de tais assinaturas ao receber os conteúdos solicitados. O problema, no entanto, é que a checagem de assinatura não é obrigatória nos roteadores, dada a complexidade envolvida no processo de validação, que pode resultar em sobrecargas adicionais de processamento provenientes da implementação

dos mecanismos de verificação. Como resultado, implementações de NDN acabam sendo suscetíveis a um variado número de ataques [Tourani et al. 2018], tais como: o envenenamento de conteúdo (do inglês, *content poisoning*), que ocorre quando um nó malicioso insere conteúdos falsos na rede que podem ser adicionados nas *cache* de roteadores causando a distribuição de dados inválidos; a injeção de conteúdo (do inglês, *content injection/spoofing*), quando um nó malicioso, ao receber um pacote de dados, pode alterar seu conteúdo para infectar o consumidor ou simplesmente prejudicar a reputação do produtor; e, por fim, a pirataria de conteúdos (do inglês, *content piracy*), que consiste na distribuição, cópia ou venda de conteúdos sem a devida permissão dos detentores dos direitos autorais.

Devido a tais ataques, um número considerável de trabalhos encontrados na literatura já buscaram solucionar os desafios citados através de diversas técnicas que, em geral, utilizam *nomes auto certificados* ou a *abordagem de nomeação hierárquica*. A técnica de nomes auto certificados consiste em um modelo de nomeação plana que se baseia no *hash* do conteúdo. Assim, qualquer usuário pode verificar a integridade da informação recebida comparando o *hash* atual com o nome informado. A desvantagem dessa técnica está na perda de inteligibilidade, uma vez que os nomes auto certificados são ilegíveis para seres humanos e cria a necessidade de um mecanismo de tradução semelhante ao DNS. A nomeação hierárquica, por sua vez, traz nomes legíveis aos conteúdos e a autenticidade dos mesmos é garantida por meio do uso de criptografia assimétrica, onde o produtor assina o conteúdo antes de enviá-lo ao usuário. Este então deve obter a chave pública do fornecedor do conteúdo por meios externos. Todavia, nessa estratégia o usuário precisa obter o conteúdo primeiro para depois validá-lo, o que permite a realização de ataques de negação de serviço por meio de conteúdos falsos [Ribeiro et al. 2012].

As questões em aberto em tais abordagens de segurança em redes NDN podem ser endereçadas através da tecnologia blockchain [Greve et al. 2018]. A blockchain oferece uma rede de confiança digital, possibilitando a realização de transações entre pares que não se conhecem. Todas as transações são acordadas através de um protocolo de consenso e, em seguida, são executadas na mesma ordem e armazenadas numa base de dados replicada, conhecida como livro-razão, que estará distribuída na rede e acessível a todos os pares. A blockchain apresenta diversas propriedades que fazem com que seja preferida no suporte à segurança da informação, dentre elas ressaltamos a manutenção da integridade, disponibilidade, e autenticidade das transações.

Em NDNs, essa tecnologia tem sido adotada para garantir a integridade de conteúdos na *cache* em redes de sensores sem fio [Mori 2018], implementar uma rede blockchain propriamente dita sobre NDN, de forma a otimizar o desempenho de transações de criptomoedas e gerenciamento de chaves criptográficas [Jin et al. 2017]. Todavia, em sua maioria, os trabalhos não abordam o esquema de permissionamento de provedores ou sequer consideraram a presença de produtores maliciosos na rede.

Este trabalho advoga o uso da blockchain como alternativa eficaz para validar a integridade de conteúdos e a autenticidade de provedores em NDNs. Desta maneira, propõe um arcabouço (*framework*) que adiciona a robustez da tecnologia blockchain às NDNs. A implementação do arcabouço explora o conjunto de ferramentas fornecidas pela

plataforma Ethereum[1] para o desenvolvimento de aplicações distribuídas na blockchain. A proposta foi avaliada por meio da técnica de emulação na qual dois ataques foram aplicados para averiguar a eficácia do arcabouço. Os resultados obtidos mostram que a ferramenta conseguiu detectar com sucesso conteúdos maliciosos, além de manter um atraso constante de ≈ 2 segundos nas verificações.

O restante do trabalho está organizado da seguinte forma: na Seção 2 são apresentados os trabalhos relacionados, na Seção 3 a proposta é detalhada e os cenários de aplicação são abordados na Seção 4. Os experimentos e resultados são discutidos na Seção 5 e, por fim, o trabalho é concluído na Seção 6.

2. Trabalhos Relacionados

Em [Yu et al. 2017], foi proposto o arcabouço NDN DeLorean para autenticação de arquivos de dados de longa duração em NDN. O DeLorean fornece um serviço de auditoria de dados por meio da verificação de assinaturas em um livro-razão público. A eficiência desse arcabouço deve-se ao tempo de resposta reduzido, devido às buscas rápidas, via o caminho *hash* na árvore de Merkel (estrutura eficaz de armazenamento das transações) [Greve et al. 2018]. Os autores ainda propõem um esquema de incentivo para consumidores realizarem a auditoria de serviços que apresentam comportamento anormal. No entanto, o NDN DeLorean foi apenas discutido sendo que sua implementação foi deixada para trabalhos futuros. Além disso, o arcabouço não apresenta nenhuma funcionalidade relacionada ao permissionamento de novos provedores.

Em [Mori 2018], propõe-se um esquema para assegurar a integridade e autenticidade da fonte de dados de sensoriamento na *cache* em redes de sensores sem fio baseadas em NDN. Nesse sentido, os autores adotaram duas tecnologias chave: criptografia assimétrica e blockchain. Isso permite que os dados registrados na blockchain sejam verificáveis por qualquer outro nó sem a necessidade de reconfirmar a chave pública da fonte. Devido às limitações energéticas dos sensores, o processo de mineração é realizado nos nós da nuvem. O trabalho adota um cenário restrito, no qual o esquema não chegou a ser implementado sendo a validação realizada através de uma avaliação analítica.

