Capítulo 2

Blockchain e Aplicações em Saúde

Arlindo F. da Conceição[1],
Vladimir Moreira Rocha[2] e
Ricardo Felipe de Paula[1]

Abstract

Blockchain technology enables reliable, secure, distributed, and fault-tolerant data storage. With the emergence of smart contracts (programs that run above the Blockchain), the technology is no longer used only for financial purposes but also for more complex applications. This chapter aims to present the fundamental characteristics of Blockchain and how it works. We show the main use cases of Blockchain and a survey on their use focused on the health area. Also, we discuss the application scenarios and the challenges to be overcome for the implementation and deployment of the technology. Finally, we present two of the key platforms for building Blockchain applications: Ethereum and Hyperledger Fabric.

Resumo

A tecnologia Blockchain permite o registro de dados de forma confiável, segura, distribuída e tolerante a falhas. Com o surgimento dos contratos inteligentes (programas que são executados de forma autônoma sobre a tecnologia), a tecnologia deixou de ser usada somente para fins monetários e passou a ser utilizada para aplicações mais complexas. Este capítulo visa apresentar as características fundamentais da Blockchain e como ela funciona. São apresentados os principais casos de uso da Blockchain e um levantamento sobre a sua utilização na área da Saúde. Os cenários de aplicação e os desafios para implantação da tecnologia também são discutidos. Por fim, duas das principais tecnologias para criação de aplicações Blockchain serão apresentadas: Ethereum e Hyperledger Fabric.

[1]Universidade Federal de São Paulo (UNIFESP)
[2]Universidade Federal do ABC (UFABC)

2.1. Introdução

A tecnologia Blockchain, uma nova tecnologia para registro confiável e consenso distribuído [1, 2], oferece alternativas para a criação de sistemas interoperáveis, auditáveis e seguros. A tecnologia foi popularizada em 2008 com a criação da criptomoeda Bitcoin por Satoshi Nakamoto [3], Blockchain tem sido utilizada principalmente para realizar transações financeiras de forma anônima, auditável, confiável e segura, evitando que terceiros (por exemplo, os bancos) intermedeiem essas transações.

O conceito utilizado na Blockchain é o de *distributed ledger* e consiste, basicamente, em uma cadeia ordenada e consistente de transações, distribuída em diversos nós de uma rede *peer-to-peer*. Após o sucesso do Bitcoin, outras tecnologias foram integradas à versão inicial proposta por Nakamoto a fim de otimizar o desempenho da solução [4, 5, 6, 7, 8]. Entre elas, os arcabouços Ethereum [9] e Hyperledger [10] fazem uso dos contratos inteligentes, pequenos programas independentes armazenados na própria Blockchain, que permitem realizar operações na cadeia de registros da Blockchain. Por simplicidade, pode-se pensar nos contratos inteligentes, como sendo similares às *store procedures* existentes nos bancos de dados relacionais [2].

O uso de contratos inteligentes expande o poder da Blockchain, que além de armazenar estados (*e.g.*, saldo de uma conta no contexto financeiro), passa a poder armazenar comportamentos (*e.g.*, enviar mensagens de saldo insuficiente). Com o uso da Blockchain e de contratos inteligentes podemos atender outros contextos mais gerais, por exemplo: verificar a consistência da identificação única de usuários, mediar a interoperabilidade de dados em tempo real, garantir a privacidade das informações e tornar auditável todas as ações de acesso a esses dados.

Estima-se que a Blockchain terá um impacto relevante nos próximos anos [11], com aplicações em: registro da cadeia de fornecimento de insumos e produtos, aplicações de governança digital [12], Internet das Coisas (*Internet of Things* ou IoT) [13], Saúde [14, 15], gestão financeira, registros de imóveis, controle de ativos, registros de certidões (nascimento, casamento, óbito), entre outras categorias de aplicações.

Na área de Saúde, a Blockchain pode ser aplicada no controle de acesso e distribuição de informações sensíveis, na transparência e auditabilidade de prestação de serviços e na interoperabilidade de dados, entre outras situações. Pesquisas recentes, contudo, apontam que apesar do seu potencial, Blockchain é uma ferramenta que não pode ser aplicada com sucesso em todos os casos [16]. Desse modo, os profissionais de Saúde devem aprender a discernir os casos de uso viáveis nos quais a tecnologia realmente faça a diferença.

O objetivo desse minicurso é prover aos profissionais de Saúde, estudantes e pesquisadores o entendimento necessário para analisar a viabilidade de aplicação da tecnologia e os primeiros passos a serem dados na implantação de novos projetos envolvendo Blockchain. Para atingir esse objetivo, este texto apresenta a seguinte estrutura: a Seção 2.2 descreve o funcionamento básico de uma Blockchain. A Seção 2.3 mostra alguns conceitos computacionais importantes para o entendimento da tecnologia Blockchain, estes conceitos são pré-requisitos para a correta compreensão do potencial da tecnologia. A Seção 2.4 apresenta um quadro histórico do desenvolvimento da tecnologia. A seguir, a

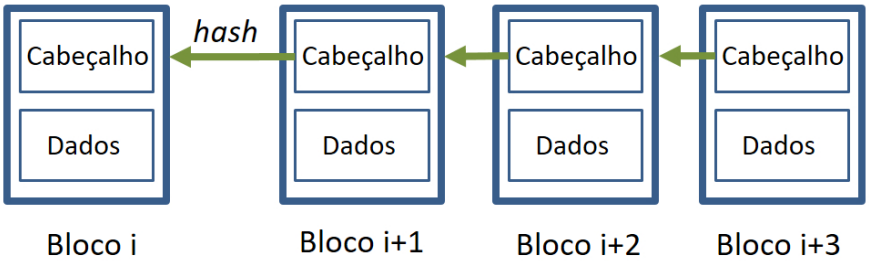


Figura 2.1. Estrutura básica de uma Blockchain

Seção 2.5 explica, passo a passo, como a tecnologia funciona. A Seção 2.6 lista alguns desafios que devem ser enfrentados na utilização de uma Blockchain. A Seção 2.7 apresenta algumas vulnerabilidades de segurança da Blockchain. A Seção 2.8 descreve em quais cenários há a necessidade de utilizar Blockchain. A Seção 2.9, que é o foco principal deste minicurso, apresenta os possíveis cenários de Blockchain aplicados à Saúde. A Seção 2.10, apresenta as tecnologias Ethereum e Hyperledger Fabric, que são as duas ferramentas/ambientes mais utilizados atualmente (destaco: abril de 2019) para a criação de novas aplicações baseadas em Blockchain. Então, a Seção 2.11 lista alguns recursos e leituras para um maior contato com a tecnologia. Por fim, na Seção 2.12, apresentamos nossas considerações finais e perspectivas sobre o futuro da tecnologia.

2.2. Estrutura básica de uma Blockchain

Blockchain implementa algo similar a um livro razão distribuído (*distributed ledger*) e consiste, basicamente, em uma cadeia ordenada e consistente de blocos; por isso o nome *Blockchain*. A Figura 2.1 ilustra a estrutura de blocos encadeados.

Outra característica importante é a de que a estrutura de blocos é replicada em uma rede *peer-to-peer*. Sempre que um novo bloco é criado, ele é enviado para todos os nós da rede. Cada nó verifica os dados do bloco antes dele ser efetivamente incorporado na cadeia de blocos.

As próximas seções explicam o funcionamento geral da estrutura de um bloco e sobre o processo de replicação. Na Seção 2.5 serão dados mais detalhes sobre o funcionamento da Blockchain.

2.2.1. Estrutura de Dados do Bloco

O bloco de uma Blockchain, em geral, pode ser dividido em duas partes: cabeçalho e dados.

O cabeçalho possui as informações responsáveis por identificar o bloco. Uma destas informações é o campo **identificador**, que consiste em um valor único e sequencial. Cada novo bloco inserido na cadeia terá como identificador o valor do identificador do bloco anterior incrementado em uma unidade. Outro campo é o ***timestamp***, que armazena as informações de data e hora aproximada da criação do bloco. Normalmente existe também um campo de assinatura, responsável por identificar e validar o criador do bloco. Por fim, o cabeçalho pode conter metadados sobre o conteúdo do bloco.

Na parte de dados ficam armazenadas as transações pertencentes ao bloco. Estas transações podem representar qualquer tipo de dados ou atividades: em uma aplicação financeira, por exemplo, podem representar a transferência de valores; em uma aplicação de saúde, podem ser um documento ou o link para um documento contendo dados médicos. Cada transação também possui seu identificador, além da informação de quem a criou, entre outras informações conforme a aplicação.

Cabe chamar a atenção para o fato de que uma transação não necessariamente carrega em si um significado financeiro. Na Blockchain, a transação é uma unidade coerente de informação que será replicada e validada por vários nós do sistema. Se o dado carregado por uma transação não for validável, talvez esse dado não precisasse ser armazenado em uma Blockchain, talvez o dado pudesse ser armazenado em um sistema de informação comum.

2.2.2. Replicação em vários nós: redes *peer-to-peer*

Desde os primórdios da Internet, o fornecimento de recursos (como páginas Web ou arquivos) é dado por uma arquitetura denominada cliente-servidor, onde um computador servidor é responsável por fornecê-los e os computadores clientes por requisitá-los. Essa arquitetura segue vigente até hoje, por exemplo nos sítios Web, onde um servidor fornece a página e seu navegador, atuando como cliente, a requisita.

A rede *peer-to-peer* (ou P2P) pode ser entendida como uma rede conectada de computadores onde cada um deles, denominado *peer*, pode atuar tanto como cliente quanto como servidor. Ela nasce como uma alternativa à arquitetura cliente-servidor, e cujos principais objetivos são: (i) aumentar a disponibilidade do recurso e (ii) aumentar a largura de banda de *upload* do sistema [17].

No primeiro ponto, assim que um *peer* P_A requisita (baixa) um arquivo de um *peer* P_B, este arquivo será armazenado por P_A e disponível para que outro *peer* P_C possa baixá-lo, repetindo o processo. Em outras palavras, o mesmo arquivo estará replicado tanto em P_A quanto em P_B. Note que se o *peer* P_B sair da rede, o *peer* P_C poderá baixar o arquivo de P_A, aumentando assim disponibilidade do mesmo.

O segundo ponto está relacionado ao primeiro. Dado que o arquivo está replicado, este poderá ser baixado de todos os *peers* que o armazenam. Note que na arquitetura cliente-servidor, somente um servidor é responsável por enviar o arquivo aos clientes (limitado à largura de banda de *upload* do servidor). Já na rede *peer-to-peer*, supondo que N *peers* tenham baixado o arquivo, a largura de banda de *upload* para enviar o arquivo será N vezes maior a da cliente-servidor, haja vista que cada *peer* se comporta como um servidor.

As redes *peer-to-peer* atualmente são utilizadas em diversas aplicações e sistemas. Exemplos do uso deste tipo de redes são o aplicativo BitTorrent [18] de compartilhamento de arquivos e o Skype [19] de vídeo-áudio conferência.

2.2.3. Confiança: a principal inovação

O uso da tecnologia Blockchain resolve alguns problemas técnicos que mostraremos mais adiante, mas a sua principal inovação é prover um mecanismo de **confiança**. O fato de

todos os blocos e transações serem validados por todos os nós dificulta que registros incorretos sejam inseridos na Blockchain. Se a maioria dos nós do sistema estiverem trabalhando para o bem da rede, então apenas registros corretos serão inseridos. Assim, a confiança não está em um nó, mas na rede de nós como um todo; a confiança está no comportamento coletivo. Por isso é possível criar aplicações sem uma entidade central confiável (*trusted third party* ou TTP); pois a confiança é depositada na rede e não em TTPs.

Mesmo que um usuário (através de um nó) tente enviar dados falsos ou incorretos na rede, os demais nós podem detectar esse comportamento e não inserir o dado na Blockchain. Podem, inclusive, banir o nó suspeito. O comportamento coletivo da rede é o que importa; enquanto houver interesse da maioria dos nós em que a rede continue apenas com dados corretos, pode-se considerar os registros confiáveis. Nesse ponto, note que os nós que sustentam a rede devem ter interesse em que a rede permaneça confiável; esse interesse pode variar de aplicação para aplicação, mas o interesse é um requisito importante para que a rede possa ser considerada confiável.

2.3. Fundamentos

A tecnologia Blockchain tem conquistado seu espaço pelas características que apresenta no desenvolvimento de aplicações, tais como descentralização, disponibilidade, integridade, auditabilidade e privacidade. A seguir serão analisadas as principais características desta tecnologia.

- **Descentralização da informação**

  A descentralização da informação refere-se à dispersão da informação, evitando que uma entidade central ou absoluta tenha o poder sobre ela. Na Blockchain, a descentralização deve ser observada por dois lados: (i) quem detém o poder de realizar alguma ação em uma informação; (ii) quem possui fisicamente a informação.

  No primeiro caso, se a informação está centralizada em uma instituição, organização ou pessoa, esta tem o poder de realizar qualquer ação sobre a informação. Note que diversas instituições no nosso dia a dia funcionam dessa forma. Por exemplo, no banco, você é dono de uma conta, mas as suas informações pessoais podem ser atualizadas em qualquer momento por algum gerente do banco. Nesse sentido, é o banco quem detém o poder da informação.

  No segundo caso, se a informação está armazenada somente nos computadores da instituição, organização ou pessoa, esta possui fisicamente a informação. Seguindo a linha do exemplo do banco, estes armazenam as informações em infra-estruturas próprias, onde pessoas externas não têm acesso.

  A Blockchain visa que as informações não estejam centralizadas nem da perspectiva do poder para realizar alguma ação, nem da perspectiva do armazenamento físico. No primeiro caso, são os participantes que decidem, em conjunto, que informações podem ser modificadas ou inseridas. Para isso, os participantes precisarão chegar a um acordo (denominado de consenso) se essa informação é válida. No segundo caso, a informação é armazenada nos computadores dos participantes, evitando a centralização física da mesma.

- **Disponibilidade**

  A disponibilidade da informação está muito relacionada à descentralização física. Do momento que a informação encontra-se armazenada nos computadores dos participantes, ela torna-se disponível para ser utilizada independente de se um deles sair do sistema. Por exemplo, se existirem dez computadores que permitem o acesso à mesma informação, nove deles poderiam sair do sistema e a informação ainda estaria acessível pelo computador que restou. Por outro lado, se a informação está centralizada em somente um computador, caso esse computador tenha alguma falha ou saia, ninguém mais poderá ter acesso à informação.

  O conceito utilizado para fornecer a disponibilidade é de replicação da informação. Nesse sentido, a mesma informação precisa estar replicada e distribuída em vários computadores. Como mencionado anteriormente, a Blockchain realiza a replicação utilizando a rede *peer-to-peer*. Além da replicação, caso novas informações sejam inseridas, é necessário que todas as réplicas sejam sincronizadas. Para isso, são utilizadas técnicas de consenso distribuído, que serão explicadas na Seção 2.5.8.

- **Privacidade**

  A privacidade permite que todas as operações na Blockchain, denominadas de transações, possam ser realizadas de forma anônima, evitando que terceiros conheçam exatamente que pessoa (ou instituição) a realizou. Para isso, são utilizadas técnicas de criptografia, que permitem que uma pessoa (no mundo real) possa ser identificada (na Blockchain) somente através de um número. Nesse sentido, um terceiro, mesmo tendo acesso a toda a Blockchain, somente visualizará as operações feitas identificadas por números, sem saber quem é a pessoa ou instituição por trás deles.

  Um ponto importante a mencionar é sobre a obtenção desse número, que corresponde à pessoa ou instituição. Basicamente existem duas abordagens para obtê-lo. A primeira, utilizando uma autoridade certificadora, quem verifica que a pessoa ou instituição, no mundo real, realmente existe. Essa abordagem é utilizada nas Blockchains denominadas privadas, como o Hyperledger, detalhada na Seção 2.10, onde somente algumas pessoas têm acesso ao sistema. A segunda, utilizando uma autoridade certificadora quem não verifica se você realmente existe. Esse conceito pode parecer abstrato, mas pense que, no mundo virtual, as pessoas podem se passar por outras ou até por entes imaginários. Essa abordagem é utilizada nas Blockchains denominadas públicas, como o Ethereum, detalhada na Seção 2.10, onde qualquer pessoa tem acesso ao sistema.

- **Integridade**

  Integridade é um ponto importante dentro da Blockchain. Note que, como a informação pode estar distribuída (replicada) em vários computadores, é necessário confiar em que essa informação é íntegra, ou seja, que não foi alterada por ninguém. Mas, como confiar nas informações que alguém está apresentando se, baseado no ponto anterior da privacidade, você não necessariamente conhece a pessoa ou instituição na vida real?

  Nesse sentido, a Blockchain utiliza o conceito de cadeia, que permite criar um enlace entre as informações. Por quê? Fazendo uma analogia com uma corrente da

vida real, uma pessoa consegue identificar facilmente que houve uma quebra se um elo for rompido; isso significa que a corrente não está íntegra.

Agora, como criar enlaces de informações? Na Blockchain, cada elo corresponde a um bloco de informações. Esse bloco será criado com um identificador único entre todos os blocos que já existem ou que serão criados posteriormente. Dentro desse bloco, serão inseridas as operações (transações) realizadas pelos usuários. O elo entre dois blocos é realizado fazendo com que um bloco contenha um identificador do bloco anterior, formando assim o enlace. Note que se alguém quiser quebrar a integridade de uma cadeia de blocos X-Y-Z (por exemplo quebrando Y), deverá criar um novo bloco W que aponte para o bloco X e fazer com que o bloco Z aponte para o novo bloco W. O problema dessa abordagem é que a criação de blocos custa muito tempo, poder computacional ou energia elétrica, o que evita na prática que possa ser realizado em um tempo adequado, como será explicado na Seção 2.5.8.2.