Já em [Jin et al. 2017], os autores demonstraram a eficiência do paradigma NDN como alternativa aos problemas de escalabilidade, fraca conectividade e atrasos enfretados pela tecnologia blockchain como consequência da ineficiência das redes IP. Dessa forma, foi proposto um sistema Bitcoin descentralizado construído inteiramente sobre NDN. A partir da implementação de um protótipo, os autores concluíram que uma blockchain baseada em NDN consegue contornar os desafios atuais ao aumentar o tempo de conectividade, redução da sobrecarga da rede causada pela inundação de mensagens, além de criar um sistema mais descentralizado e simples.

Em [Fotiou and Polyzos 2016] foi apresentado um esboço de um esquema para validação de integridade e fornecimento de conteúdos em NDN baseado no mecanismo HIBE, que permite que uma identidade (p.ex., nome) possa ser usada como uma chave pública. O problema abordado reflete um cenário em que um detentor de um conteúdo deseja compartilhá-lo com alguns consumidores. Contudo, nesse trabalho não são relatados detalhes precisos sobre como se deu a implementação em NDN, assim como sua avaliação.

[1] `https://www.ethereum.org/`

3. Arcabouço Proposto

O arcabouço proposto neste trabalho tem como objetivo oferecer suporte à validação de conteúdos em NDN por meio da adoção da tecnologia blockchain. O arcabouço permite aos consumidores, e outros elementos da rede, averiguar a integridade dos conteúdos obtidos a partir de informações previamente cadastradas por seus respectivos produtores. A Figura 1 apresenta a arquitetura geral do arcabouço proposto. As próximas subseções detalham os elementos arquiteturais, suas interações e papéis atuados na implementação de mecanismos de segurança na NDN, as propriedades de segurança garantidas, a blockchain utilizada e, por fim, um exemplo de aplicação.

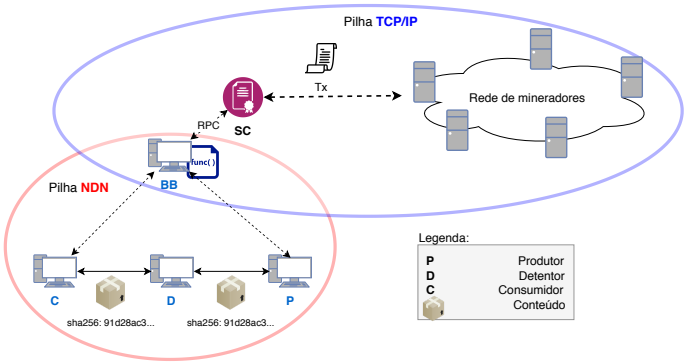


Figura 1. Arquitetura do arcabouço proposto.

3.1. Elementos Arquiteturais

O arcabouço possui seis elementos arquiteturais, utilizados na implementação dos recursos da blockchain e na NDN. Para a Blockchain (BC) prevê-se o Blockchain Broker (BB), o *Smart Contract* (SC) e a Rede de mineradores, enquanto que na NDN estão o Produtor de conteúdos (P), o Consumidor de conteúdos (C) e o Detentor de conteúdos (D).

O Blockchain Broker (BB) é o elemento responsável pelo acesso ao contrato inteligente (do inglês, *Smart-Contract* - SC) da BC. Sua função pode ser implantada num elemento de comutação da rede (p.ex., *Access point* - AP), nos próprios nós responsáveis pela produção de conteúdos ou em qualquer nó da rede com acesso ao SC. O BB é responsável por intermediar a comunicação entre um nó NDN e o SC que é construído sobre TCP/IP. Dessa forma, o BB possui as pilhas NDN e TCP/IP instaladas. Como alternativa à necessidade de um BB que atua como intermediário NDN-TCP/IP, uma blockchain inteiramente construída sobre NDN pode ser a opção adequada, como visto em [Jin et al. 2017]. O contrato inteligente (SC) é o elemento principal do arcabouço, dado que seu papel é realizar as principais funcionalidades do mesmo, desde o cadastro de novos conteúdos/provedores até consultas para validação de conteúdos. A Rede de mineradores é responsável por receber, validar e executar o conjunto de transações (Tx), que são realizadas através do SC, além de armazená-las no livro-razão da blockchain. Ou seja, são os nós que realizam o consenso e mantêm o livro-razão replicado.

Na estrutura da NDN, o Produtor (P) de conteúdo é responsável pela primeira disponibilização de cada conteúdo na NDN. O mesmo registra as informações do conteúdo na blockchain através do BB. O produtor também possui a capacidade de atualizar as informações do conteúdo além de registrar provedores visando ampliar a área

de distribuição. No papel de consumidor (C), o nó requisita conteúdos baseado no nome do mesmo (p.ex., `/ufba/dcc/main.html`), ou seja, o emissor desconhece as características do dado em si. Por isso, qualquer resposta contendo o nome solicitado será inocentemente aceita pelo consumidor. Por fim, o detentor de conteúdos (D) é o elemento da NDN que armazena temporariamente o conteúdo na sua memória *cache*, a qual é gerenciada por políticas de substituição. Essa função é desempenhada por qualquer nó que atua como encaminhador. O conteúdo na *cache* não é checado por razões de complexidade e desempenho [Ribeiro et al. 2012, Tourani et al. 2018]. Por conseguinte, caso um atacante esteja divulgando conteúdos falsos pela rede estes serão facilmente guardados na *cache* [Tourani et al. 2018].

3.2. Propriedades de Segurança

O arcabouço assegura a integridade dos dados ao armazenar o *hash* h do conteúdo (c) na blockchain para que qualquer usuário possa consultá-lo. Da mesma forma, as chaves públicas K_P^+ dos detentores dos direitos de cópia e produção do conteúdo também são disponibilizadas. Então, para realizar uma verificação de conteúdo o usuário deve ter em mãos a tripla $(K_P^-(c'), h', nome)$, em que K_P^- é a assinatura digital presente no conteúdo c' recebido, h' é o *hash* calculado sobre c', finalmente, o $nome$ do conteúdo. Com essas informações é possível checar tanto a integridade, ao comparar h' com h, quanto a autenticidade, utilizando K_P^+ sobre $K_P^-(c')$ que irá retornar verdadeiro se a assinatura for autêntica ou falso, caso contrário.

Nesse contexto, o objetivo deste trabalho é garantir integridade e autenticidade dos conteúdos em NDN. Sendo assim, as seguintes hipóteses foram consideradas: tanto o BB quanto os consumidores são considerados honestos, enquanto os produtores e os nós mineradores são Byzantinos, podendo assumir comportamento malicioso. Para lidar com BB e consumidores Byzantinos, futuras adaptações de segurança devem ser adicionadas ao arcabouço.

3.3. Elementos da Blockchain

O livro-razão, que armazena o conjunto de transações da blockchain, tem o formato de uma lista encadeada de blocos de transações. Cada um desses blocos, exceto o primeiro (i.e., o gênese), possui um apontador *hash* para seu predecessor, o que garante a resiliência de toda a cadeia de blocos. Os blocos carregam uma quantidade variável de transações válidas. A imutabilidade assegura a integridade das transações, pois tomando como exemplo um cenário hipotético onde um atacante consegue modificar um bloco B_k, nesse momento, haverá uma inconsistência que pode ser facilmente detectada verificando a lista na qual o apontador $hash(B_k)$ presente no bloco sucessor B_{k+1} estará inválido [Greve et al. 2018]. A verificação pode ser feita por qualquer par, pois a lista está presente, de maneira replicada, em todos os nós da blockchain.

A blockchain utilizada neste trabalho é responsável por validar e manter em sua cadeia de blocos transações que possuem: (i) informações sobre conteúdos e respectivos produtores, (ii) identidade de provedores autorizados e (iii) pagamentos realizados por candidatos a provedores pelos direitos de cópia do conteúdo.

A escolha entre *Blockchain Pública* ou *Federada* é fundamental no projeto de aplicações descentralizadas baseadas em blockchain, sendo as nuances de cada categoria

significativas no seu funcionamento e desempenho. A blockchain pública ou aberta não exige permissão dos nós na rede e requer apenas o par de chaves pública e privada para a realização de transações. Entretanto, a rede está sujeita a uma alta presença de nós maliciosos, além de uma maior exposição dos dados, compromentendo a sua privacidade. A blockchain federada ou privada requer autenticação dos nós na rede, além de um esquema maior de segurança, envolvendo, por exemplo, autoridade certificadora. Por outro lado, a blockchain pública requer um alto poder computacional para realizar o consenso prova de trabalho, próprio desse tipo de rede, enquanto a blockchain federada utiliza algoritmos de consenso de tolerância a falhas byzantinas tradicionais, tais como o PBFT, BFT-Smart, e cujo desempenho é o da latência da rede [Greve et al. 2018, Bano et al. 2017].

Para o caso específico do arcabouço, podemos utilizar tanto uma blockchain pública, quanto uma federada, sendo que a segunda categoria atende mais diretamente aos interesses de certas aplicações NDN, que precisam ter reconhecimento e maior controle do conjunto de nós oferecendo os serviços no sistema; vide, por ex., a distribuição de conteúdo no Netflix. Uma terceira via, de solução híbrida, também pode ser aplicada e esta é seguida no presente trabalho. Assim, propomos o uso da blockchain da Ethereum, que permite maior flexibilidade na manipulação das transações, além de apresentar os instrumentos para criação de *smart-contracts*. Ela oferece uma rede aberta, mas também contempla o suporte às aplicações da classe de identidade e sistemas de reputação, úteis à identificação dos nós (segundo o *white paper* da Ethereum [Buterin et al. 2013]).

3.3.1. Contrato Inteligente e Pacote de Interesse Especial

O SC é acessado através de seu endereço hexadecimal definido no momento em que o contrato é instanciado na blockchain. As funções desempenhadas pelo SC e suas respectivas *flags* são: (i) *registro de conteúdo* (REG), realizado pelo produtor original visando evitar fornecimento indevido; (ii) *validação de conteúdo* (VER), que se baseia nos dados registrados pelo produtor no momento do cadastro; (iii) *cadastro de provedores* (AUT), que são autorizados na blockchain pelo detentor original do(s) conteúdo(s); (iv) *atualização de conteúdo* (REG), restrito ao detentor dos direitos de modificação do(s) conteúdo(s); e (v) *listagem de produtores maliciosos* (VER), que consiste nas assinaturas inválidas detectadas pelo arcabouço. As *flags* são nomes de uso restrito em nosso cenário. Ademais, é importante destacar, as funcionalidades podem ser estendidas e novas podem ser criadas.

A comunicação em NDN se dá por meio dos pacotes de interesse e de dados. Sendo o pacote de interesse utilizado pelo consumidor para requisitar um conteúdo desejado. Uma vez que o pacote de interesse alcança um produtor do conteúdo mencionado, a requisição será satisfeita através de um pacote de dados retornado ao nó consumidor [Saxena et al. 2016, de Sousa et al. 2018]. Na implementação das funções do SC, a seguinte alteração básica foi necessária na arquitetura NDN padrão. Ela consiste na introdução de *pacotes de interesse especiais* que interagem com o SC requisitando alguma de suas funções. Nesse caso, nenhum campo adicional foi acrescentado à estrutura original do pacote de interesse da NDN padrão. Para tanto, foi utilizado o campo de dados opcional, já existente no próprio pacote, para transportar as informações exigidas para cada *flag* presente nos interesses especiais, ou seja, os rótulos que informam qual funcionalidade está sendo solicitada ao SC.