- **Imutabilidade**

  A imutabilidade na Blockchain refere-se a que as informações (sejam estas as transações contidas em um bloco, ou as informações do cabeçalho do bloco) não poderão ser alteradas a partir do momento que forem inseridas na cadeia.

  Agora, como é possível realizar a imutabilidade se as informações variam? Por exemplo, o saldo de uma pessoa pode variar no dia (pelas operações de crédito e débito de um determinado montante), o prontuário de uma pessoa pode variar no tempo, etc.

  Para permitir a alteração de uma informação, a Blockchain cria um novo bloco, inserindo essa alteração como uma nova transação. Note que com isso, a cadeia conterá todas as modificações realizadas na informação, como se fosse o histórico completo, e não somente o último estado dela.

  Para o exemplo do saldo de uma pessoa, imagine que uma pessoa denominada A acabou de entrar no sistema. Nesse momento é criada uma transação de ingresso. A seguir, uma pessoa B envia 10 unidades para A. Nesse momento é criada uma nova transação onde A recebe 10 unidades. Finalmente, A envia para B 3 unidades. Nesse momento é criada uma nova transação onde a A envia 3 unidades. Note que todas as operações realizadas são inseridas na Blockchain, bem diferente a ter somente o último estado do saldo final de A, que seriam 7 unidades.

  A imutabilidade, que não permite a alteração ou remoção das informações, nem sempre é algo desejado. Por exemplo, suponha um sistema de saúde baseado em Blockchain, que adiciona dados aos prontuários dos pacientes. Em um determinado momento, um paciente pede (amparado pela lei) para remover todas suas informações. Note que a imutabilidade evita que isso aconteça, levando à clínica a ter possíveis problemas legais se as mantêm.

- **Auditabilidade**

  A auditabilidade na Blockchain permite verificar, por qualquer um que possua a cadeia, que todas as informações contidas nela são válidas. A auditabilidade faz uso das características apresentadas acima, como privacidade, integridade, entre outras.

Para verificar se as informações são válidas, é necessário verificar que tanto os blocos quanto as transações são válidas. No primeiro caso, é necessário validar que cada bloco aponte para o bloco anterior, até chegar ao primeiro bloco gerado, seguindo o enlace explicado na integridade de blocos e que será detalhado na Seção 2.5.

Já no segundo caso, é necessário verificar se todas as transações de uma determinada pessoa ou instituição (pelo anonimato, somente será mostrado um identificador) apresentam coerência no contexto em que está sendo utilizada a Blockchain.

Por exemplo, a coerência no contexto de operações financeiras está relacionada com os fundos para realizar compras de uma determinada pessoa. Quem realizar a auditoria pode verificar se, em um determinado momento, a pessoa tinha saldo suficiente, utilizando a condição de imutabilidade apresentada anteriormente.

2.4. Revisão histórica sobre o desenvolvimento de Blockchain

A tecnologia de Blockchain foi popularizada com a criação da criptomoeda Bitcoin. Assim, a tecnologia denominada de primeira geração, nasce como base tecnológica para realizar transações financeiras de débito e crédito de moedas virtuais entre pessoas. Seguindo o crescimento do mercado de moedas digitais, a Blockchain evolui como suporte para realizar transações em qualquer contexto, não somente financeiros. Nesse sentido, a tecnologia denominada de segunda geração, permite a inserção de funcionalidades (contratos inteligentes), que visam o cumprimento das normas de negócios a serem aplicadas nessas transações. A seguir serão detalhadas as duas gerações.

2.4.1. O mercado de moedas digitais - 1ª geração

Os primeiros estudos sobres moedas digitais datam do início da década de 90 com o movimento *Cypherpunk* [20]. Criptógrafos eram os principais integrantes deste movimento que lutou pela liberdade de ação dentro da internet desde seu início. Neste período surgiram os primeiros projetos de moedas digitais utilizando criptografia como base, entretanto estes projetos não conseguiram atingir um grande público e foram descontinuados por problemas de segurança.

Em 2008, Nakamoto [3] propõe o uso da estrutura Blockchain como base para um sistema de intercâmbio de dinheiro eletrônico, assim nascia o Bitcoin. Criado em 2009, o Bitcoin foi a primeira moeda digital a utilizar Blockchain para fins monetários em larga escala. Utilizado com o objetivo de manter a confiança entre as partes, a principal função da Blockchain no Bitcoin é armazenar, validar e distribuir as transações de valores realizada entre as mesmas.

Um desafio que as moedas digitais enfrentaram foi o problema conhecido na computação como gasto duplo. Este problema aborda a dificuldade de se garantir que um certo dado digital seja multiplicado de maneira indiscriminada; em outras palavras, evita que uma mesma transação seja utilizada mais de uma vez. Servindo como base para transações financeiras, a Blockchain do Bitcoin conseguiu resolver este problema empregando os conceitos mencionados anteriormente no capítulo de fundamentos e explorados na Seção 2.5.8. Este foi um dos principais motivos da criptomoeda conseguir ganhar a confiança dos usuários e assim expandir sua utilização ao redor do mundo.

A moeda digital Bitcoin pode ser considerada uma aplicação que funciona sobre a Blockchain. Em seu protocolo está descrito todas as funcionalidades e regras, entre elas o número máximo de moedas a ser criado: 21 milhões de unidades. Para entender como são criadas estas moedas é necessário falar sobre o procedimento denominado mineração.

Definido por Nakamoto como *Proof of Work* (PoW ou prova de trabalho), a mineração se dá pela resolução de um problema matemático com alto custo de processamento. O computador responsável pela resolução deste problema (denominado minerador) recebe moedas como recompensa, além de ganhar o direito de criar um novo bloco, que contém as transações. Além da criação, também é uma função do minerador a de validar estas transações e verificar se todos os usuários possuem os saldos transferidos.

Em seu início, a recompensa financeira pela resolução da prova de trabalho era de 50 Bitcoins, cujo valor cai pela metade a cada 4 anos até se esgotar a quantidade de total de moedas proposta. Estima-se que esta quantidade máxima seja atingida no ano de 2140.

No Bitcoin, para armazenar e movimentar um saldo é necessário que os usuários utilizem uma carteira digital (denominada *wallet*). As carteiras digitais podem ser *softwares* para computadores, aplicativos para dispositivos móveis ou *hardware* e sua principal função é armazenar uma chave privada que será utilizada para assinar as transações. Além da chave privada a carteira também armazena chaves públicas, estas chaves são o endereço de recebimento que o usuário deverá utilizar para solicitar uma transação a outro usuário.

Antes de ser validada e inserida na Blockchain, uma transação em Bitcoin fica aguardando a resolução da prova de trabalho para que um novo bloco seja criado. Definido em seu protocolo, este tempo é aproximadamente 10 minutos. Após uma transação ser inserida na Blockchain, a cada novo bloco inserido esta transação recebe novas confirmações. No Bitcoin, quanto mais confirmações uma transação receber maior é a segurança de que esta não será revertida. Um dos motivos para uma transação ser revertida é quando dois ou mais blocos válidos são criados simultaneamente, mas apenas um destes blocos fará parte da cadeia de blocos. A importância da confirmação da transação será abordada na Seção 2.5.8.2.

2.4.2. Contratos inteligentes - 2ª geração

A Blockchain de primeira geração foi aplicada principalmente para executar transações financeiras, onde um valor é transferido de uma pessoa para outra. Nesse sentido, os computadores responsáveis por manter a coerência da Blockchain, isto é, de inserir informações nela, cuidam de que a pessoa que está transferindo o dinheiro realmente possua o saldo suficiente para realizar a transação. Um ponto importante a destacar é que essa funcionalidade de verificação está inserida de forma estática dentro de cada um dos computadores que cuidam da Blockchain.

Aproximadamente em 2015, surgem no cenário da Blockchain algumas ferramentas, como Ethereum [9] e Hyperledger [10], que permitem inserir novas funcionalidades de forma dinâmica. O interessante dessa abordagem é que as funcionalidades não precisam ser somente para verificar um saldo (no contexto financeiro), mas podem ser para qualquer funcionalidade de negócio, por exemplo, verificar se um lote de remédios atin-

giu a data de vencimento e lançar um alerta. Nasce assim a segunda geração da Blockchain, onde o foco está nas funcionalidades, também denominadas de contratos inteligentes (*smart contracts*, em inglês, proposto conceitualmente por Szabo em 1996 [21]).

O contrato inteligente é análogo a um contrato em papel firmado por pessoas. No contrato em papel, são definidas as regras que estabelecem as responsabilidades e comunicação entre as partes que a assinaram. Na Blockchain, a diferença é que o contrato é digital, porém são mantidos os mesmos preceitos do contrato em papel.

No contexto da computação, as regras que estabelecem as responsabilidades que devem ser realizadas pelas partes são denominadas de regras de negócios, ou funcionalidades do sistema. Uma regra de negócio, nesse contexto, conterá uma sequência lógica de passos que serão transformados e implementados em um código executável utilizando alguma linguagem de programação.

Para dar um exemplo mais concreto, suponha que, no contexto de um software de gerenciamento de um hospital, o diretor pede para criar uma funcionalidade que verifique se o lote de um determinado medicamento está vencido, alertando-o dessa situação. A sequência lógica de passos para essa funcionalidade seria:

1. Obter os lotes gerenciados pelo hospital;
2. Recuperar as informações do lote, dentre elas o nome do medicamento, a data de vencimento e o email do diretor;
3. Verificar se a data de vencimento é maior que a data atual;
4. Se for maior, enviar um email para o diretor e gerentes, avisando da situação.

Para alguém que trabalha com sistemas informatizados, a funcionalidade descrita acima será desenvolvida em alguma linguagem de programação (por exemplo, Python, Java, etc.) e implantada no software de gerenciamento do hospital. Então, imagine que o desenvolvedor, por alguma razão, modifica o passo 4 da funcionalidade com a seguinte regra: "*se a data de vencimento for maior, não envie o email ao diretor*". Após a modificação, a regra é implantada no sistema. Note que o diretor (no passo 4) nunca ficará sabendo do vencimento do lote, mesmo que inicialmente foi ele quem solicitou a funcionalidade. Nesse sentido, o contrato foi quebrado sem uma das partes, no caso, o diretor, ter ideia disso.

Eis onde entra o contrato inteligente. De acordo ao mencionado na Seção 2.3, uma das características de Blockchain é a imutabilidade, que permite que uma informação, uma vez inserida na Blockchain, não possa ser alterada. Nesse sentido, agora imagine que a funcionalidade de vencimento foi inicialmente implementada e inserida na Blockchain (ou seja, como se ela fosse uma transação). Note que a funcionalidade não poderá ser alterada a não ser que uma nova transação (com o novo código modificado) seja inserida na Blockchain. O importante é que o diretor poderá observar tanto a funcionalidade inicial quanto a modificada, podendo realizar a auditoria desta. Nesse sentido, a funcionalidade fica transparente para todas as partes que a utilizam.

Atualmente, existem mais de 2100 criptomoedas virtuais parecidas ao Bitcoin, onde dado o interesse pelo investimento, centenas delas apareceram em questão de meses [22].

2.5. Como funciona a Blockchain?

Nesta seção serão apresentadas as diferentes tecnologias que compõem a Blockchain e como estas interagem para permitir seu bom funcionamento. Para isso, primeiro será dada uma visão geral de como funciona. A seguir, será explorado o conceito de bloco, transação e de como a cadeia formada por estes formam a base da Blockchain. A partir daí, será explicado alguns conceitos de criptografia e *hash*, que permitiram aumentar a segurança e eficiência da Blockchain. Finalmente, será descrito o conceito de consenso, ou seja, de como os computadores que fazem parte da Blockchain podem chegar a um acordo de quais blocos são os válidos.

2.5.1. Visão geral

Nesta seção será construído passo a passo o funcionamento da Blockchain através de dois cenários, o primeiro no contexto de uma compra de um bem entre duas pessoas e o segundo no contexto da área da saúde.

No primeiro cenário, imagine uma pessoa que compra um determinado bem de outra através de um meio eletrônico, como um sítio Web. Para que ambas tenham certeza de que a transação financeira ocorreu, ou seja, que o dinheiro foi enviado por uma e recebido pela outra, será necessário que algum computador armazene essa transação.

No segundo cenário, imagine uma médica que atende um paciente e registra essa consulta utilizando um prontuário eletrônico. Para que ambas as pessoas possam visualizar essa informação, também será necessário que algum computador armazene o prontuário.

Na Figura 2.2 é possível visualizar conceitualmente um bloco, que dentro contém um cabeçalho e a transação. Note que o conceito de transação é bem geral, podendo ser tanto as informações de um movimento financeiro quanto às informações de um prontuário do paciente.

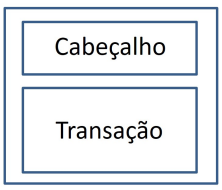


Figura 2.2. Bloco

Agora, o paciente se consulta uma segunda vez com a mesma médica. Após o atendimento, ela registra as informações no prontuário do paciente, gerando um novo bloco, que por sua vez contém uma nova transação. Como mostra a Figura 2.3, nesse novo bloco, a transação 2 fará parte do prontuário do paciente, com a alteração realizada pela nova consulta. Note que o segundo bloco aponta para o primeiro, criando realmente

uma cadeia interligada de blocos.

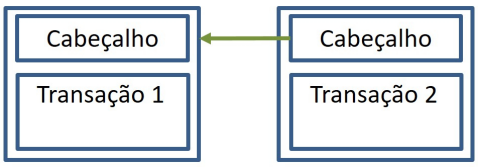


Figura 2.3. Cadeia de Blocos

Nesse momento é necessário fazer duas perguntas: (1) o que acontece se o computador que registrou as transações não tivesse mais acesso à internet ou se seu disco rígido onde estavam armazenadas as transações queimasse? (2) o que acontece que se alguma pessoa, que não tem permissão, modifica as informações do prontuário?

No primeiro ponto, há um problema de disponibilidade da informação. Ou seja, pode ser que as transações se percam e não consigam ser recuperadas. No segundo ponto, a informação está disponível, porém não está íntegra, ou seja, não é confiável.

Para resolver essas questões, uma das alternativas seria replicar a cadeia de blocos em diversos computadores. Assim, se um computador ficar indisponível, outro computador poderia tomar seu lugar. Já no caso de uma informação modificada, as outras réplicas poderiam verificar se houve alguma fraude e tentar chegar a um consenso. Esses casos serão analisados na Seção 2.5.8.

Assim que um computador tiver os blocos, será necessário que este seja capaz de analisar se os blocos e as transações contidas nos blocos, são válidos ou não. Nesse sentido, quais seriam as informações que cada bloco deveria ter para realizar a validação?

2.5.2. Bloco e cadeia de blocos

Como mencionado, a Blockchain é composta por uma cadeia interligada de blocos, que por sua vez contém uma ou mais transações. Agora, faz-se necessário entender quais são as informações que compõem o bloco.

O bloco é uma estrutura composta por dois módulos: cabeçalho e a lista de transações. O cabeçalho consiste em diversos metadados que identificam unicamente o bloco. Já a lista de transações identificam as transações realizadas e contidas nesse bloco. Por simplicidade, nesse texto, representamos a lista de transações sempre com apenas uma transação, mas lembramos que as listas podem conter dezenas ou centenas de transações.

Na Tabela 2.1 pode-se observar os principais campos que determinam o cabeçalho de um bloco junto com sua funcionalidade. Os campos *Merkle Root*, *Difficulty Target* e *Nonce* serão explicados com mais detalhes nas Seções 2.5.5, 2.5.6 e 2.5.8, respectivamente.

Com as informações da tabela, vamos criar os blocos do segundo cenário. Imagine que no primeiro atendimento, realizado às 8.30, um bloco 1 foi criado com uma transação (no caso, o prontuário eletrônico do primeiro atendimento). No segundo atendimento, realizado às 19.30 do mesmo dia, um novo bloco foi criado com a segunda transação (no caso, a adição de um novo prontuário eletrônico). Na Figura 2.4 pode-se observar

Tabela 2.1. Cabeçalho do Bloco.

Nome	Funcionalidade
Previous Block Hash	Apontador para o cabeçalho do bloco anterior.
Merkle Root	Número único que determina as transações que existem no bloco.
Timestamp	Data e hora aproximada da criação do bloco
Difficulty Target	Nível de dificuldade na criação do bloco
Nonce	Número que determina como foi criado o bloco

algumas informações do cabeçalho que o sistema gerou para esse segundo bloco 2.

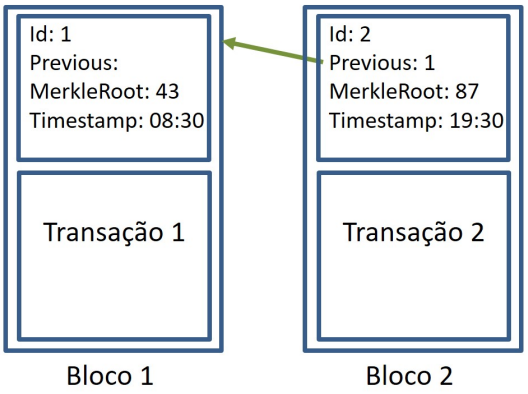


Figura 2.4. Cadeia de Blocos com cabeçalho

No bloco 2, o valor do *Previous Block Hash* corresponderá ao valor que identifica unicamente o bloco 1 (na Seção 2.5.5 será visto como é criado esse valor). Além disso, o sistema gerará um valor que determinará as transações que existem no bloco (Merkle Root) e a hora em que foi criado o bloco, isto é às 19.30 (timestamp).