3.3.2. Formato das Transações

No arcabouço proposto, as transações (Tx) têm como objetivo registrar conteúdos ou assinalar provedores confiáveis. As consultas não criam uma nova Tx, pois não alteram o estado do SC. Diante disso, as transações possuem os seguintes campos: (i) Nome (*Content Name*), onde considera-se o esquema de nomeação hierárquica (e.g., `/ufba/dcc/main.html`), baseando-se em sua vasta adoção na literatura de NDN; (ii) Dono (*Owner*), que contém a chave pública do produtor original do conteúdo. (iii) *Content Hash* que consiste na função *hash* executada sobre o conteúdo, em que no caso da Ethereum é o SHA256; e (iv) Provedor (*Provider*) que contém a chave pública do provedor autorizado a distribuir o conteúdo. Para o registro de conteúdo, a transação possui os campos *Content Name*, *Content Hash* e *Owner*. Já para o permissionamento de provedores, a transação tem os campos *Owner* e *Provider*.

4. Aplicação do Arcabouço em NDN

4.1. Registro e Validação de Conteúdo

A Figura 2 ilustra um cenário de aplicação da proposta, considerando validação e presença de ataque de injeção de conteúdo. Nesse ambiente, há quatro protagonistas: `Alice`, `Bob`, `Eve` (atacante) e o contrato inteligente (SC). A personagem `Alice` é o produtor original do conteúdo `/ufba/dcc/main.html`. Sendo assim, `Alice` deseja garantir que nenhum outro produtor não autorizado o forneça. Para isso, `Alice` envia um interesse especial contendo o prefixo `/REG/ufba/dcc/main.html` (1º passo), no qual a *flag* "REG" indica uma requisição de registro de conteúdo. Nesse interesse são anexadas as informações do conteúdo como *hash* (h), nome e as credenciais do produtor (p.ex., K_P^+). O SC recebe a requisição de `Alice` e executa a função de cadastro de conteúdo, criando uma nova transação (Tx) que será adicionada a um bloco na blockchain. `Bob`, por sua vez, é um consumidor que solicita o conteúdo fornecido por `Alice` (2º passo) emitindo um interesse via *broadcast*. A mensagem é ouvida por `Eve` que também a encaminha para o produtor. `Alice` responde com um pacote de dados à requisição de `Bob` (3º passo). No entanto, `Eve`, que é um nó malicioso presente na rede, possui uma entrada em sua tabela de interesses pendentes (do inglês, *Pending Interest Table* - PIT) para o respectivo pacote de dados, pois atuou como encaminhador. A partir disso, visando efetuar um ataque contra `Bob`, `Eve` interrompe o fluxo normal da rede para alterar o conteúdo do pacote, além de se autodenominar produtor do mesmo (4º passo).

Para contornar o ataque, no 5º passo executa-se o principal papel do arcabouço que consiste na validação, quando `Bob` decide verificar a integridade do conteúdo, bem como a autenticidade de seu fornecedor. Assim sendo, `Bob` emite um interesse especial com a *flag* "VER" no prefixo seguido do nome do conteúdo, seu *hash* (h), chave pública (K_P^+) e assinatura digital do produtor ($K_P^-(c)$). É importante destacar que mesmo que `Eve` não altere $K_P^-(c)$ e mantenha a chave pública K_P^+ de `Alice`, o *hash* h' do conteúdo será suficiente para detectar uma provável adulteração. Por fim, o SC conclui que `Eve` não é um fornecedor autorizado e informa a `Bob` sobre o ataque em curso.

Neste trabalho foi assumido que as operações de "REG" e "AUT" são digitalmente asssinadas e podem ser verificadas com a chave pública do produtor presente na blockchain. Isso foi determinado a fim de evitar que `EVE` falsificasse alguma das mensagens especiais. Um esquema adequado para evitar problemas desse tipo poderia envolver

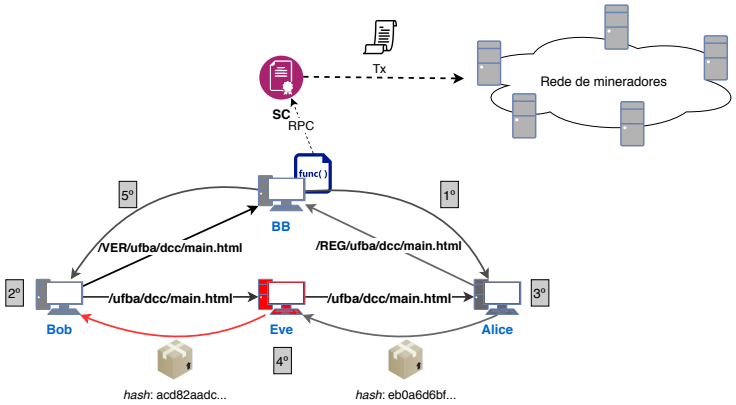


Figura 2. Aplicação do arcabouço em NDN.

técnicas complementares, como, esquemas de reputação, que poderão ser objeto de trabalhos futuros.

4.2. Permissionamento de Provedores

A Figura 3 ilustra a situação de permissionamento do provedor. Nesta, o personagem `Trudy` retrata um candidato a provedor do conteúdo fornecido por `Alice`. Com esse intuito, `Trudy` gera um interesse especial portando a *flag* "REQ" (1º passo) e o envia para `Alice`. Ao receber essa mensagem, `Alice` deduz por meio da *flag* que `Trudy` deseja participar da distribuição do conteúdo `/ufba/dcc/main.html` (2º passo). Nessa fase, `Alice` pode estabelecer um conjunto de regras com `Trudy` acerca da concessão do direito de fornecimento de cópias do conteúdo em questão. Tais regras podem ser definidas, mas não exclusivamente, pela relevância do conteúdo, tempo limite da licença, critérios de qualidade do aspirante a provedor, dentre outras. Quando o acordo é fechado, `Alice` envia um novo interesse especial contendo a chave pública de `Trudy` para o `SC` alertando através da *flag* "AUT" que um novo provedor para `/ufba/dcc/main.html` deve ser adicionado na blockchain (3º passo). No campo de carga útil desse interesse constam as chaves públicas e assinaturas digitais de ambos (produtor e provedor), endereços das carteiras digitais além de informações opcionais sobre o dado. É válido destacar que neste trabalho o produtor não impõem regras complexas no permissionamento. Além disso, é assumida a premissa de que o solicitante (i.e., `Trudy`) é confiável e apenas lhe é estipulada uma taxa pelos direitos de provimento do conteúdo que pode ser paga por meio de uma criptomoeda gerida pela própria blockchain.

5. Prova de Conceito

Os benefícios do *framework* proposto neste trabalho foram verificados através de uma prova de conceito, implementada por meio de emulação. Esta seção apresenta os detalhes de implementação, testes realizados e análise.

5.1. Ambiente de Testes

Para o desenvolvimento do arcabouço proposto, foi utilizado o ambiente para desenvolvimento de aplicações descentralizadas (do inglês, *Decentralized Applications* - DApps) para a plataforma Ethereum, chamado Truffle[2]. Adicionalmente, a ferramenta Ganache[3]

[2]`https://truffleframework.com/`
[3]`https://truffleframework.com/ganache`

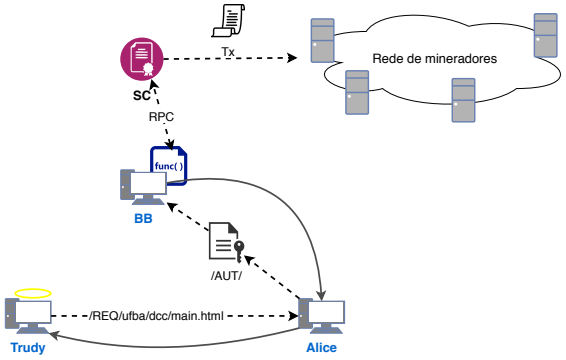


Figura 3. Processo de autorização de novos provedores.

também pertencente ao projeto Ethereum foi adotada em sua versão para linha de comando (i.e., ganache-cli). A Ganache provê uma blockchain privada que permite instanciar contratos inteligentes, bem como testá-los antes de serem propriamente lançados em uma blockchain real. A escolha dessas ferramentas foi motivada levando em consideração a acessibilidade, a documentação, curva de aprendizagem e baixo custo. O contrato inteligente foi escrito utilizando a linguagem Solidity[4]. A Solidity foi criada exclusivamente para implementação de SCs, sendo uma linguagem de alto-nível baseada em Python, C++ e JavaScript. Após o SC ser instanciado na blockchain, é preciso conhecer seu endereço para ser possível utilizar suas funções. A interação com o SC foi feita mediante o uso da API Web3 (versão 4.6.0) para a linguagem Python (versão 3.7)[5] que invoca as funções do SC via chamada de procedimento remoto (do inglês, *Remote Procedure Call* - RPC). Já a pilha NDN foi implementada sobre o protocolo UDP. O código completo, gráficos juntamente com um tutorial sobre instalação do cenário deste projeto estão disponíveis no Github[6].

5.2. Cenário de Avaliação

Para avaliar o protótipo criado foi empregada a técnica de emulação, onde o emulador de redes escolhido foi o CORE Emulator[7]. O cenário de avaliação implementado no CORE consiste em uma rede sem fio com 20 nós e a blockchain sendo executada fora da rede. Nesse ambiente, o AP atua como um intermediário (i.e., BB) para a rede local traduzindo as requisições de NDN para IP e em seguida as encaminhando ao SC. O meio sem fio foi escolhido como cenário de experimentação motivado por suas características de complexidade na manutenção de enlaces, mobilidade dinâmica, comunicação *broadcast* culminando em uma segurança fragilizada. Para aplicar mobilidade nos nós, um *trace* de mobilidade baseado no modelo *RandomWalk* foi gerado e integrado à emulação. A Tabela 1 sumariza os demais detalhes da emulação.

5.3. Métricas de Desempenho

Para certificar-se que o objetivo proposto foi alcançado e analisar o desempenho do arcabouço, as seguintes métricas foram fixadas:

[4]https://solidity.readthedocs.io/en/v0.4.25/
[5]https://web3py.readthedocs.io/
[6]https://github.com/mateus-n00b/mateus-sbrc19
[7]https://github.com/COREemu/CORE

Tabela 1. Parâmetros da Emulação.