Nesse momento deve surgir uma questão. Para quem aponta o bloco 1 no seu campo *Previous Block Hash*? Intuitivamente, deve apontar para um bloco anterior. Porém, deve existir algum bloco que não aponte para um bloco anterior, representando assim o começo da cadeia. Esse bloco inicial é chamado de bloco gênesis.

Em um sistema Blockchain, o primeiro bloco da cadeia é denominado de bloco gênesis e todos os computadores que façam uso da Blockchain devem conhecê-lo. Nesse sentido, se a partir de qualquer bloco se retrocede para o anterior, e deste para o anterior, utilizando o valor do campo *Previous Block Hash*, finalmente chegará ao bloco gênesis.

Até aqui, o exemplo do segundo cenário poderá ser complementado com o bloco gênesis, como mostra a Figura 2.5. Mais informações técnicas sobre o bloco podem ser encontradas em [23].

2.5.3. Transação e cadeia de transações

Dentro de cada bloco estão inseridas as transações realizadas pelos usuários do sistema. Um transação permite mostrar que um determinado item (seja este um valor monetário,

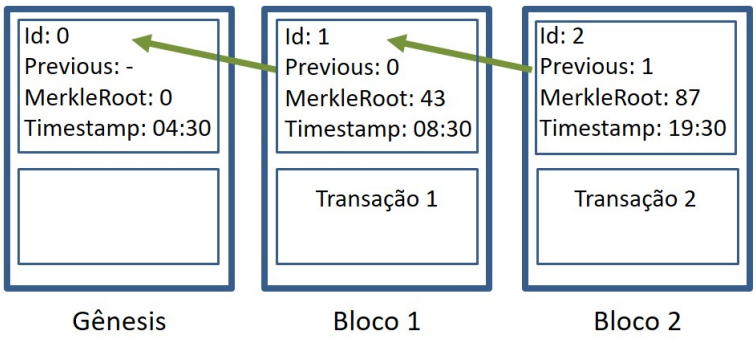


Figura 2.5. Cadeia de Blocos com gênesis

um documento, uma permissão de acesso, etc.) foi autorizado por um certo dono e disponibilizado para outro.

Conceitualmente, uma transação pode ser enxergada como um evento que ocorreu no sistema. Nesse sentido, no primeiro cenário, o evento seria uma transação financeira entre duas pessoas. Já no segundo cenário, o evento seria a inserção do registro eletrônico do paciente e a permissão de visualização da médica para o paciente.

De forma geral, uma transação é composta por um identificador único da transação e dois módulos: entradas e saídas. A entrada representa o identificador do dono que está realizando a transação e a saída representa o identificador do dono que está recebendo a transação. A Figura 2.6 abaixo mostra como seria a transação de uma transferência financeira de 5 unidades da pessoa com identificador ID1 para uma com identificador ID2.

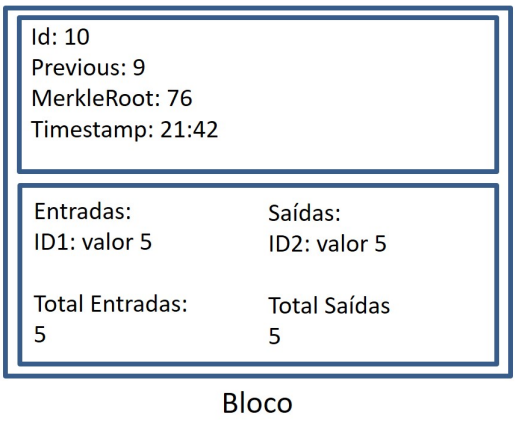


Figura 2.6. Transação

Entretanto, como saber se ID1 realmente tinha posse do item transferido (no exemplo acima, as 5 unidades)? Nesse sentido, ao igual que os blocos, as transações precisam apontar para uma transação válida (isto é, já existente em algum bloco da cadeia de blocos). Com isso, caso alguém queira verificar se ID1 tem ou não o item, basta analisar todas as transações realizadas por esse identificador, utilizando para isso os apontadores das transações. Mais informações técnicas sobre as transações podem ser encontradas

em [24].

2.5.4. Chave pública e privada

Até aqui, sabemos que a Blockchain é composta por blocos interligados, que por sua vez é composto por uma ou mais transações também interligadas. Também foi mencionado que a transação contém identificadores de usuários do sistema que a autorizaram e efetivaram. Por outro lado, como um computador pode validar que o identificador realmente é da pessoa que fez a transação e não de alguém se passando por ela?

Para explicar essa validação, será necessário entender antes alguns conceitos advindos da área de criptografia, cuja responsabilidade, entre outras, é a de segurança da informação. A seguir serão explicados esses conceitos através de um exemplo.

Imagine que você tem um texto que precisa enviar para alguém usando algum meio eletrônico, como um e-mail ou uma aplicação de bate papo. No envio, esse texto poderá passar por diversos computadores intermediários até chegar ao destino (é o que acontece normalmente na Internet). Para evitar intromissões, será necessário codificá-lo na origem de alguma forma, para que os intermediários não possam saber o que diz a mensagem, e decodificá-lo no destino, para recuperar a mensagem.

Na criptografia, a codificação na origem é denominada encriptar e a decodificação no destino é denominada de decriptar. Já o texto codificado, que não possui nenhum significado para quem não souber decodificá-lo, é denominado de "texto cifrado". A Figura 2.7 mostra os conceitos aplicados ao texto "Paciente: João. Tuberculose: negativo".

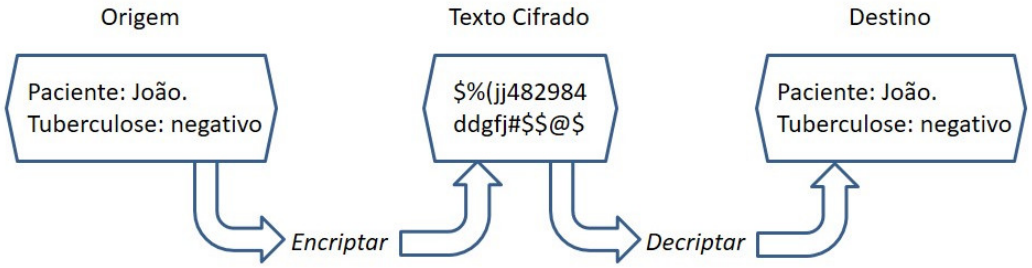


Figura 2.7. Aplicação da Criptografia

Agora, como é possível codificar o texto? Para isso, a criptografia utiliza um código secreto, denominado de chave (pense nela como se fosse o PIN do seu cartão de crédito) e o uso de um mecanismo de codificação. Dentro do mecanismo, existem duas alternativas a simétrica e a assimétrica. Na simétrica, se utiliza o mesmo código tanto para a codificação quanto para a decodificação, assim, origem e destino precisarão conhecer o mesmo código. Na assimétrica, se utilizam dois códigos diferentes, um para a codificação e outro para a decodificação. A seguir veremos o segundo caso, que é a alternativa utilizada pela maioria das Blockchains.

Na criptografia assimétrica existem duas chaves, a pública e a privada, que estão relacionadas entre si. O mais interessante dessa abordagem é a propriedade de que o texto encriptado por uma chave, somente pode ser decriptado pela outra. Como exemplo, veja o fluxo mencionado na Figura 2.8. Na figura, o texto "Paciente: João. Tuberculose:

negativo" é encriptado com uma chave privada (cor verde), transformando-o em um texto cifrado. A seguir, o texto cifrado é decriptado por uma chave pública (chave vermelha), recuperando o texto inicial.

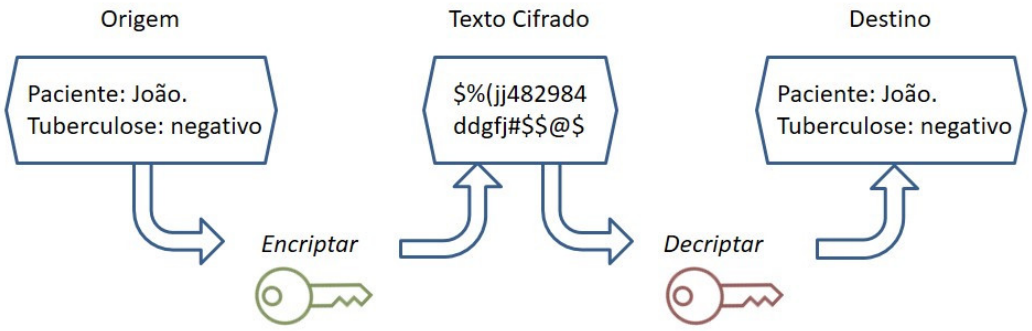


Figura 2.8. Fluxo de criptografia assimétrica

Na Figura 2.9 pode-se observar o fluxo inverso, onde o texto "Paciente: João. Tuberculose: negativo" é encriptado com uma chave pública (cor vermelha), transformando-o em um texto cifrado. A seguir, o texto cifrado é decriptado por uma chave privada (cor verde), recuperando o texto inicial.

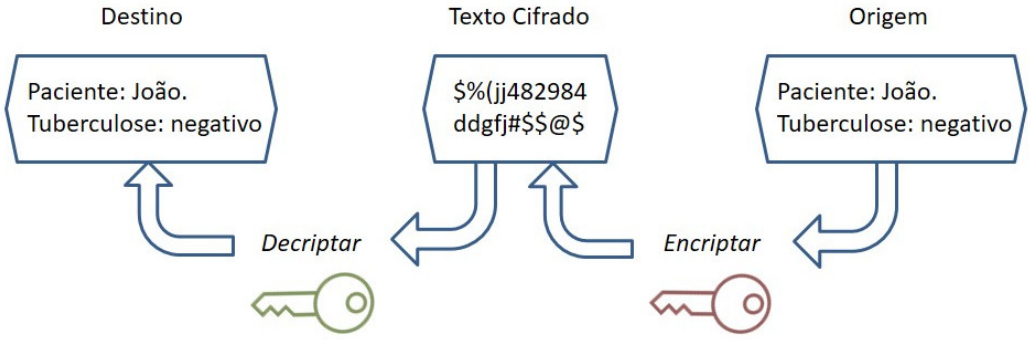


Figura 2.9. Fluxo inverso de criptografia assimétrica

Um ponto importante é que a chave privada permite identificar unicamente o dono emissor da mensagem, já que é a única pessoa que deveria ter posse dela (nesse sentido, essa chave não deveria ser compartilhada com ninguém). Do outro lado, o receptor da mensagem poderá identificar que essa mensagem realmente corresponde ao dono, dado que a única forma de decriptá-lo é utilizando a chave pública relacionada com essa chave privada. Mais informações técnicas sobre a criptografia podem ser encontradas em [25].

Finalmente, cabe mencionar que a geração da chave privada, e sua correspondente chave pública, deverão ser realizadas por uma entidade reconhecida, com a qual os usuários do sistema tenham confiança. Na vida real, por exemplo, diversas empresas realizam essa atividade, como a Verisign, Digicert, entre outros. Já na Blockchain, existe um componente de software que realiza essa ação, como veremos mais à frente.

2.5.5. Função de Hash

Como mencionado até agora, um usuário pode criar uma transação utilizando sua chave privada, inseri-la dentro de um bloco, concatená-la com um bloco anterior e replicá-la em outros computadores. Os outros computadores, por sua vez, obterão esses blocos e verificarão se a informação contida é válida ou não.

Entretanto, note que quanto mais transações hajam, mais informação precisará ser validada, chegando a um ponto em que precisará-se muito tempo para realizar essa ação. Não haveria uma forma de validar se a informação está correta somente comparando dois números, independente do tamanho da informação?

Novamente a criptografia entra em ação, provendo um mecanismo de compactação da informação que possui as seguintes características: determinístico, evita conflitos e de uma via [26].

Para entender essas características, imagine que precisamos compactar inicialmente o texto "Bloco 101 com 1 transação advinda do usuário João". Vamos supor que o mecanismo compactou o texto para "B101-1-J". O determinismo está relacionado com o fato de que a compactação do texto inicial sempre terá como resultado o mesmo valor "B101-1-J".

Já o evite de conflitos, denominado de colisão, está relacionado com que diferentes textos não deveriam ter como resultado o mesmo valor. Assim, por exemplo o texto "Bloco 101 com 1 transação advinda do usuário Joã" (sem a vogal 'o' no final) deveria criar um texto compactado completamente diferente.

Finalmente, a característica de uma via, permite que não seja possível recuperar o texto original a partir das informações do texto compactado. Nesse sentido, "B101-1-J" não cumpriria essa condição, pois alguém poderia entender que 'B101' corresponde ao número do bloco, 1 corresponde à quantidade de transações e 'J' corresponde ao nome de algum usuário.

O mecanismo geralmente utilizado pela Blockchain para compactar a informação é denominado de *hash*, que permite compactar um texto de qualquer tamanho em outro de tamanho fixo. Existem diversas implementações que realizam a compactação, como por exemplo o MD5, SHA1, entre outras. A seguir será dado um exemplo com MD5, para entender as características mencionadas anteriormente.

Para o texto "Bloco 101 com 1 transação advinda do usuário João" (sem aspas), o MD5 o compacta no seguinte código: 0F1D18186FD5E1C9072627CC9677446E. Independente de quantas vezes aplicar o MD5 no texto, será obtido o mesmo código anterior, corroborando o determinismo. Agora, para o texto "Bloco 101 com 1 transação advinda do usuário Joã" (sem aspas e sem a vogal 'o'), o código obtido será bem diferente: 8899FA17AE7A802024D96E101C85B0FC. Veja que, mesmo com um pequena alteração, ele é completamente diferente do anterior, corroborando a ideia de evitar conflitos ou colisões. Finalmente, note que para o código obtido, não há sequer uma dica de como obter o texto antes de ser compactado, corroborando a característica de uma via. Mais informações técnicas sobre as funções de *hash* podem ser encontradas em [27, 25].

2.5.6. Árvores de Merkle

Uma árvore na computação pode ser descrita como uma estrutura composta por um conjunto de nós ligados, que começam em um nó (denominado raiz) e terminam em um ou mais nós (denominados folhas). O mais interessante dessa estrutura é que para chegar do nó raiz até uma folha, somente haverá um caminho possível de ser percorrido. Além disso, cada nó não folha (denominado pai) poderá ter um ou mais nós (denominados filhos) A Figura 2.10(a) mostra uma árvore numerada desde o nó raiz (com o número 1), onde cada nó tem dois possíveis filhos (direita e esquerda) até os 4 nós folha (com os números 4 ao 7).

(a) Árvore. (b) Árvore de Merkle.

Figura 2.10. Exemplos de Árvores.

A árvore de Merkle [28] é uma árvore bastante utilizada na Blockchain para conferir que as transações inseridas dentro de um bloco são válidas [3]. Antes de entender como a Blockchain usa a árvore de Merkle, vamos analisar como esta é construída. Imagine que um bloco possui 4 transações, cada um com seu identificador único. Como mostra a Figura 2.10(b), primeiro, de cada identificador da transação será obtido o valor *hash* e adicionado à folha correspondente. Assim, a folha 4 terá um valor v_4 calculado com a função H(ID1). Já a folha 5 terá um valor v_5 calculado com H(ID2), onde H é uma função *hash*, como MD5. A seguir, cria-se o valor *hash* da concatenação de pares de transações e insere-se esse valor no nó pai destas. Para o exemplo, o nó 2 terá um valor v_2 calculado com a função $H(v_4v_5)$. O mesmo para o caso do nó 3 que terá um valor v_3, calculado com a função H para as transações ID3 e ID4. Finalmente, o nó raiz terá o valor v_1 através do *hash* da concatenação dos valores nos nós 2 e 3 (*i.e.*, $H(v_2v_3)$), formando assim a árvore de Merkle.

Como mencionado na seção anterior, sabemos que o cálculo do *hash* possui certas propriedades que permite que o cálculo do valor, usando a função, seja determinístico. No caso da árvore, sempre teremos o mesmo valor da raiz se começarmos com os mesmos valores dos identificadores nas folhas. Note que se qualquer transação das folhas tiver outro identificador, o nó raiz também terá outro valor.

Agora, onde a Blockchain utiliza a árvore de Merkle? Como mencionado na Tabela 2.1, um dos campos do cabeçalho que deve ser preenchido na criação do bloco é o Merkle Root, que determina as transações que existem no bloco. Note que o valor v_1 do

nó raiz da Figura 2.10(b), representa todas as transações, por ser uma combinação delas. Assim, na criação do bloco, o Merkle Root do cabeçalho será preenchido com o valor existente na raiz da árvore de Merkle.

2.5.7. Tipos de Blockchain

Segundo Buterin [29], criador do Ethereum, existem dois tipos de Blockchains: permissionadas e não permissionadas. Nas Blockchain não permissionadas, qualquer membro pode realizar modificações e auditar a cadeia. Já nas Blockchain permissionadas, somente membros autorizados podem realizar operações na cadeia. Além disso, é comum associar o termo Blockchain não permissionada a Blockchain pública e Blockhain permissionada a instâncias privadas, federadas ou em consórcio.