Parâmetro	Valor
Número de nós	22
Tempo de emulação	200 segundos
Estratégia de encaminhamento	*Multicast*
Política de *cache*	sem *cache*
Densidade de nós maliciosos (%)	0, 10, 20 e 30
Densidade de produtores e consumidores (%)	10 e 30, respectivamente
Alcance de transmissão	200m
Dimensões do ambiente de emulação	700 $\times$ 700m
Tamanho do pacote de dados	1024B

- Taxa de satisfação de interesses (do inglês, *Interest Satisfaction Rate* - ISR): descreve a porcentagem de pacotes de interesse satisfeitos durante a emulação. O ISR auxilia na análise dos efeitos colaterais causados pelos ataques, pois os conteúdos alterados são descartados pelo consumidor.
- Não Verificados (NV): retrata a porcentagem de conteúdos que o consumidor recebeu mas não conseguiu verificar, isto é, os pacotes de interesse "VER" foram perdidos em consequência de desconexões ou por estar distante do AP.
- Atraso na satisfação de interesses (do inglês, *Interest Satisfaction Delay* - ISD): reflete o tempo decorrido entre o envio de um interesse e o retorno do conteúdo. Na avaliação do arcabouço o atraso total inclui o tempo de verificação.
- Verificações Positivas (VP): quantidade de verificações de conteúdos onde não foram detectadas alterações.
- Verificações Negativas (VN): quantidade de verificações de conteúdos nas quais foram detectadas ataques à sua integridade ou autenticidade.

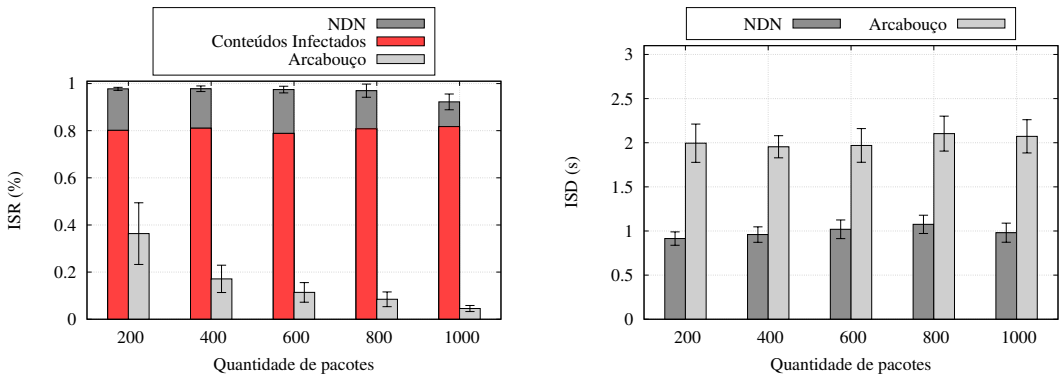
5.4. Análise de Desempenho

A avaliação do arcabouço consistiu na análise do desempenho em relação a uma NDN padrão, considerando que o mesmo implementa os recursos de segurança não existentes. Além disso, foi verificada a capacidade de detecção de conteúdos corrompidos.

5.4.1. Arcabouço versus NDN Padrão

Os experimentos iniciais realizados visaram demonstrar os efeitos da adoção do arcabouço proposto sobre a NDN. Para isso foram analisadas as métricas que refletem o atraso (ISD) e a resolução de interesses (ISR) sendo a quantidade máxima de requisições variada em cada experimento. Na Figura 4 são apresentados os resultados comparativos, onde a NDN padrão obteve um desempenho superior na métrica ISR, Figura 4(a), alcançando em média $\approx 74\%$ de satisfação enquanto o arcabouço obteve apenas $\approx 7\%$. Entretanto, o ISR mostrado na Figura 4(a) foi obtido em um cenário de ataque de injeção de conteúdo no qual 30% dos nós eram atacantes, dessa forma, na mesma figura também é apresentado a porcentagem de *pacotes infectados* que foram recebidos pelo consumidor. Ou seja, cerca de 80% dos conteúdos retornados tiveram sua estrutura original alterada, denotando que o alto índice de satisfação não reflete a confiança desejada pela NDN.

Já na Figura 4(b), é destacado a métrica ISD, a NDN padrão superou o desempenho obtido pelo arcabouço permanecendo abaixo do tempo de 2 segundos. No entanto, o desempenho inferior do arcabouço é consequência do acréscimo da etapa de verificação, isto é, o envio de um interesse especial "VER" para confirmação das informações do conteúdo na blockchain. Além disso, o atraso adicional não foi substancial mantendo-se estável apesar do aumento do número de requisições. Essa desvantagem pode ser contornada com a adoção de estratégias de encaminhamento eficientes que tratam o problema da inundação de pacotes na NDN sem fio.



(a) Taxa de satisfação. (b) Média do ISD.

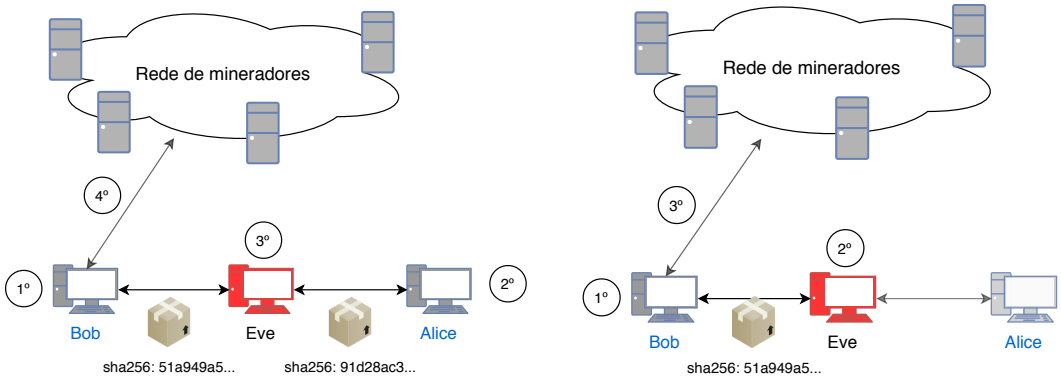
Figura 4. NDN padrão versus Arcabouço.