Na Blockchain não permissionada, qualquer entidade pode entrar e sair do sistema em qualquer momento. Ao entrar, a entidade transforma-se em um membro que poderá realizar modificações e auditorias na cadeia inteira de blocos. Como é possível que potencialmente cada membro possua a cadeia de blocos, nesse tipo de Blockchain há uma total descentralização da informação. Exemplos deles são Bitcoin [3] e Ethereum [9].

Na Blockchain permissionada, somente algumas entidades serão transformadas em membros do sistema e terão permissão para realizar operações na cadeia. Assim, algumas poderão ler os blocos, outras poderão escrever e outras poderão auditar. Para permitir a identificação e autorização dos membros, será necessário criar responsáveis (confiáveis) por gerenciar as permissões. Nesse tipo de Blockchain existem entidades que realizam o papel de autorizadores. Exemplos de Blockchain permissionada são as plataformas Hyperledger Fabric [30] e Corda [31].

Wüst e Gervais [32] propuseram uma subdivisão da Blockchain permissionada entre pública e privada. A divisão somente considera a auditabilidade, onde a *"Blockchain permissionada pública"* permite que qualquer membro possa verificar os dados da cadeia e na *"Blockchain permissionada privada"* somente é permitida a verificação para um conjunto bem definido e autorizado de membros.

As aplicações Blockchain que requerem identificação de usuários tendem a ser construídas usando infraestrutura permissionada. Por outro lado, estruturas não permissionadas tendem a oferecer maior anonimato. Outra questão importante para a escolha de tipo de Blockchain é a criação e manutenção da infraestrutura que suporta a rede de nós. Uma Blockchain privada normalmente é de responsabilidade de uma instituição que a mantém operante; nesse sentido, as Blockchains públicas ou federadas concentram menos o poder de decisão sobre a rede.

2.5.8. Consenso

Lembrando que uma cadeia de blocos é replicada em diferentes computadores por questões de disponibilidade (caso um dos computadores saia do sistema, outros poderão tomar seu lugar), veja a seguinte situação que pode acontecer.

Imagine que em um primeiro momento os três computadores A, B e C da Figura 2.11 possuem a mesma cadeia de blocos 0 e 1.

Como mostra a Figura 2.12(a), em um segundo momento, o computador D envia

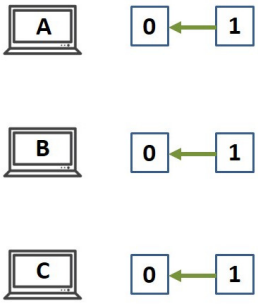
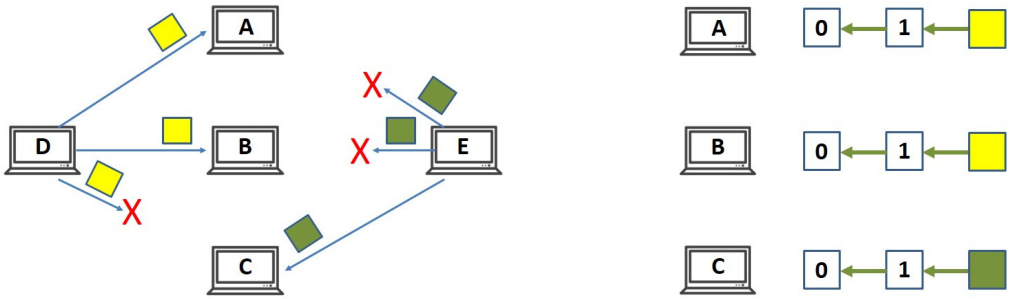


Figura 2.11. Consenso inicial

uma mensagem para que A, B e C adicionem o bloco amarelo, porém somente A e B a recebem. Em seguida, o computador E também envia a mensagem para que A, B e C adicionem o bloco verde, porém somente C a recebe. Uma pergunta que pode surgir é, porque algumas mensagens não foram recebidas? Na Internet, muitas vezes as mensagens são perdidas, principalmente, por questões de congestionamento nos roteadores por onde passa a mensagem até chegar a seu destino. No final do segundo momento, pode-se observar na Figura 2.12(b) que os computadores A e B possuem o bloco amarelo e o computador C possui o bloco verde, ambos apontando para o bloco 1 (portanto, tanto o bloco verde quanto o amarelo são válidos).



(a) Recebendo blocos. (b) Adicionando blocos.

Figura 2.12. Consenso intermediário.

O consenso tem como objetivo que os computadores cheguem a um acordo sobre um determinado valor [27]. No caso da Blockchain, eles devem chegar a um acordo de qual bloco (verde ou amarelo) deverá ser adicionado no final da cadeia, ou seja após o bloco 1. No final do acordo, todos os computadores (no exemplo, A, B e C) deverão ter a mesma cadeia. A seguir veremos duas estratégias empregadas pela Blockchain para chegar ao consenso.

2.5.8.1. Processo de consenso via Paxos

No consenso via Paxos [33], somente alguns poucos computadores são os encarregados por realizar o processo de acordo, geralmente algumas dezenas destes. A escolha desses

computadores pode-se basear em diferentes características, tais como, poder computacional, tempo sem interrupções, largura de banda, entre outros. Para o exemplo, vamos supor que foram escolhidos 7 computadores.

Tendo os computadores que farão o consenso, o primeiro passo do processo é a escolha, dentre eles, de um líder, ou seja, um computador responsável por dirimir qual será o bloco a ser inserido no final da cadeia. Existem diversas alternativas para realizar a escolha, por exemplo, cada computador pode ter um número identificador único e o menor destes será escolhido como o líder. Caso os outros seis computadores do consenso enxerguem que o líder não está mais disponível (*e.g.* caiu), o computador com o segundo menor identificador será escolhido como o novo líder, e assim sucessivamente.

Tendo o líder, cada computador que não faz parte do consenso (denominado de cliente) pode propor um novo bloco para ser inserido na cadeia. O cliente pode propor a inserção de um bloco a qualquer dos sete do consenso, porém somente o líder poderá inseri-lo na cadeia. Nesse sentido, o computador do consenso que recebeu o bloco, redirecionará o pedido para o líder, caso não o seja.

O líder, por sua vez, receberá os pedidos, direto dos clientes ou dos redirecionamento, e dará uma ordem neles (geralmente, o primeiro pedido que chega é o primeiro pedido a ser atendido). Após a ordenação dos pedidos, adicionará os blocos nessa ordem, e replicará essa informação para os outros seis computadores do consenso.

Quando os computadores do consenso receberem a ordem dos blocos, inserirão nas suas respectivas cadeias, respondendo ao líder que conseguiram realizar a inserção. Finalmente, assim que o líder obtiver a maioria das respostas (por maioria, entenda-se à metade mais um, ou seja, quatro respostas), responderá ao computador cliente que seu bloco foi adicionado com sucesso.

Os detalhes do funcionamento do processo serão analisados na Seção 2.10.1 sobre o Hyperledger.

2.5.8.2. Processo de consenso via Proof of Work (POW)

Neste tipo de consenso, todos os computadores podem ser passíveis por realizarem o processo de consenso, diferente do Paxos onde somente alguns são os escolhidos. Para comportar os possíveis milhares de computadores, a escolha de um líder não é a mais adequada, haja vista que esse líder talvez não será capaz de lidar com todas as mensagens advindas de todos os computadores. Assim, será necessário criar um processo que não dependa somente de um computador.

A Blockchain utilizada no Bitcoin foi uma das primeiras técnicas, aplicadas em um sistema real, em possibilitar o consenso em milhares de computadores. O funcionamento dele é dado a seguir.

O processo começa com um computador (denominado minerador) obtendo de outro computador a cadeia de todos os blocos, com suas respectivas transações, que existe até esse momento. Para se ter uma ideia, o histórico em 2019 possui milhares de blocos, com um tamanho aproximado de 150 gigabytes. Para o exemplo, imagine que a cadeia

tem somente 100 blocos.

Em posse desse histórico, o minerador, que será denominado de M1, deve verificar que cada uma das transações é válida (utilizando o conceito de cadeia de transações mencionado na Seção 2.5.3) e que cada bloco é válido (utilizando o conceito de cadeia de blocos mencionado na Seção 2.5.2).

Tendo realizado as validações, o minerador M1 obterá novas transações que foram realizadas pelos clientes (e que não existem em nenhum bloco anterior), criando um novo bloco com essas transações.

Na criação do bloco é necessário preencher as informações do cabeçalho do mesmo. Primeiro, o campo 'timestamp' corresponde à hora do computador do minerador, por exemplo, 12/02/2019 13:30. A seguir, o campo '*Previous Block Hash*' deverá apontar para o bloco 100, calculado através do *hash* desse bloco 100, por exemplo usando o MD5 explicado na Seção 2.5.5. '*Difficulty Target*' e 'nonce' são números que o minerador deverá utilizar para provar aos demais mineradores que realmente foi realizado um trabalho computacional para criar o bloco.

O trabalho é realizado da seguinte maneira: '*Difficulty Target*' é um número calculado pelo sistema e gerado aproximadamente a cada duas semanas. Esse número, que permite que cada duas semanas sejam criados no máximo 2016 blocos, geralmente começa com uma quantidade de zeros, por exemplo: 000101827749837 (começando com 3 zeros). A seguir, o minerador precisará encontrar um número menor que o '*Difficulty Target*', obtido através do *hash* do bloco que está sendo criado. Porém, pense o seguinte, só com as informações do cabeçalho (timestamp, previous block *hash* e *Difficulty Target*) pode ser que o MD5 não consiga gerar um valor menor que o *Difficulty Target*'. Por exemplo, vamos supor que o MD5 do texto '12/02/2019 13:30 bloco100 000101827749837' dê o valor 100405682589837. Note que esse valor é maior ao 0001018277498372371. Eis onde entra o nonce. O nonce é um atributo do cabeçalho que permite ser modificado para que o *hash* seja menor ao *Difficulty Target*. Veja o exemplo na Tabela 2.2 abaixo para diferentes nonces, aplicados ao cabeçalho anterior.

Tabela 2.2. Aplicação de diferentes Nonces.

Nonce	Texto a ser usado no MD5	Resultado
0	'12/02/2019 13:30 bloco100 000101827749837 0'	100405682589837
1	'12/02/2019 13:30 bloco100 000101827749837 1'	320106680549244
2	'12/02/2019 13:30 bloco100 000101827749837 2'	000047479763563

Note que o computador teve que realizar três cálculos (trabalho computacional com os nonces 0, 1 e 2) para encontrar um número menor que o *Difficulty Target*. Nos sistemas reais de Blockchain, como Bitcoin, o computador realiza milhões ou bilhões de cálculos, dai o nome *Proof of Work*, ou prova de trabalho.

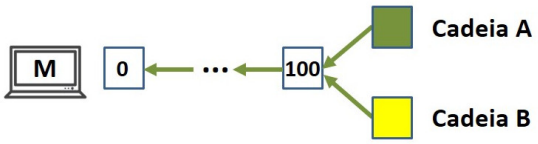
Após encontrar o nonce adequado, o minerador M1 o insere no cabeçalho e cria o bloco (denominemos esse bloco de amarelo, para efeitos ilustrativos). Após a criação do bloco amarelo, o minerador M1 dissemina essa informação a outros mineradores, denominado M. O consenso acontecerá em dois casos: (1) o minerador M que recebeu o bloco

amarelo de M1 também estava tentando criá-lo, e (2) o minerador M já tinha recebido um bloco verde, de outro minerador M2, que apontava para o bloco 100.

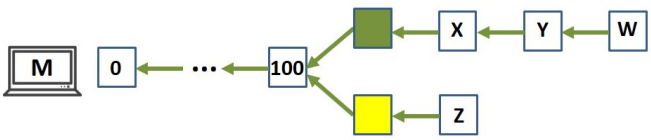
Para o primeiro caso, assim que o minerador M receber o bloco amarelo, imediatamente deverá verificar que o bloco recebido é válido (olhando que aponta para o bloco 100, por exemplo). Caso seja válido, M parará de criar seu bloco e adicionará o bloco amarelo no final da sua cadeia, começando a criação de um novo bloco, apontando para o amarelo.

Para o segundo caso, o minerador M tinha recebido um bloco verde válido que apontava para o bloco 100. Mas, como pode ter acontecido isso se o minerador M1 criou o bloco amarelo, também válido, nesse instante? Note que a criação de um bloco verde pode ter acontecido por um outro minerador M2 (nada impede isso) e ter sido disseminado antes que o bloco amarelo de M1.

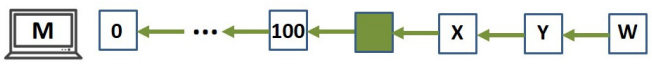
Agora, como mostra a Figura 2.13(a), M terá dois blocos válidos, um verde e um amarelo, ambos apontando para o bloco 100. O que fazer? A regra para chegar ao consenso será esperar a chegada de novos blocos e, depois de um certo tempo, escolher aquele que tenha a maior cadeia a partir do bloco 100. Imagine o seguinte caso, após um certo tempo, M recebe três novos blocos que contém o bloco verde, denominada cadeia A, e somente um novo bloco que contém o bloco amarelo, denominada cadeia B, como mostra a Figura 2.13(b). Finalmente, como a cadeia A é maior que a cadeia B, o minerador M descartará a cadeia B (que contém o bloco amarelo e o bloco Z), como mostra a Figura 2.13(c). Note que o consenso ocorrerá dado que a regra de somente continuar com a maior cadeia será seguida por qualquer minerador (inclusive M1).



(a) Duas cadeias.



(b) Recebendo blocos.



(c) Escolhendo a cadeia mais longa.

Figura 2.13. Consenso de qual bloco inserir.

Os detalhes do funcionamento do processo serão analisados na Seção 2.10.2.

2.6. Desafios para o uso da Blockchain

Zhang *et al.* [15] descrevem alguns desafios para a implantação de sistemas de informação baseados em Blockchain: *i.* capacidade de evoluir, *ii.* capacidade de armazenamento, *iii.* privacidade e *iv.* escalabilidade. Além desses, serão tratados alguns outros desafios como a interoperabilidade, a compatibilidade legal dos sistemas e o gasto de energia.

- A evolução é algo natural em sistemas de informação. Tanto a informação quanto as funcionalidades que as tratam sofrem alterações durante a vida do sistema, sejam por questões do projeto em si ou até por questões normativas (leis) que devem ser seguidas. Por exemplo, imagine que em um determinado momento todos os prontuários dos pacientes do SUS, armazenados na Blockchain, sejam identificados pelo CPF. Anos depois, cria-se uma lei que obriga aos sistemas a usarem, como identificador único, o número do cartão do SUS.

  A capacidade de evoluir refere-se ao suporte que entrega a ferramenta ou tecnologia utilizada para facilitar a evolução dos sistemas, diminuindo ao máximo as mudanças que devem ser realizadas. Note que no caso da Blockchain, todas as transações antes da aplicação da lei foram feitas usando o CPF. Nesse sentido, se quisermos adequar o sistema à nova lei, ou mantém-se na Blockchain a mistura de informações com CPF e o nº do SUS (lidando com possíveis inconsistências) ou elimina-se a Blockchain inteira e inserem-se novamente todas as transações que tinham o CPF, mas agora com o nº do SUS. Nesse sentido, é necessário ter extremo cuidado na escolha dos dados que vão compor a transação, para que seja flexível o suficiente para lidar com mudanças.

- O armazenamento permite a persistência da Blockchain nos computadores. Um sistema de saúde deve considerar que a Blockchain conterá informações sobre pacientes, médicos, prontuários, pagamentos, medicamentos, etc. Nesse sentido, grandes volumes de dados deverão ser armazenados nos computadores. Por outro lado, além do armazenamento, será necessário também considerar a infraestrutura de hardware (computadores, acesso à Internet, etc) e de software (sistema operacional, memória RAM, etc) que permitirá o acesso à Blockchain.

  A capacidade de armazenamento refere-se ao suporte que entrega a ferramenta ou tecnologia utilizada para facilitar o armazenamento da Blockchain. Por exemplo, usando o Ethereum (analisada na Seção 2.10.2) é possível abstrair todos os problemas, já que a rede oficial dessa tecnologia provê a infraestrutura e acesso. Entretanto, para usá-la, é necessário comprar GAS (unidade de medida similar ao kilowatt/hora da eletricidade) para executar as transações, incorrendo em gastos que é necessário considerar. Já usando o Hyperledger (analisada na Seção 2.10.1) é possível criar a rede entre os participantes do sistema, evitando os gastos por transação do Ethereum. No entanto, a criação e manutenção da rede também incorrerá em gastos que devem ser considerados.

- A privacidade da informação é um ponto sensível em qualquer sistema. Para o caso dos sistemas de saúde, alguns requisitos em privacidade incluem: autenticação dos participantes; armazenamento seguro, baseado em princípios de criptografia; controle de acesso às informações. Mesmo para sistemas considerados seguros, a cada

dia novos ataques tentam encontrar suas vulnerabilidades. Na Blockchain existe um risco maior. Primeiro, as informações são replicadas em todos os computadores que fazem parte da rede. Nesse sentido, caso no futuro o processo de encriptação seja comprometido, potencialmente todos os registros poderão ser lidos por uma pessoa (note que não há como evitar isso, haja vista que a pessoa, que faz parte da rede, poderá ter armazenado a Blockchain completa). Segundo, as funcionalidades (contratos inteligentes) implementadas e implantadas são abertas e possíveis de serem auditadas. No entanto, isso também gera um problema, caso uma pessoa descubra um erro de segurança na programação da funcionalidade, realizando ataques em todos os computadores que a usem.