A partir dos resultados apresentados é possível observar que a NDN padrão está sujeita a fraudes nos conteúdos recebidos, logo há um comprometimento da confiabilidade na entrega do conteúdo. Dessa forma, o ISR por si só não reflete a real satisfação por não considerar a qualidade dos dados que estão sendo consumidos.

5.4.2. Detecção de Conteúdos Corrompidos

Na segunda etapa de avaliação, o arcabouço foi aplicado sobre o mesmo cenário mas dessa vez com a presença de nós maliciosos encarregados de interferir no ciclo normal da comunicação consumidor $\leftrightarrow$ produtor. Os ataques analisados, mostrados na Figura 5, são o *envenenamento de conteúdo* e *injeção de conteúdo*. No ataque *envenenamento de conteúdo* (Figura 5(a)), *Bob* envia um interesse solicitando um conteúdo qualquer (1º) que é encaminhado por *Eve* para *Alice*. *Alice* responde com um pacote de dados (2º), mas ao retornar *Eve* altera a estrutura original do pacote (3º). Apesar disso, com o auxílio do arcabouço, *Bob* evita o ataque (4º) descartando o pacote adulterado. O segundo ataque estudado foi o *injeção de conteúdo* (Figura 5(b)) no qual *Bob* emite um interesse para *Alice* (1º), no entanto, dessa vez, *Eve* não encaminha a requisição, ao invés disso o atacante gera um falso pacote que é entregue como resposta a *Bob* (3º). Novamente, *Bob* realiza uma consulta na blockchain para validar a autenticidade do pacote que finalmente é contestada (3º).

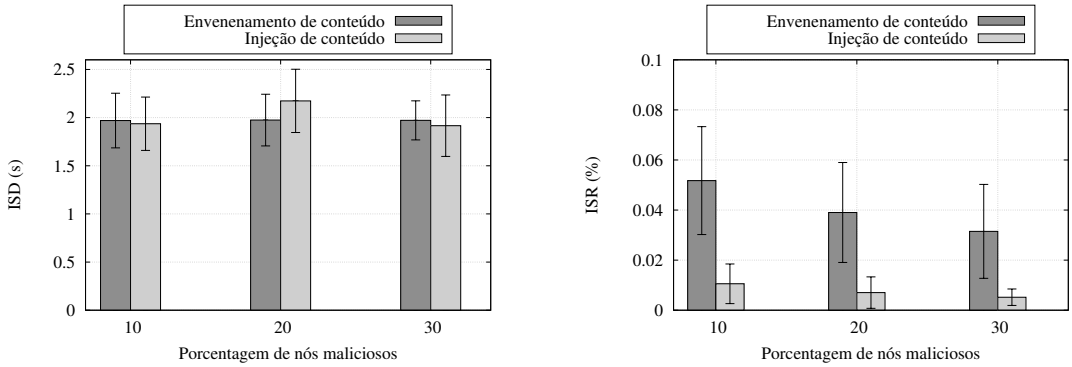
Nos experimentos executados foram variados a densidade de nós maliciosos na rede com o intuito de simular uma infecção gradativa tornando o cenário cada vez mais inóspito aos consumidores. A Figura 6 expõe os efeitos e intensidade de cada ataque sobre



(a) *Envenenamento de conteúdo*. (b) *Injeção de conteúdo*.

Figura 5. Cenários de ataques observados.

as métricas estabelecidas. O ISD mostrado na Figura 6(a) não sofreu impacto expressivo independente do ataque enfrentado. Por outro lado, o ataque *injeção de conteúdo* obteve sucesso considerável ao manter os níveis de satisfação de interesses em cerca de 1,6%, como visto na Figura 6(b). A eficácia desse ataque em comparação ao *envenenamento de conteúdo* está em sua rapidez, pois o nó malicioso não necessita se preocupar com o encaminhamento ou sequer esperar pelo conteúdo autêntico para realizar o ataque.



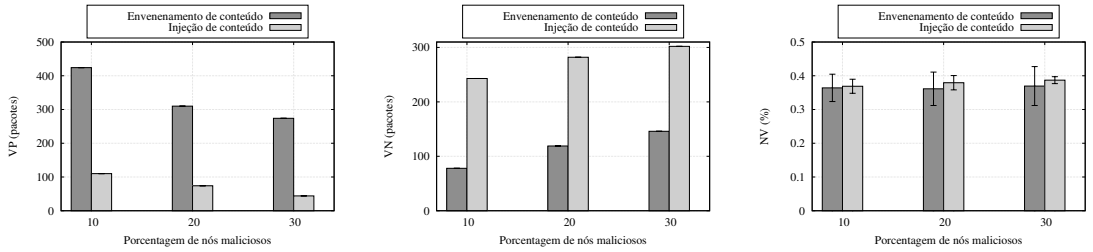
(a) Atraso relacionado ao arcabouço. (b) Taxa de satisfação.

Figura 6. Influência dos ataques sobre o arcabouço.

Por fim, na Figura 7 são agrupados os resultados referentes às detecções efetuadas pelo arcabouço. A quantidade total de pacotes autênticos analisados está destacada na Figura 7(a). Nela é possível observar que o tipo de ataque claramente influência nos resultados, e no caso do *envenenamento de conteúdo* há uma maior taxa de sucesso pois o atacante precisa esperar o retorno do conteúdo para concluir o ataque. Durante essa espera, cria-se a possibilidade do conteúdo alcançar o consumidor por meio de um caminho alternativo dificultando a consolidação do ataque. Para confirmar isso, a Figura 7(b) traz a quantidade de conteúdos adulterados identificados pelo arcabouço, o ataque *injeção de conteúdo* obteve maior sucesso em seu ataque do que o *envenenamento de conteúdo* visto que o atacante constrói um falso pacote de dados para cada interesse que chega.

Finalizando a avaliação, na Figura 7(c) é exibido o percentual de conteúdos recebidos mas não verificados. Essa métrica é influenciada pelo ambiente de avaliação, nesse

caso o meio sem fio onde há colisões e inundações que resultam em pacotes perdidos. Mesmo com as ações do meio físico, a porcentagem permaneceu estável em torno de 38% a 40% em ambos os cenários de ataque.



(a) Conteúdos válidados pelo consumidor. (b) Conteúdos inválidos detectados. (c) Conteúdos não verificados.

Figura 7. Detecção de fraudes e conteúdos não verificados.

5.5. Projeto da Blockchain e Desempenho

Como visto na Seção 3.3, a plataforma de blockchain adotada, se pública (aberta) ou federada (permissionada), e o consequente protocolo de consenso interferem sobremaneira no desempenho da rede, e por sua vez afetam diretamente na vazão ou quantidade de transações realizadas por segundo. A escolha de qual tipo de blockchain utilizar varia de acordo com o modelo de negócios da aplicação, requisitos de escalabilidade, desempenho e confidencialidade. No caso das NDN, ambas as blockchains podem estar sendo usadas, sendo que, para os requisitos da aplicação proposta nesse *framework*, um balanceamento entre desempenho e privacidade deve ser feito. Isto é, pode-se adotar uma arquitetura em que tanto o conjunto de mineradores (ou de nós que implementam a blockchain), quanto o conjunto de produtores, estariam numa rede federada, enquanto os clientes teriam autonomia para usar os serviços, sem a necessidade de identificação. Vale observar que, na implementação inicial do *framework* proposto foram assumidas premissas para fins de simplicidade e prova de conceito. Nesse sentido, a rede pública Ethereum foi usada. Mas, uma rede federada, como o Hyperledger Fabric[8] poderia ser também passível de utilização.

6. Conclusão e Trabalhos Futuros

O paradigma NDN é visto como uma possível solução frente aos problemas persistentes na arquitetura TCP/IP, como mobilidade e endereçamento. De qualquer forma, para uma melhor adaptação da NDN ao cenário atual, é preciso lidar com novos desafios em aberto. Neste trabalho foi abordado o problema da confiabilidade e integridade de conteúdos. Nesse contexto, este trabalho propôs um arcabouço que possibilita a um consumidor validar a confiabilidade de um conteúdo recebido com o uso da tecnologia blockchain. Um arcabouço foi implementado na blockchain Ethereum e os resultados obtidos com as emulações demonstraram que a proposta detecta um ataque em progresso, evitando assim a divulgação de conteúdos falsificados. Como trabalhos futuros serão adicionadas novas funcionalidades ao arcabouço, realizadas avaliações em cenários de maior densidade e com a presença de falsos positivos, além da comparação da solução proposta com as estratégias existentes na literatura.

[8] `https://www.hyperledger.org/projects/fabric`

Agradecimentos

Os autores agradecem o apoio do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

Referências

Bano, S., Sonnino, A., Al-Bassam, M., Azouvi, S., McCorry, P., Meiklejohn, S., and Danezis, G. (2017). Sok: Consensus in the age of blockchains.

Buterin, V. et al. (2013). Ethereum white paper, 2014. *URL https://github.com/ethereum/wiki/wiki/White-Paper*.

de Sousa, A. M., Araújo, F. R. C., and Sampaio, L. N. (2018). A link-stability-based interest-forwarding strategy for vehicular named data networks. *IEEE Internet Computing*, 22(3):16–26.

Fotiou, N. and Polyzos, G. C. (2016). Decentralized name-based security for content distribution using blockchains. In *Computer Communications Workshops (INFOCOM WKSHPS), 2016 IEEE Conference on*, pages 415–420. IEEE.

Greve, F., Sampaio, L., Abijaude, J., Coutinho, A. A. R., Brito, I. V. S., and Queiroz, S. (2018). Blockchain e a revolução do consenso sob demanda. In *Livro de Minicursos do XXXVI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC). Cap. 5. 2018*, Campos do Jordão, SP.

Jin, T., Zhang, X., Liu, Y., and Lei, K. (2017). Blockndn: A bitcoin blockchain decentralized system over named data networking. In *Ubiquitous and Future Networks (ICUFN), 2017 Ninth International Conference on*, pages 75–80. IEEE.

Mori, S. (2018). Secure caching scheme by using blockchain for information-centric network-based wireless sensor networks. *Journal of Signal Processing*, 22(3):97–108.

Ribeiro, I., Guimarães, F. Q., Kazienko, J. F., Rocha, A., Velloso, P., Moraes, I. M., and Albuquerque, C. V. (2012). Segurança em redes centradas em conteúdo: Vulnerabilidades, ataques e contramedidas. *Minicursos do XII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais—SBSeg 2012*, pages 151–195.

Saxena, D., Raychoudhury, V., Suri, N., Becker, C., and Cao, J. (2016). Named data networking: a survey. *Computer Science Review*, 19:15–55.

Tourani, R., Misra, S., Mick, T., and Panwar, G. (2018). Security, privacy, and access control in information-centric networking: A survey. *IEEE Communications Surveys Tutorials*, 20(1):566–600.

Yu, Y., Afanasyev, A., Seedorf, J., Zhang, Z., and Zhang, L. (2017). Ndn delorean: An authentication system for data archives in named data networking. In *Proceedings of the 4th ACM Conference on Information-Centric Networking*, pages 11–21. ACM.

Zhang, L., Afanasyev, A., Burke, J., Jacobson, V., Claffy, K., Crowley, P., Papadopoulos, C., Wang, L., and Zhang, B. (2014). Named data networking. *ACM SIGCOMM Computer Communication Review*, 44(3):66–73.