- A escalabilidade, no contexto da Blockchain para saúde, refere-se tanto à quantidade de transações que o sistema permite realizar quanto às pesquisas que consegue responder dentro de um período de tempo. Como mencionado até agora, transações podem ser operações financeiras, registro de prontuários eletrônicos, cadastro de pacientes, entre outros. Nesse sentido, a ferramenta ou tecnologia deve ser capaz de suportar possivelmente dezenas de milhares de transações ou pesquisas em um curto espaço de tempo. Das duas ferramentas analisadas na Seção 2.10, Ethereum conseguiu realizar 1.349.890 transações no dia 4 de janeiro de 2018 (aproximadamente 14 transações por segundo). Já no Hyperledger foi possível realizar 1.7 bilhão de transações por dia (aproximadamente 20 mil por segundo) com modificações na arquitetura [34], provendo a escalabilidade necessária para atender os requisitos dos sistemas de saúde.

- O tempo de confirmação de uma transação pode variar entre as diferentes tecnologias Blockchain. Este tempo é o intervalo entre a criação da transação até o momento que a mesma é inserida em um novo bloco e distribuída entre os participantes. Na Blockchain utilizada pelo Bitcoin este intervalo é de aproximadamente 10 minutos. Com as melhorias implementadas na Blockchain do Ethereum e Hyperledger Fabric, este tempo foi reduzido para aproximadamente 15 segundos e 1 segundo, respectivamente. Este tempo deve ser observado com muito cuidado visto que, dependendo do contexto a ser utilizado, esta demora pode inviabilizar a sua aplicação.

- A interoperabilidade, refere-se à habilidade dos diferentes sistemas de informação de se comunicarem, transferirem e usarem informações [35]. A interoperabilidade é dividida em dois níveis: funcional, focada na interação entre sistemas usando regras de negócios; semântica, focada na compreensão do significado dos conceitos envolvidos na transferência.

  Por exemplo, imagine que um paciente se atende em um hospital do estado onde mora, mas em uma viagem a outro estado se atende em uma clínica privada. A interoperabilidade funcional proverá que o sistema da clínica possa ter acesso ao prontuário armazenado no sistema do hospital, desde que cumpridos os requisitos de privacidade da informação. Já a interoperabilidade semântica permitiria à clínica entender um significado específico de uma frase utilizada no contexto do hospital.

  Atualmente, alguns padrões de interoperabilidade de dados médicos (como HL7 e

FHIR [36]) provem as bases para intercambiar informações entre sistemas, entretanto, a implementação desses padrões ainda não é amplamente utilizada.

- A compatibilidade legal refere-se à capacidade do sistema se adequar aos regulamentos e leis existentes ou que possam ser criados no futuro [37]. Note que esse desafio está muito interligado à capacidade de evolução do sistema. Um exemplo para esse desafio é o artigo 17 do Regulamento Geral sobre a Proteção de Dados na Europa, que define o direito ao apagamento dos dados (o "direito a ser esquecido"). Entretanto, a tecnologia Blockchain não permite se adequar a essa lei de forma fácil, haja vista que as informações não podem ser eliminadas.

- Manter uma rede Blockchain em funcionamento requer uma quantidade de participantes *online* para validar e distribuir as transações. O gasto de energia para manter estes computadores ligados é um desafio a ser analisado. Em geral, as Blockchains públicas e não permissionadas possuem uma grande quantidade de participantes e o consumo de energia pode não ser sustentável (a rede Bitcoin[3] e Ethereum[4] possuem cerca de 8 mil participantes). Bitcoin, por ser baseado em solução de problemas matemáticos por força bruta, tem a sua sustentabilidade energética contestada [38]: em 2018, a energia elétrica utilizada para mineração foi superior ao consumo da Irlanda. Nas Blockchains privadas, por exemplo o Hyperledger Fabric, o consumo de energia ainda existirá, mas não é uma preocupação haja vista que normalmente utilizam algoritmos de consenso bizantino (que são mais baratos computacionalmente) e são poucos os computadores responsáveis por ele.

2.7. Vulnerabilidades de segurança da Blockchain

Todo sistema de informação está sujeito a ataques de segurança. Entre os ataques mais comuns estão a tentativa de quebra de senhas (quebra de segredos criptográficos) ou ataques de disponibilidade (DDOS, *Distributed Denial of Service*, em inglês). Blockchain, por sua natureza distribuída, pode estar exposto a ataques adicionais [39]. Na Blockchain, os três problemas de segurança mais discutidos são:

- **Ataque de 51%**. Apesar de conhecido pelo nome de 51%, na verdade, esse ataque é caracterizado quando uma única entidade (ou um arranjo de membros atuando como uma única entidade) detém uma fatia expressiva, ou a maioria, do poder computacional. Em uma rede Bitcoin, por exemplo, se uma mesma entidade detivesse a maioria do poder computacional, essa entidade poderia influenciar ou manipular a formação da cadeia de blocos. Em outras palavras, poderia influenciar a formação da cadeia mais longa de blocos para permitir, maliciosamente, o cancelamento de transações ou de decisões de consenso.

- **Gasto Duplo**. A tecnologia não permite a existência de gasto duplo; mas é estatisticamente possível que uma transação seja cancelada uma hora após ter sido registrada em um bloco. Em vendas no varejo, por exemplo, o cancelamento tardio de uma transação pode levar, na prática, ao não pagamento de uma transação.

[3] Vide https://bitnodes.earn.com/

[4] Vide https://ethstats.net/

- **Perda da Chave Privada**. Um recurso digital na Blockchain só pode ser alterado ou transferido usando a chave privada que o gerou. A perda dessa chave privada leva a perda permanente do recurso. Na rede Bitcoin, por exemplo, se um usuário perde a sua chave privada, suas moedas não podem mais ser movimentadas; esses recursos estarão perdidos para sempre.

Além dessas três preocupações mais comuns colocadas acima, um sistema Blockchain pode estar sujeito a ataques realizados para revelar a identidade de usuários e dados sensíveis. Esses ataques podem ser baseados em estratégias em engenharia social, que cruzem, por exemplo, registros Blockchain e hábitos individuais. Outro ataque relevante em aplicações que envolvem criptomoedas é o roubo de moedas por meio da exploração de falhas dos programas.

Por fim, uma preocupação cada vez mais importante é a corretude dos contratos inteligentes. Cabe lembrar que assim como uma transação, um contrato inteligente não pode ser modificado após sua escrita nos blocos. Encontrar um erro em um contrato inteligente significa: perder os valores gastos para sua publicação original, a necessidade de lançar uma nova aplicação e a respectiva migração de usuários. A corretude dos contratos inteligentes passa não apenas pela verificação de que os requisitos funcionais estão corretamente implementados, mas também que os usuários não possam explorar brechas que comprometam a rede (por exemplo, laços infinitos[5]) ou que permitam o roubo de dados e de chaves criptográficas.

O teste e validação de programas sempre foi importante para a ciência da computação, mas com o advento de contratos inteligentes essa importância ficou ainda maior. Em alguns casos, a não corretude de um programa pode ser diretamente associado a perda de recursos econômicos.

2.8. Será que preciso usar a Blockchain?

Apesar da Blockchain ter se tornado uma tecnologia muito popular nas criptomoedas, cabe perguntar-se se ela é aplicável a qualquer cenário. Segundo uma pesquisa realizada em 2018 pela PricewaterhouseCoopers, 84% de 600 executivos de empresas das mais diversas áreas estão de alguma forma envolvidos ativamente com a Blockchain [40]. No entanto, será que realmente essas empresas precisam utilizar a Blockchain ou poderiam utilizar tecnologias já consolidadas (como banco de dados) para realizar as mesmas funcionalidades de forma mais simples e até mais eficiente?

A seguir, será analisada uma metodologia, baseada no estudo de Wust [32], que permite identificar se faz sentido utilizar a Blockchain. Caso seja identificada a necessidade, será mostrado qual é o tipo da Blockchain (permissionada ou não) que deveria ser utilizada.

A Figura 2.14 mostra o fluxograma da metodologia, onde a circunferência representa uma pergunta a ser respondida, cada linha representa a resposta (sim S, não N) à

[5] A presença de laços infinitos em contratos inteligentes obviamente comprometem a capacidade computacional das redes. Na rede Ethereum, por exemplo, esse problema é contornado por meio do uso de GAS. Como todo tipo de execução computacional custa um certo volume de GAS, um laço só será executado enquanto houver GAS. Todo laço infinito, portanto, será executado apenas até o fim do GAS.

pergunta e o retângulo representa a resposta final (o tipo da Blockchain ou se não deveria ser utilizada). Para cada pergunta será dado um exemplo no contexto dos sistemas de informação na área de saúde.

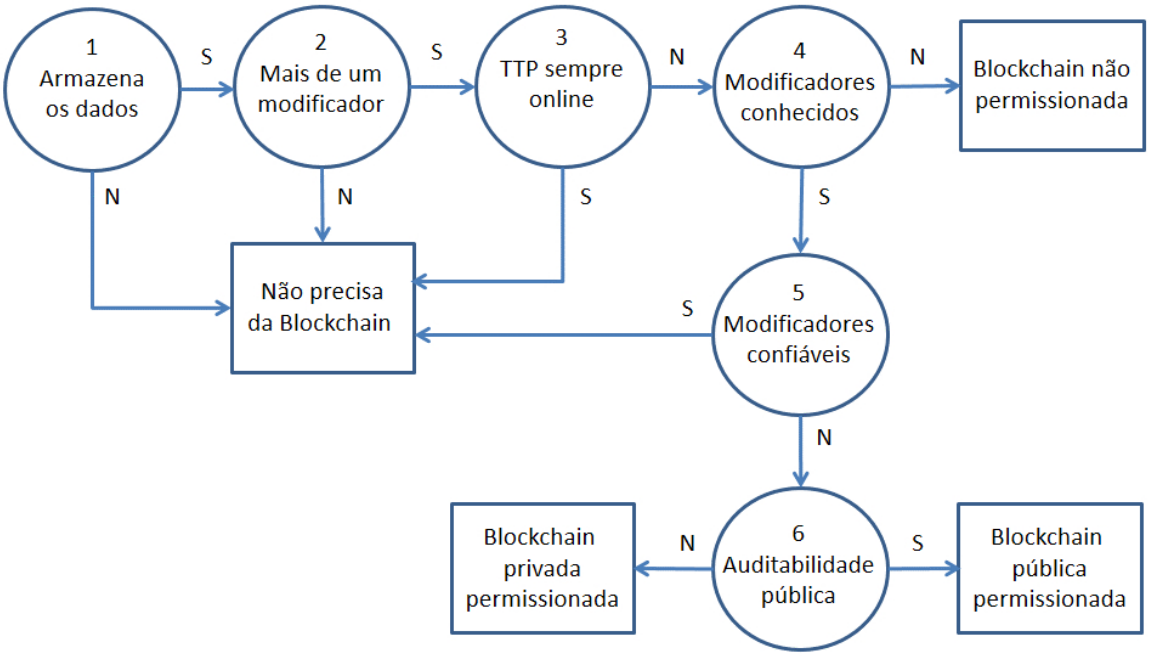


Figura 2.14. Fluxograma da Metodologia

A primeira pergunta a ser respondida é: *Precisa armazenar os dados*? A resposta pode ser sim ou não. No primeiro caso, o sistema armazena todas as mudanças nos prontuários dos pacientes. No segundo caso, o sistema captura a temperatura do paciente, talvez de algum aparelho, e envia um alerta se passar de um certo valor, mas sem armazenar a temperatura. Se responder não à pergunta, não é necessário utilizar a Blockchain. Se responder sim, passe à segunda pergunta.

A segunda pergunta é: *Há mais de um modificador*?. Entenda-se por modificador a entidade que realiza mudanças nas informações, seja inserindo-as, modificando-as ou removendo-as. A resposta pode ser sim ou não. No primeiro caso, diversas entidades (*e.g.*, cada sistema de informação das clínicas e hospitais) realizam modificações nos prontuários dos pacientes. No segundo caso, somente uma entidade é a responsável pela informação (vide centralização da informação, explicada na Seção 2.3). Se responder não à pergunta, não é necessário utilizar a Blockchain e talvez um banco de dados tradicional resolveria. Se responder sim, passe à terceira pergunta.

A terceira pergunta é: *Os TTP estão sempre online*? Entenda-se por TTP (*Trusted Third Party*, em inglês) às entidades às quais sempre é possível confiar, tanto na procedência quanto nas informações que entrega. Exemplos de um TTP seriam sistemas de informação do ministério da saúde ou do conselho federal de medicina. A resposta pode ser sim ou não. Se responder sim, talvez esse sistema (por ser confiável e permanecer sempre online) poderia ser responsável por armazenar as informações, não sendo necessário o uso da Blockchain. Se responder não, talvez esse sistema (por ser confiável, mas nem

sempre permanecer online) poderia ser usado como um certificador dos modificadores. Assim sendo, passe à quarta pergunta.

A quarta pergunta é: *Os modificadores são conhecidos*? A resposta pode ser sim ou não. Nesse caso, a pergunta refere-se a se os modificadores podem ser certificados e portanto responsabilizados em algum momento caso haja um comportamento malicioso. Por exemplo, conhecer quem é o responsável por um sistema de informação de um determinado hospital. Se responder não, cabe o uso de uma Blockchain do tipo não permissionada. Se responder sim, passe à quinta pergunta.

A quinta pergunta é: *Os modificadores são confiáveis*? A resposta pode ser sim ou não. No primeiro caso, quer dizer que os modificadores confiam que não existirão comportamentos maliciosos entre eles (*e.g.*, modificações indevidas, acessos sem permissão, etc). No segundo caso, não é possível garantir a confiança de todos (talvez há uma entidade terceira, *e.g.*, um plano de saúde, com problemas na justiça). Se responder sim à pergunta, não é necessário utilizar a Blockchain e talvez um banco de dados tradicional resolveria. Se responder não, passe à sexta e última pergunta.

A sexta pergunta é: *A auditabilidade é pública*? A resposta pode ser sim ou não. No primeiro caso, qualquer participante (podendo ser entidades ou até sistemas externos ou pessoas) tem permissão para validar todas as informações armazenadas. No segundo caso, somente um conjunto restrito e bem definido de participantes terá acesso às informações. Se responder sim, cabe o uso de uma Blockchain do tipo permissionada e pública. Se responder não, cabe o uso de uma Blockchain do tipo permissionada e privada.

2.9. Aplicações da Blockchain na área de Saúde

A Seção 2.3 apresentou que o uso da Blockchain permite a construção de sistemas descentralizados e auditáveis, com a característica de oferecer interoperabilidade e privacidade de dados. Esta seção explora como a Blockchain pode ser colocada a serviço dos sistemas de informação em Saúde. Para isso, primeiro, apresenta-se um levantamento sobre os usos da Blockchain em Saúde. A seguir, discorre-se sobre o potencial de algumas serem aplicadas no Brasil; para isso, são enumerados alguns cenários e casos de uso.

2.9.1. Levantamento sobre aplicações da Blockchain no contexto da Saúde

Um estudo bibliométrico recente [41] demonstrou que o número de artigos sobre Blockchain publicados em revistas da área de saúde ainda é pequeno se comparado com outras áreas. Dabbargh *et al.* [41] analisaram 995 artigos sobre Blockchain (de 2013 a 2018) indexados pelo portal Web of Science; destes, apenas 30 ($0,3\%$) foram publicados em revistas da área de saúde. O estudo também demonstrou um salto no número de artigos de 2016 para 2017, quando a atenção deixou o Bitcoin e passou a focar em usos gerais da Blockchain e de contratos inteligentes.

Apesar de não haver muitos artigos sobre Blockchain publicados em revistas da área de saúde, a literatura internacional (principalmente de Ciência da Computação e Engenharia) tem apresentado inúmeras proposta de utilização da Blockchain em saúde. A seguir apresenta-se uma lista de trabalhos relevantes publicados recentemente.

Casino *et al.* [42] realizaram uma revisão sistemática sobre os desafios e as diferentes áreas de aplicação da Blockchain. Fez uma análise qualitativa de 260 artigos e 54 relatórios (manuais, *white papers*, etc.) publicados entre 2014 e abril de 2018. O trabalho classificou as aplicações da Blockchain em 9 grandes áreas, a saber: financeira, negócios e indústria, gerenciamento de dados, verificação de integridade, governança, Internet das Coisas, privacidade e segurança, educação e saúde. Na área de saúde, considerou que o registro eletrônico de saúde provavelmente será a aplicação de maior impacto. Outras aplicações em saúde que poderiam se beneficiar da tecnologia Blockchain e, sobretudo, de contratos inteligentes seriam: transparência de recursos públicos aplicados em saúde, obtenção de dados para estudos longitudinais, arbitragem de processos (por exemplo, liberação automática de exames), acesso *online* de pacientes aos seus dados, compartilhamento de dados de saúde, controle de medicamentos e de ensaios clínicos e medicina de precisão.

Alonso *et al.* [43] fizeram uma revisão sistemática especificamente sobre os usos da Blockchain em Saúde, em artigos publicados entre 2010 e 2018 em algumas das principais bases de artigos (IEEE Xplore, Google Scholar, PubMed, Science Direct, Web of Science e ResearchGate). O trabalho encontrou 18 referências relevantes, sendo uma de 2016 [44], quatro de 2017 e treze de 2018, o que aponta para um crescimento da importância dessa temática. Segundo a revisão, os principais obstáculos são a escalabilidade e a implementação de controles de acesso. A principal vantagem seria o acesso a uma grande quantidade de informação anonimizada, que poderia ser utilizada para: desenvolvimento e aprimoramento de tratamentos personalizados, racionalização dos custos de ações de saúde e melhorias nas políticas de saúde. Alonso *et al.* [43] destacam ainda que a tecnologia Blockchain, associada ao avanço dos sistemas de informação e a maior participação e envolvimento dos pacientes, poderia levar a uma nova era do cuidado com a Saúde.

De forma geral, os trabalhos analisados por Alonso *et al.* [43] demonstram a viabilidade do uso da Blockchain para:

- Gerenciamento de identidade de pacientes.
- Registro de informações médicas com segurança e privacidade, permitindo a verificação de autenticidade de registros e preservando a identidade de pacientes e de profissionais de saúde [45].
- Rastreabilidade das ações médicas.
- Redução do tempo para interoperabilidade de dados [46]

Outra revisão sistemática sobre o uso da Blockchain em Saúde, realizada por Vazirani *et al.* [47], analisou qualitativamente 71 trabalhos, extraídos das seguintes bases: PubMed, Scopus, EMBASE, MEDLINE, ProQuest, CINAHL, AMED, Global Health, Books@Ovid e Cochrane Library. Este trabalho chamou a atenção para um detalhe importante não apontado em Alonso *et al.* [43], a compatibilidade legal. Citou o exemplo europeu, cujo Regulamento Geral sobre a Proteção de Dados[6] [48], no artigo 17, define

[6]Disponível em https://eur-lex.europa.eu/legal-content/PT/TXT/PDF/?uri=CELEX:02016R0679-20160504from=EN. Visitado em março de 2019.

o direito ao apagamento dos dados, também chamado "direito a ser esquecido". Desse modo, os dados devem poder ser apagados; entretanto, os dados gravados em uma estrutura Blockchain usual não podem ser apagados. No Brasil, a lei nº 13.709, de 14 de agosto de 2018, classifica os dados de saúde como dados sensíveis e estabelece o direito a ser esquecido nos artigos 7, 8, 15 e 16 desta lei. Desse modo, o projeto de armazenamento de dados de Saúde na Blockchain deve ter como requisito a funcionalidade de permitir apagar os dados de um paciente ou de revogar a sua autorização de acesso a estes dados. Voltaremos mais adiante a falar sobre este ponto.

Outro fator apontado por Vazirani *et al.* [47] foram os custos associados ao desenvolvimento de sistemas de informação, o que pode ser uma barreira para a adoção de sistemas baseados em Blockchain.

Park *et al.* [49] tentaram demonstrar experimentalmente os limites de uso da Blockchain para o compartilhamento de dados médicos. Para isso, criaram uma rede privada baseada na tecnologia Ethereum com um hospital e 300 pacientes. Ao final do experimento concluíram que era fundamental: minimizar a quantidade de dados efetivamente gravada na Blockchain, melhorar a privacidade dos dados e considerar os custos das transações. O trabalho foi baseado na rede Ethereum, onde cada transação tem um custo medido em *GAS*. Esse custo é proporcional ao tamanho do dado armazenado e inversamente proporcional a prioridade de gravação. Se fosse usada a rede oficial da Ethereum, o custo de uma transação, atualmente, seria de no mínimo cerca de 10 centavos de dólar. Parece pouco, mas quem absorve estes custos? Clientes ou unidades de saúde? Cabe ainda chamar a atenção para a imprevisível flutuação do valor da moeda digital Ethereum (ETH), que pode, eventualmente, tornar estes custos mais relevantes. O trabalho também chamou a atenção para o fato de que dados de saúde são dados sensíveis e que não podem ficar abertos nas redes Blockchain, devem ser protegidos por mecanismos criptográficos.

O trabalho de McGhin *et al.* [50] faz um resumo dos desafios tecnológicos para a implantação real de sistemas de informação em Saúde baseados em Blockchain. O texto traz dois pontos interessantes: *i*) apresenta uma compilação de projetos de software que dão suporte a aplicações de saúde usando Blockchain e *ii*) enumera vulnerabilidades agregadas aos sistemas de saúde pelo uso da Blockchain. Este último, em outras palavras, lembra que os sistemas de informação de saúde já estão sujeitos a requisitos severos de segurança e ao se usar Blockchain novas vulnerabilidades devem ser levadas em consideração, como, por exemplo, a susceptibilidade a ataques de 51% [51].

Ao todo, McGhin *et al.* [50] destacam nove iniciativas (MedRec [52], Gem Health Network [53], OmniPHR [54], PSN [55], Virtual Resources [56], Context-driven Data Logging [57], MedShare [58], Trial and Precision Medicine [59] e Healthcare Data Gateways [60]) e chama a atenção para o fato de que todos os trabalhos possuem limitações de escalabilidade e/ou de segurança. Por exemplo, o projeto MedRec é, atualmente, o artigo mais citado aplicações da Blockchain em Saúde [52]. O trabalho trata do uso da Blockchain para o armazenamento auditável do histórico de interações médicas do paciente. A arquitetura, baseada em *Proof of Work*, permite a mediação de permissão de acesso aos dados, mas, segundo McGhin *et al.* [50], o protótipo MedRec não garante satisfatoriamente o anonimato dos dados individuais e torna possível, por meio de técnicas forenses, descobrir a identidade dos pacientes e de prestadores de serviços.

A partir das leituras acima, pode-se concluir que ainda não existem aplicações robustas para o uso da Blockchain em saúde, ou que estas aplicações ainda estão em desenvolvimento. Mas podemos esperar que em breve possamos encontrar aplicações em produção. Segundo Perera *et al.* [37][7], o surgimento de aplicações Blockchain mais complexas pode levar de 5 a 10 anos.

O caminho para o desenvolvimento de aplicações relevantes, seguras e escaláveis, passa pelo refinamento de seus requisitos. A Seção 2.9.2, a seguir, organiza alguns dos principais cenários de aplicações e seus requisitos funcionais e não funcionais.

2.9.2. Cenários e casos de uso da Blockchain em Saúde

O trabalho de Liam Bell [61] argumenta que são quatro as principais potenciais contribuições da Blockchain para a área da saúde, a saber:

- **Compartilhamento de dados** de saúde, principalmente entre médicos e entre instituições de saúde, com a anuência do paciente. Sem dúvida, a principal aplicação nessa categoria são os prontuários eletrônicos.
- **Interoperabilidade de dados** de saúde em contexto nacional, permitindo avanços em desenvolvimento de sistemas para controle epidemiológico e de saúde coletiva. Nessas aplicações são fundamentais as aplicações de técnicas de anonimização dos dados.
- **Rastreamento da cadeia de suprimentos e de dispositivos médicos**, permitindo a auditoria sobre o uso de equipamentos em ambiente hospitalar, a prevenção de subutilização e a análise de fraudes.
- **Rastreamento da cadeia de distribuição de medicamentos**, o que permite verificar a prescrição de medicamentos aos pacientes, identificar lotes de medicamentos com problemas e realizar, se necessário, *recalls* de lotes de medicamentos (vencidos, falsificados ou com problemas no processo de fabricação).

Zhang *et al.* [15], por sua vez, acrescentam mais itens a essa lista. O trabalho aponta seis casos de uso principais, sendo que os três últimos são substancialmente diferentes dos apontados por Liam Bell [61]:

- **Compartilhamento seguro de dados** para, por exemplo, viabilizar telemedicina.
- **Monitoração da prescrição de drogas**. Nesse item, inclui-se a monitoração de drogas controladas e de alto custo.
- **Observação agregada de eventos** (Big Data) com aplicações principalmente em saúde coletiva.
- **Identificação de pacientes** (PKI [62]).
- **Compartilhamento de dados para pesquisa científica**.

[7] Artigo sob avaliação, disponível em https://peerj.com/preprints/27529/.

- **Implementação de estruturas autônomas**, por exemplo, para a gestão e controle de seguros de saúde suplementar.

A identificação de pacientes pode ser considerada uma das partes fundamentais de sistemas de Prontuário Eletrônico do Paciente — e uma das componentes mais complexas. Por exemplo, atualmente no Brasil ainda não há um sistema de informação para identificação única dos pacientes. Um dos primeiros esforços foi a implementação do Cartão SUS e mais recentemente a proposta de um Documento Nacional de Identificação. Por sua importância, a componente de gestão de identidade de saúde deve ser tratada como um caso de uso isolado, separado dos Prontuários, com requisitos próprios de segurança, privacidade e de integração com outros sistemas.

O compartilhamento de dados para pesquisa científica visa a análise de dados de saúde, longitudinais, no curso de longos períodos de tempo, para desenvolvimento de drogas e de tratamentos mais eficientes. Requer, segundo os padrões modernos de proteção de dados individuais, mecanismos para concessão e revogação de direitos de uso dos dados.

A última sugestão de Zhang *et al.* [15] passa pela organização de estruturas autônomas para prestação de serviços médicos. Nesse tipo de aplicação a Blockchain é responsável por, automaticamente, analisar demandas e ofertas de serviços de saúde. Por exemplo, um prestador de serviços pode ser diretamente remunerado utilizando mecanismos de pagamento semelhantes ao Bitcoin. O principal desafio em aplicações como essa é a escrita de contratos inteligentes capazes de tratar toda a complexidade de sistemas de saúde complementar.

A seguir, tratamos os principais requisitos para aplicações de Identificação de Pacientes, Prontuário Eletrônico e Gestão de Medicamentos. Escolhemos estes casos de uso por eles serem de aplicação imediata [37] e relevantes para a área de saúde.

2.9.2.1. Identificação Única de Pacientes

O leitor pode estranhar que o primeiro caso de uso da Blockchain em Saúde a ser apresentado seja a identificação única de pacientes e não os prontuários eletrônicos. Ao longo do texto indicamos que os prontuários devem ser a aplicação de maior impacto, mas para isso é necessário um serviço de identidade e de gerenciamento de credenciais digitais.

O processo de atendimento médico é rico de momentos em que a privacidade de um paciente fica exposta. Dados pessoais ficam expostos quando preenchemos formulários de papel, quando falamos nossos dados pessoais a um atendente e até mesmo quando são transmitidos ou armazenados sem seguir os devidos requisitos de segurança [63].

Essas fragilidades de segurança, quer sejam de processo, quer sejam de sistema, podem ser minimizadas com o uso de sistemas para gerenciamento de identidades. Orman *et al.* [62] fazem reflexões sobre a migração da identidade dos indivíduos do papel para os meios digitais. Aponta que Blockchain é, talvez, a mais forte das candidatas para a criação de identidades digitais.

Um serviço digital de gerenciamento de identidades e de credenciais deve:

- Permitir a identificação única de um usuário por um **ID principal**, que pode ser um número, uma validação biométrica ou uma chave pública, entre outras possibilidades. Cabe observar que este ID principal pode ser mantido privado e que pode não ser possível identificar um indivíduo a partir dele.

- Permitir a criação de ilimitados **IDs secundários**. Se um paciente utiliza o mesmo ID em diferentes instituições, ele expõe a sua identidade; de modo alternativo, cada relação do paciente poderia utilizar um ID diferente e o serviço de gerenciamento de identidades seria responsável em gerenciar esses IDs.

- Armazenar dados pessoais, chaves de acesso a estes dados e validar solicitações de acesso aos dados. Deveria, inclusive, ser capaz de revogar acessos.

- Emitir credenciais de acesso que permitam a um requisitante ter acesso a dados específicos, sem expor totalmente os dados de saúde do paciente. Essas credenciais podem ter atributos adicionais, tais como localização, nível de acesso aos dados e validade da autorização de acesso.

O sistema deve proteger a identidade e gerenciar as credenciais de acesso aos dados pessoais. Diferentemente de como é hoje, um paciente, ao chegar da triagem de um hospital, poderia, ao invés de repassar todos os seus dados pessoais, mostrar um QRCode contendo uma credencial de acesso aos seus dados, protegendo assim sua identidade e privacidade.

No contexto de sistemas públicos de saúde, o sistema pode ainda ser utilizado para garantir a consistência de cadastros de usuários. Atualmente, na prática, não existe um registro único de pacientes. Talvez, por uma questão de privacidade, esse registro único nem devesse existir. Talvez trouxesse mais problemas do que benefícios. Sem um sistema de gestão de identidade, é difícil garantir que não haja vazamentos e maus usos de dados pessoais.

Na busca por uma padronização de cadastros, o Ministério da Saúde propôs a ampla adoção do Cartão Nacional de Saúde (Cartão SUS), regulamentado pela portaria número 940/2011, para construção de cadastros de usuários. Outra iniciativa brasileira para padronização da identificação, esta mais recente, foi a proposta do Documento Nacional de Identificação (DNI), implementado pela lei Nº 13.444/2017 e regulamentado no decreto Nº 9.278/2018.

Entretanto, os esforços em torno do Cartão SUS e do DNI não garantem a criação de uma base consistente de usuários. Mas a existência de uma base consistente e com identificação única (mesmo que não revele a identidade do paciente) é essencial para construção de sistemas de saúde que permitam a interoperabilidade de dados.

A identificação de usuários poderia ser realizada a partir da construção de uma infraestrutura Blockchain, onde o usuário é representado por seu ID principal. Os dados pessoais sensíveis podem ser gravados na Blockchain, ou em um *data lake*, de forma protegida por chaves criptográficas [14]. Contratos inteligentes podem ser implementados para a gestão de credenciais digitais, assim como para verificar requisições de dados.

A implementação de tal rede Blockchain certamente não é trivial. Uma primeira questão é a sua manutenção. Quem deve manter a rede no ar? Nessa aplicação não pode-se contar com os benefícios financeiros para manutenção da rede, como acontece nas redes Bitcoin e Ethereum. Acreditamos que tal rede poderia ser mantida por órgãos públicos (Ministério da Saúde e secretarias estaduais e municipais). A arquitetura do serviço provavelmente seria a de uma Blockchain permissionada.

Dentre os requisitos não funcionais mais importantes estaria o tempo de resposta às requisições de autorização de acesso a dados, que idealmente não deveria ultrapassar a ordem de segundos. A tecnologia atual da Blockchain permissionada permite a realização de transações com tempo de execução na ordem de segundos [10]. Mas como poderia ser a implantação de uma Blockchain permissionada com tamanho e abrangência compatíveis com o SUS?

2.9.2.2. Registro Eletrônico de Dados de Saúde

A maioria dos autores que descrevem os casos de uso da Blockchain [37, 50, 41, 49, 47, 15] colocam as aplicações de Prontuário Eletrônico e de compartilhamento de dados de saúde dentre as de maior impacto na área de saúde. Segundo Zhang*et al.* [15], esse tipo de tecnologia pode ser vista sob dois pontos de vista: o ponto de vista das unidades de saúde, que normalmente se utilizam de abordagens centralizadas para armazenar dados de saúde dos pacientes, e ponto de vista do paciente, que deseja ter uma visão geral dos seus dados de saúde. O primeiro está normalmente associado ao conceito de *Electronic Health Records* (EHR) e o segundo a *Personal Health Records* (PHR).

Imagine, para simplificar a apresentação da ideia, que todos os dados de saúde de um paciente sejam gravados em uma Blockchain. Suponha ainda que essa Blockchain conta com um serviço de identificação única de pacientes, como descrito na seção anterior. O uso da Blockchain nessas categorias de aplicação poderia [14]:

- Simplificar a **interoperabilidade** de dados. Os dados do paciente, coletados em um hospital A, poderiam ser acessados durante uma consulta em um outro hospital B, desde que o acesso fosse autorizado.
- Dar real **controle** ao paciente sobre os dados que dizem respeito a ele. Segundo a legislação atual, um paciente pode solicitar que os seus dados médicos sejam apagados de uma instituição. Na prática isso é muito difícil de ser realizado, o uso de contratos inteligentes poderia permitir a revogação de direitos de acesso a documentos [14].
- Tornar **transparente** os usos e os acessos aos dados dos pacientes. Pacientes poderiam monitorar o uso dos seus dados para a agregação de dados e geração de estatísticas de saúde coletiva. Caso estes dados fossem utilizados em pesquisas e desenvolvimentos de produtos, seria possível reclamar direitos sobre a propriedade intelectual desenvolvida [64]. É interessante observar que o uso correto da Blockchain e de técnicas de anonimização de dados pode permitir a disponibilização de dados sem o comprometimento da identidade dos pacientes.

A implementação de tais redes de compartilhamento de dados depende da migração de sistemas de EHR atuais para um novo padrão. Um padrão que contemple restrições severas de segurança e que especifique interfaces claras para compatibilidade de trocas de dados. Além de implementar restrições de segurança da área de saúde (*e.g.* HIPAA e NGS2), sistemas baseados em Blockchain estão sujeitos a novos ataques, como os ataques de 51% e *Sybil*.

Tal rede provavelmente seria permissionada, mantida por unidades de saúde e financiadores do sistema de saúde. A principal restrição técnica para a implantação seria a escalabilidade em termos de volume de transações. Hoje, por exemplo, o Brasil conta com cerca de 7.522 hospitais [65] — muitos deles mal conectados à Internet.

O leitor pode se perguntar sobre onde deveriam ficar armazenados os dados do paciente. Certamente não devem ficar na Blockchain, mas em *data lakes* dedicados. Em outro trabalho [14], sugerimos o uso de serviços atuais de armazenamento em nuvem, tais como Google Drive e Dropbox. Mas cabe citar que o próprio Governo Federal brasileiro possui uma iniciativa para criação de um *data lake*, que deverá ser utilizado por unidades de saúde tanto públicas quanto privadas: o Conjunto Mínimo de Dados (CMD, vide: https://conjuntominimo.saude.gov.br/#/cmd). Um sistema de EHR baseado em Blockchain poderia gravar os dados nesses *data lakes* genéricos e gravar na Blockchain apenas o link para esses dados.

É interessante pensar que as unidades de saúde podem beneficiar-se da rede, pois a qualidade dos atendimentos tende a aumentar. Para gestores de saúde complementar, por exemplo, a principal vantagem seria econômica, por exemplo, com a não realização de exames que tenham sido recentemente feitos. Entretanto, deve-se ter em mente que os maiores beneficiados seriam os pacientes, quer por ter mais dados, quer por ter maior controle sobre estes dados.

2.9.2.3. Gestão de Medicamentos e de Prescrições (*e-Prescription*)

Uma aplicação mais específica é a gestão de medicamentos e de prescrições médicas (receitas). A aplicação de controle eletrônico de prescrições também pode ser considerada como uma parte dos sistemas de EHR.

No Brasil, quase a totalidade das receitas são em papel. O maior problema com as receitas de papel são os erros de interpretação; mas podemos citar outros ploblemas, como a perda do registro em papel e a facilidade para falsificação.

No mundo, diversos países já têm adotado sistemas de prescrição eletrônica (*e-Prescription* ou *e-Rx*). Na Europa, os sistemas avançaram, sobretudo, devido a necessidade de criar um padrão europeu comum [66]. Em alguns países, como na Estônia, a prescrição digital já contabiliza cerca de 80% do total de prescrições. As vantagens do uso de sistemas de *e-Prescription*, além de eliminar papel e aumentar a qualidade da informação, são:

- Controlar o abuso de uso de medicamentos.
- Permitir a análise de interações medicamentosas.

- Simplificar comércio eletrônico de medicamentos, inclusive de medicamentos controlados.

O uso da Blockchain permitiria aos sistemas de *e-Prescription* melhorar a interoperabilidade entre sistemas, ampliar a capacidade de fazer o *recall* de lotes de medicamentos (se a cadeia de produção for registrada na Blockchain) e aumentar a confiabilidade da informação.

No Brasil, o SUS é responsável por garantir o acesso a medicamentos de alto custo; sabe-se que o sistema de distribuíção é sujeito a falhas, fraudes [8] e até mesmo a furtos [9]. Especialmente no contexto de distribuição de medicamentos de alto custo, um sistema baseado em Blockchain permitiria o rastreamento desses medicamentos, o controle de validade e previsão de consumo.

Sistemas de *e-Prescription* baseados em Blockchain teriam como principais requisitos a existência de identificação única e o volume de transações. Pode-se imaginar que cadeias de distribuição de medicamentos possam ter interesse em manter tais sistemas, quer para melhorar a qualidade do serviço prestado, quer para conhecer hábitos e demandas dos clientes. Uma arquitetura permissionada seria a escolha mais provável para esse tipo de aplicação.

2.10. Ferramentas

Os projetos Hyperledger e Ethereum são principais referências sobre desenvolvimento de contratos inteligentes para o contexto de, respectivamente, Blockchains privadas e públicas. A seguir são apresentadas as características gerais de cada uma dessas soluções.

2.10.1. Hyperledger Fabric

O projeto *Hyperledger*, e seu software *Hyperledger Fabric*, é de código aberto. Nascido em 2015, seu objetivo é prover uma plataforma flexível, que permita usar a Blockchain em diferentes cenários além das criptomoedas. O projeto é mantido pela Linux Foundation e é apoiado por centenas de empresas de tecnologia, como a IBM, Intel, entre outras. As quatro principais característica do *Hyperledger Fabric* são: (1) as redes não são anônimas (*permissioned networks*), estabelecendo assim uma maior confiança na rede; (2) as transações têm critérios de permissão para leitura; (3) possui uma arquitetura flexível; (4) facilidade de uso, permitindo escrever aplicações em diversas linguagens de programação. Além disso, por ser permissionada, o Hyperledger Fabric não exige consumo de energia excessivo, como o Bitcoin.

No Hyperledger Fabric, os componentes são divididos por responsabilidades, sendo eles: os nós autorizadores, armazenadores, coletores, coordenadores e clientes. Além disso, os componentes na arquitetura se comunicam utilizando canais, estruturas criadas especificamente para realizar transações de forma privada e confidencial, isolando dife-

[8] Leia *"SUS joga fora R$ 16 milhões em medicamentos de alto custo"*, por André Shalders. Disponível em https://www.bbc.com/portuguese/brasil-41007650.

[9] Leia *"Ladrões roubam medicamentos de alto custo do Hospital São Paulo"*, por Roberto Paiva e Paula Paiva Paulo. Disponível em https://g1.globo.com/sp/sao-paulo/noticia/2018/08/02/ladroes-roubam-medicamentos-de-alto-custo-do-hospital-sao-paulo-veja-video.ghtml

rentes aplicações. Assim, o *canal* é o meio pelo qual os componentes podem se comunicar de forma segura e confiável com a Blockchain.

Os nós autorizadores (*fabric certificate authorities*) são os responsáveis por duas tarefas: a primeira, em certificar que qualquer componente, seja este um usuário ou um contrato inteligente, que quiser utilizar o sistema seja quem diz ser (em outras palavras, pelo reconhecimento da autenticidade do componente); a segunda, em autenticar o componente e autorizá-lo a usar certas funcionalidades (*e.g.*, realizar transações) ou acessar outros componentes, após sua certificação.

Os nós armazenadores (*committing peer*) são os responsáveis por persistir uma ou mais cadeias de transações, que foram transmitidas através dos canais criados no sistema. Assim, são os nós que armazenam as diversas Blockchains, um por cada canal criado. Essa abordagem, de uma Blockchain por canal, permitirá obter dois benefícios: privacidade e escalabilidade.

Em se tratando da privacidade, um componente não poderá acessar (*i.e.*, visualizar ou modificar) uma Blockchain de um nó armazenador associado a um canal, se não tiver acesso a esse canal. Nesse contexto, a privacidade e confidencialidade das informações fica assegurada para os componentes que obtiveram a permissão.

Já na escalabilidade, note que poderão haver diversas Blockchains, uma por canal, o que permitirá distribuir a quantidade de transações entre diversos nós armazenadores, aumentando a quantidade de requisições que um nó deve atender, consequentemente, aumentando a escalabilidade do sistema.

Os nós coletores (*endorsing peer*) são os responsáveis por duas tarefas: a primeira, coletar as transações advindas dos clientes; a segunda, analisar, utilizando contratos inteligentes, se a transação tem alguma política ou regra associada.

Os nós coordenadores (*ordering peer*) são os responsáveis por duas tarefas: a primeira, receber as transações dos clientes; a segunda, realizar uma ordenação nessas transações para que a Blockchain esteja consistente (*i.e.*, fique igual em todos os nós que armazenarão essas transações). Nesse sentido, todos os nós coordenadores atuando sobre uma determinada Blockchain deverão chegar a um consenso da ordem na qual as transações serão adicionadas na Blockchain pelos nós armazenadores. O recebimento das transações é realizada usando a tecnologia Apache Kafka [67], que permite armazená-las de forma distribuída e com tolerância a folhas. Já o consenso é realizado utilizando a tecnologia Zookeeper [68], que aplica uma versão do Paxos, explicada na Seção 2.5.8.

Os nós clientes (*application*) são encarregados por efetuar transações no sistema, por enviá-las aos nós coletores e repassá-las aos nós ordenadores. No Hyperledger Fabric, um cliente do sistema pode ser uma pessoa física (que utiliza o sistema através de um aplicativo), ou uma pessoa jurídica que utiliza o sistema através de interfaces de comunicação ou aplicativos.

A Figura 2.15 mostra um exemplo de uma arquitetura que pode ser criada para um cenário onde existem três organizações que utilizam diferentes canais de comunicação do Hyperledger. Na figura, a cor representa a que organização o componente está associado.

O primeiro passo para criar a arquitetura é gerar o nó autenticador, responsável

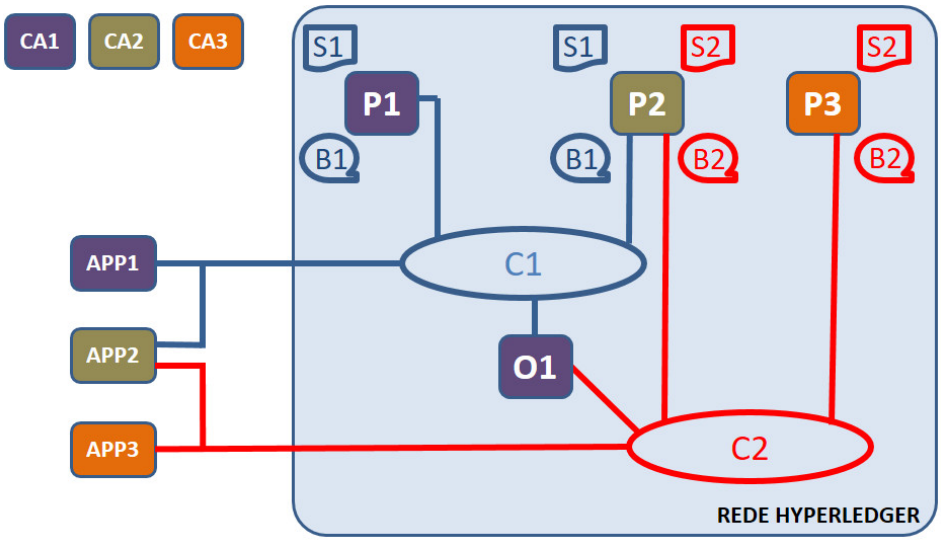


Figura 2.15. Exemplo da Arquitetura Hyperledger

pela autenticação dos componentes do sistema. Assim, tanto a organização 1 quanto a 2 e a 3 geram seus próprios nós autenticadores (CA1, CA2 e CA3).

Após os autenticadores, será necessário criar os nós armazenadores, coletores e ordenadores. Nesse sentido, pode-se observar que a organização 1 designou o nó (P1) para realizar tanto as funções de coletor quanto de armazenador. Note que, por ser um coletor, também terá a responsabilidade de analisar as transações utilizando contratos inteligentes (S1). Além disso, por ser um armazenador, terá a responsabilidade de armazenar a Blockchain (B1) do canal 1 (C1). Da mesma forma, o nó P3, designado pela organização 3, será responsável por analisar as transações utilizando o contrato inteligente S3 e armazenar a Blockchain (B2) do canal 2 (C2). Por outro lado, note que o nó P2 foi designado pela organização 2, porém é responsável por analisar e armazenar as transações em ambas Blockchains B1 e B2.

A seguir, é necessário definir a comunicação entre as organizações, em outras palavras, definir os canais. Note que na arquitetura de exemplo existem dois canais C1 e C2. No canal C1, somente os componentes autorizados pela organização 1 e 2 poderão se comunicar e utilizar a Blockchain B1. Nenhum outro componente (por exemplo, da organização 3) poderá acessar esse canal. Da mesma forma, o canal C2 permite somente a comunicação e utilização da Blockchain B2 pelas organizações 2 e 3.

Finalmente, é necessário definir os nós responsáveis pela ordenação das transações e as aplicações que utilizarão a arquitetura. Na figura, pode-se observar que há somente um nó ordenador O1, designado pela organização 1 e três clientes (cada um autorizado pelo respectivo componente autorizador da organização correspondente). Note que o cliente APP1 ou APP3 somente possuem acesso ao canal 1 ou 2, respectivamente. Já o cliente APP2 possui acesso tanto ao canal 1 quanto ao canal 2 (nesse sentido, o cliente pertence às duas organizações, mas com dois acessos diferentes).

A Figura 2.16 mostra como o fluxo das mensagens transita na arquitetura, criada

no exemplo anterior, quando um cliente quer enviar uma transação. Vamos assumir que a transação requer a modificação de um prontuário eletrônico do cliente e que somente a aplicação APP1 realiza a transação. Para uma melhor visualização, serão eliminadas as conexões e os componentes que fazem parte do canal 2, haja vista que a APP1 somente utilizará o canal 1.

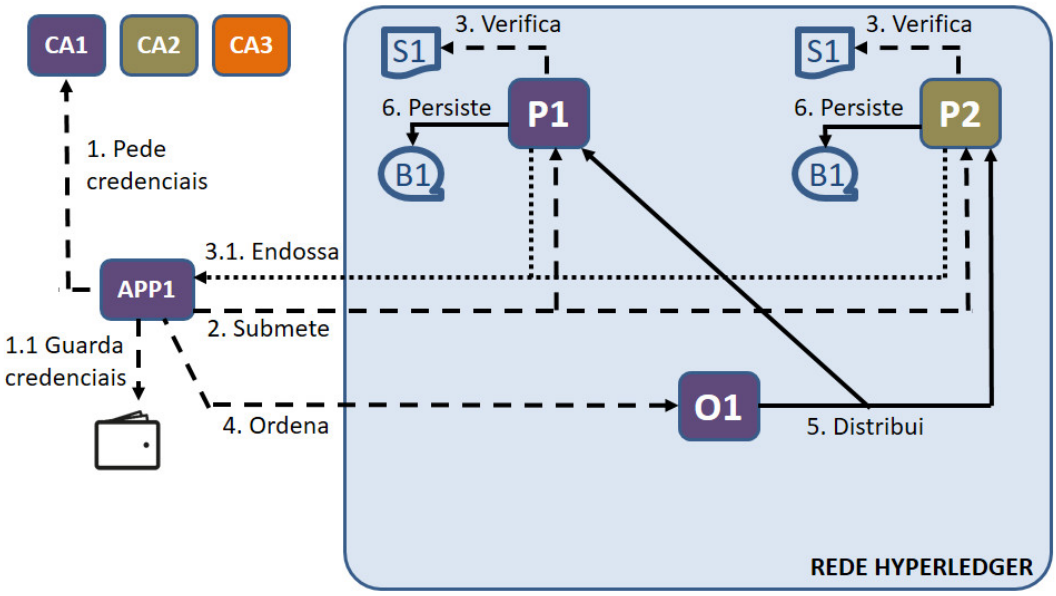


Figura 2.16. Fluxo de Mensagens

1. O cliente APP1 obtém o acesso do autorizador CA1 correspondente a sua organização. Quando o autorizador entrega as credenciais de acesso, o cliente as mantêm em uma estrutura denominada carteira (Wallet) e as armazena no seu disco rígido, pendrive, etc. Note que esse armazenamento permitirá ao cliente APP1 conectar-se novamente sem precisar pedir a autorização ao CA1.

2. O cliente se conecta ao canal C1 e submete a transação para um dos nós coletores desse canal, no caso P1, P2 ou ambos.

3. O peer coletor P1 verifica, utilizando o contrato inteligente S1, se a transação pode ser realizada, devolvendo essa informação ao cliente (em outras palavras, endossando a transação). Note que, dependendo do contrato inteligente, pode ser que seja necessário que o nó coletor P2 também tenha que verificar a transação.

4. O cliente recebe a informação de que a transação pode ser realizada e a envia para o nó ordenador.

5. O nó ordenador recebe a transação, cria um bloco com esta transação, e a distribui para os nós armazenadores (no caso P1 e P2).

6. Os nós armazenadores analisam novamente a transação (validam se houve alguma outra transação anterior que também modificou o prontuário). Se não houver nenhum problema, o bloco (com a transação) é persistido na Blockchain B1, enviando a informação ao cliente. O envio da informação não é mostrado na figura.

Como mencionado, (contratos inteligentes) são programas que permitem a validação de uma transação em termos de regras de negócios. Por exemplo, se o negócio está relacionado com operações financeiras entre pessoas, uma regra de negócio a ser implantada seria a verificação de se a pessoa tem saldo para realizar a transferência.

No Hyperledger, um contrato inteligente pode ser implementado em diversas linguagens de programação, atualmente com versões para Node.js (Javascript), Go e Java. Entretanto, independente da linguagem, o contrato inteligente possui uma estrutura a ser seguida pelo desenvolvedor.

Na estrutura, o desenvolvedor deve definir como mínimo: o modelo (com os ativos e participantes); as transações a serem realizadas; o controle de acesso. Em posse das definições, o Hyperledger criará um arquivo .bna, o qual será implantado nos coletores.

Para entender a estrutura, daremos um exemplo simples de um contrato inteligente que valida se um usuário pode ou não visualizar o prontuário eletrônico armazenado na Blockchain.

No modelo, os ativos (denominados *asset*) serão os prontuários eletrônicos. cada um com um identificador único, o site onde o pdf está armazenado e o cpf do paciente ao qual está associado, definido como:

```
asset Prontuario identified by idProntuario {
   o String idProntuario
   o String urlProntuario
   o String cpfPaciente
}
```

Os participantes (denominados participant) serão todas os pacientes que inseriram seu prontuário eletrônico, definido como:

```
participant Paciente identified by cpfPaciente {
   o String cpfPaciente
}
```

A única transação (denominadas transaction) permitida será a de visualizar o prontuário, desde que o cpf do participante seja o mesmo do cpf inserido no ativo. Assim, a transação será definida como:

```
transaction Visualizar {
   --> Prontuario prontuario
   --> Paciente pac
}
```

Cuja implementação, mostrada abaixo, recebe a transação 'Visualizar' (linha 1), a seguir verifica se os cpf são os mesmos (linha 2), e mostra a url do pdf (linha 3) ou uma mensagem de "sem acesso" (linha 5), dependendo da verificação dos cpf.

```
1 function verProntuario(Visualizar) {
2    if (Visualizar.prontuario.cpfPaciente == Visualizar.pac.cpfPaciente) {
3       return Visualizar.prontuario.urlProntuario
4    } else {
5       return "sem acesso"
6    }
```

```
7 }
```

Finalmente, o controle de acesso (denominado rule) permite a definição das permissões que o sistema dará tanto para a execução das transações, quanto para os participantes. Por exemplo, a regra abaixo permite que todos os participantes possam acessar a função verProntuario.

```
rule AcessoTotal {
  description: "Permissoes para todos os participantes"
  participant: "org.hyperledger.composer.system.Participant"
  operation: ALL
  resource: "org.hyperledger.composer.system.**"
  action: ALLOW
}
```

2.10.2. Ethereum

O projeto Ethereum também é de código aberto. Nascido em 2015, seu objetivo é prover uma plataforma descentralizada, que permita a execução de contrato inteligente para usar a Blockchain em diferentes cenários além das criptomoedas. O projeto é mantido pela Ethereum Foundation e é apoiado pela Enterprise Ethereum Alliance e por centenas de programadores do mundo todo. A principal característica, que a diferencia com a Blockchain do Bitcoin, é a possibilidade de executar os contrato inteligente, permitindo criar regras específicas para cada tipo de negócio.

No Ethereum, os componentes são divididos por responsabilidades, sendo eles: os nós armazenadores, mineradores, coletores e clientes. A diferença do Hyperledger, que permite ter diversas Blockchains no mesmo sistema, no Ethereum existem duas Blockchains: o *testnet*, Blockchain de teste utilizado pelos desenvolvedores na criação de suas aplicações e o *mainnet*, Blockchain oficial com as informações reais de todas as transações da rede.

Os nós armazenadores, denominados no Ethereum de Full nodes, são os encarregados de armazenar a cadeia de blocos e transações validados desde o bloco gênesis. Nesse sentido, esses nós permitem ter a prova da integridade da Blockchain.

Os nós mineradores são nós armazenadores, ou seja, possuem a Blockchain completa, com a capacidade de criar novos blocos, utilizando o consenso via prova de trabalho POW explicada na Seção 2.5.8.

Os nós coletores, denominados no Ethereum de Light-weight nodes, geralmente são nós cujo poder computacional ou de armazenamento é pequeno (por exemplo, dispositivos móveis, plugins em navegadores Web, entre outros). Esses nós são encarregados de verificar certas transações, mas confiando nas informações dos armazenadores, haja vista que não possuem a Blockchain completa. Um exemplo de uso é verificar o saldo de um determinado identificador, olhando somente as transações onde ele aparece.

Os nós clientes, ao igual que no Hyperledger, são encarregados por efetuar transações no sistema e por enviá-las aos nós coletores ou armazenadores (que, no caso, podem ser eles mesmos, dependendo do poder computacional). No Ethereum, os clientes podem ser uma pessoa física ou também um contrato inteligente, já que estes também geram

transações de acordo a certas regras de negócios. Cada cliente possui uma chave pública e outra privada, a chave publica é seu identificador na rede Ethereum e a chave privada é utilizada para efetuar uma transação na rede.

Os contratos inteligentes do Ethereum podem ser considerados programas que funcionam sobre a Blockchain, esses programas são executados em uma máquina virtual chamada *Ethereum Virtual Machine* (EVM). A EVM é responsável por se comunicar com as informações armazenadas na cadeia de blocos e conforme as regras do contrato, pode transferir valores, inserir novas informações ou requisitar chamadas para APIs externas a rede.

Os contratos inteligentes podem ser desenvolvidos em linguagens de programação de alto nível como o java ou python. Há também a linguagem Solidity, linguagem de alto nível semelhante ao JavaScript, esta é a mais utilizada pelos desenvolvedores. Existe algumas IDEs que auxiliam no processo de desenvolvimento, uma destas é o Remix, plataforma que se conecta a rede *testnet* e possibilita criar e testar os contratos inteligentes.

Diferente do Hyperledger, no Ethereum para cada inserção de informação na Blockchain é cobrada uma taxa chamada gás, esta taxa deve ser enviada junto a transação e para seu pagamento é utilizado o éter, moeda digital baseada na rede Ethereum. Este taxa foi criada com a intenção de proteger a rede de ataques e abusos por parte dos participantes.

Para entender melhor um contrato inteligente criado na rede Ethereum, mostraremos um exemplo simples da estrutura de dados utilizada para armazenar as informações do pacientes e uma função criada para validar se um usuário possui acesso para exibir estas informações, neste exemplo foi utilizado a linguagem *Solidity*. A estrutura denominada Paciente possui alguns atributos: *paciente* do tipo *address* é o identificador único do paciente. O tipo *address* é um endereço público da rede Ethereum. O campo *url* armazena o site onde esta armazenado o pdf. O campo *acessos* é um *array* de endereços, será o responsável por armazenar os usuarios que possuem acesso as informações.

```
struct Paciente {
    address paciente;
    string urlProntuario;
    address[] acessos;
    mapping(address => bool) roles;
}
```

Utilizando a estrutura anterior, foi criado uma função para validar se o usuário que está solicitando as informações possui tal acesso. A função *VerificaAcesso* recebe por parâmetro o identificador do paciente *_paciente*. A primeira verificação realizada é se o solicitante é o próprio paciente, caso seja, este poderá visualizar suas informações (Linha 2). A próxima validação é se o identificador do solicitante esta listado na relação de identificadores com permissão de acesso do paciente (linha 3). Caso as duas verificações anteriores sejam falsas, a função não permitirá que as informações do paciente sejam exibidas.

```
1 modifier VerificaAcesso (address _paciente) {
2     if ( msg.sender != _paciente &&
3         !pacientesLista[_paciente].roles[msg.sender]) {
```

```
4             revert();
5         }
6         _;
7     }
```

2.11. Leituras sugeridas

Caberá ao leitor continuar a pesquisa sobre Blockchain e suas ferramentas. Esse mercado está em grande atividade e diariamente surgem novas técnicas, ferramentas e algoritmos. Para maior contato com a área, sugerimos uma série de recursos e leituras:

- Não apenas por uma questão histórica, mas também por sua clareza e para tomar contato com a concepção inicial, sugerimos a leitura do artigo seminal de Satoshi Nakamoto [3] sobre Bitcoin.
- O artigo de Zhang *et al.* [15] oferece um bom panorama sobre os usos de Blockchain em saúde.
- As plataformas Coursera (www.coursera.org), edX (www.edx.org) e Udacity (www.udacity.com) oferecem vários cursos sobre Blockchain, sendo alguns gratuitos.
- A documentação das plataforma Ethereum (http://www.ethdocs.org) e os tutoriais para o Hyperledger Fabric (https://hyperledger-fabric.readthedocs.io/en/latest/tutorials.html) são excelentes fontes de informação. Essa documentação está atualizada e nela pode-se encontrar o *roadmap* para os futuros desenvolvimentos.

  Àqueles que pretendem ensinar Blockchain, sugerimos apresentar a plataforma Ethereum para a escrita dos primeiros contratos inteligentes. A plataforma Hyperledger Fabric seria outra escolha natural para uma experiência didática, mas a instalação da plataforma pode ser uma barreira para os estudantes menos avançados.

Além dos sites e referências acima, pode-se acompanhar o mercado de criptomoedas e de lançamentos de novas ferramentas na mídia especializada (exemplos: https://cointelegraph.com/ e https://www.coindesk.com).

2.12. Conclusões

Este capítulo apresentou as características principais de Blockchain, suas limitações e seus casos de uso principais em Saúde. Esperamos ter oferecido uma visão do potencial e das limitações de Blockchain, tal que os profissionais de Sistemas de Informação possam discernir quando usar Blockchain e quais devem ser as escolhas de projeto. O que deve conter uma transação? Qual o mecanismo de consenso mais apropriado? Entre outras decisões de projeto.

Antes de encerrar, chamamos a atenção para o conceito de confiança. A principal inovação de Blockchain foi deslocar a garantia de confiança das instituições para a arquitetura computacional. Mesmo que essas arquiteturas ainda sejam muito complexas, os próximos anos devem ser prolíficos na redução de complexidade das soluções,

assim como na mitigação dos desafios de escalabilidade, privacidade, mutabilidade, entre outros.

Em poucos anos, as aplicações baseadas em Blockchain poderão ser uma das principais formas para registro confiável de dados. Para que isso aconteça serão necessários desenvolvedores de software capacitados para compreender e dominar a complexidade das implantações de Blockchain e dos contratos inteligentes. Esperamos que esse capítulo possa ajudar nessa caminhada.

Referências

[1] Melanie Swan. *Blockchain: Blueprint for a new economy*. O'Reilly Media, Inc., 2015.

[2] Fabíola Greve, Leobino Sampaio, Jauberth Abijaude, Antonio Coutinho, Ítalo Valcy, and Sílvio Queiroz. Blockchain e a revolução do consenso sob demanda. *Minicursos do Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (Minicursos_SBRC)*, 36, 2018.

[3] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf, 2008.

[4] Ralph C Merkle. Method of providing digital signatures (1979). *URL: https://www. google. com/patents/US4309569 (visitado em abril de 2019)*.

[5] Eric Brewer. Cap twelve years later: How the "rules" have changed. *Computer*, 45(2):23–29, 2012.

[6] Takuro Nakagawa and Naohiro Hayashibara. Energy efficient raft consensus algorithm. In *International Conference on Network-Based Information Systems*, pages 719–727. Springer, 2017.

[7] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 4(3):382–401, 1982.

[8] Leslie Lamport. Generalized consensus and paxos. Technical report, Microsoft Research MSR-TR-2005-33, 2005.

[9] G. Wood. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper*, 2014.

[10] Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, Srinivasan Muralidharan, Chet Murthy, Binh Nguyen, Manish Sethi, Gari Singh, Keith Smith, Alessandro Sorniotti, Chrysoula Stathakopoulou, Marko Vukolić, Sharon Weed Cocco, and Jason Yellick. Hyperledger fabric: A distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference*, EuroSys '18, pages 30:1–30:15, New York, NY, USA, 2018. ACM.

[11] Don Tapscott and Alex Tapscott. *Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World*. Brilliance Audio, 2016.

[12] Lemuria Carter and Jolien Ubacht. Blockchain applications in government. In *Proceedings of the 19th Annual International Conference on Digital Government Research: Governance in the Data Age*, dg.o '18, pages 126:1–126:2, New York, NY, USA, 2018. ACM.

[13] Seyoung Huh, Sangrae Cho, and Soohyung Kim. Managing IoT devices using blockchain platform. In *Advanced Communication Technology (ICACT), 2017 19th International Conference on*, pages 464–467. IEEE, 2017.

[14] Arlindo F. da Conceição, Flavio S. Correa da Silva, Vladimir Rocha, Angela Locoro, and João Marcos M. Barguil. Eletronic health records using blockchain technology. *Workshop em Blockchain: Teoria, Tecnologias e Aplicações (WBlockchain, SBRC)*, 1(1/2018), 2018.

[15] Peng Zhang, Douglas C. Schmidt, Jules White, and Gunther Lenz. Blockchain technology use cases in healthcare. Advances in Computers. Elsevier, 2018.

[16] Cécile Monteil. Blockchain and health. In *Digital Medicine*, pages 41–47. Springer, 2019.

[17] A. Oram. *Peer-to-Peer: Harnessing the Power of Disruptive Technologies*. O'Reilly Media, 2001.

[18] Bram Cohen. The BitTorrent protocol specification, 2008.

[19] Salman Baset and Henning Schulzrinne. An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol. *Proceedings - IEEE INFOCOM*, 2005.

[20] Brendan McCallum. Digital dollars, masks, and black markets: The cypherpunk legacy. 2013.

[21] Nick Szabo. Smart contracts: Building blocks for digital markets. http://www.alamut.com/subj/economics/nick_szabo/smartContracts.html, 1996. Acessado: 28 Mar. 2019.

[22] Top 100 Criptomoedas por Capitalização de Mercado. https://coinmarketcap.com. Acessado: 22 Abr. 2019.

[23] I. Bashir. *Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained, 2nd Edition*. Packt Publishing, 2018.

[24] A.M. Antonopoulos. *Mastering Bitcoin: Programming the Open Blockchain*. O'Reilly Media, 2017.

[25] D.R. Stinson. *Cryptography: Theory and Practice, Third Edition*. Discrete Mathematics and Its Applications. Taylor & Francis, 2005.

[26] Ilya Mironov. Hash functions: Theory, attacks, and applications. Technical report, November 2005.

[27] Andrew S. Tanenbaum and Maarten van Steen. *Distributed Systems: Principles and Paradigms (2nd Edition)*.

[28] R. C. Merkle. Protocols for public key cryptosystems. In *1980 IEEE Symposium on Security and Privacy*, pages 122–122, April 1980.

[29] Vitalik Buterin. On public and private blockchains. *Ethereum blog*, 7, 2015.

[30] Elli Androulaki et al. Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains. In *Proceedings of the Thirteenth EuroSys Conference*, EuroSys '18, pages 30:1–30:15, New York, NY, USA, 2018. ACM.

[31] Brown, Carlyle, Grigg, & Hearn. *Corda: An Introduction*. https://docs.corda.net/_static/corda-introductory-whitepaper.pdf, 2016.

[32] K. Wüst and A. Gervais. Do you Need a Blockchain? In *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*, pages 45–54, June 2018.

[33] Leslie Lamport. Paxos made simple. *ACM SIGACT News (Distributed Computing Column) 32, 4 (Whole Number 121, December 2001)*, pages 51–58, December 2001.

[34] Christian Gorenflo, Stephen Lee, Lukasz Golab, and Srinivasan Keshav. FastFabric: Scaling Hyperledger Fabric to 20,000 Transactions per Second. *CoRR*, abs/1901.00910, 2019.

[35] Brasil. Ministério de Saúde. Política nacional de informação e informática em saúde. http://bvsms.saude.gov.br/bvs/publicacoes/politica_nacional_infor_informatica_saude_2016.pdf, 2016. Acessado: 28 Mar. 2019.

[36] Duane Bender and Kamran Sartipi. Hl7 fhir: An agile and restful approach to healthcare information exchange. In *Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems*, pages 326–331. IEEE, 2013.

[37] Srinath Perera, Frank Leymann, and Paul Fremantle. A use case centric survey of blockchain: *status quo* and future directions. *PeerJ Preprints*, 7:e27529v1, 2019.

[38] The Economist. Why bitcoin uses so much energy. https://www.economist.com/the-economist-explains/2018/07/09/why-bitcoin-uses-so-much-energy. Acessado: 21 de abril de 2019.

[39] Xiaoqi Li, Peng Jiang, Ting Chen, Xiapu Luo, and Qiaoyan Wen. A survey on the security of blockchain systems. *Future Generation Computer Systems*, 2017.

[40] Steve Davies. blockchain is here. what's your next move?

[41] Mohammad Dabbagh, Mehdi Sookhak, and Nader Sohrabi Safa. The evolution of blockchain: A bibliometric study. *IEEE Access*, 2019.

[42] Fran Casino, Thomas K Dasaklis, and Constantinos Patsakis. A systematic literature review of blockchain-based applications: current status, classification and open issues. *Telematics and Informatics*, 2018.

[43] Susel Góngora Alonso, Jon Arambarri, Miguel López-Coronado, and Isabel de la Torre Díez. Proposing new blockchain challenges in ehealth. *Journal of medical systems*, 43(3):64, 2019.

[44] Ariel Ekblaw, Asaph Azaria, John D Halamka, and Andrew Lippman. A case study for blockchain in healthcare:"medrec" prototype for electronic health records and medical research data. In *IEEE Open & Big Data Conference*, volume 13, page 13, 2016.

[45] You Sun, Rui Zhang, Xin Wang, Kaiqiang Gao, and Ling Liu. A decentralizing attribute-based signature for healthcare blockchain. In *2018 27th International Conference on Computer Communication and Networks (ICCCN)*, pages 1–9. IEEE, 2018.

[46] Alevtina Dubovitskaya, Zhigang Xu, Samuel Ryu, Michael Schumacher, and Fusheng Wang. Secure and trustable electronic medical records sharing using blockchain. In *AMIA Annual Symposium Proceedings*, volume 2017, page 650. American Medical Informatics Association, 2017.

[47] Anuraag A Vazirani, Odhran O'Donoghue, David Brindley, and Edward Meinert. Implementing blockchains for efficient health care: Systematic review. *Journal of medical Internet research*, 21(2):e12439, 2019.

[48] Paul Voigt and Axel Von dem Bussche. The EU General Data Protection Regulation (GDPR). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 2017.

[49] Yu Rang Park, Eunsol Lee, Wonjun Na, Sungjun Park, Yura Lee, and Jae-Ho Lee. Is blockchain technology suitable for managing personal health records? mixed-methods study to test feasibility. *Journal of medical Internet research*, 21(2):e12533, 2019.

[50] Thomas McGhin, Kim-Kwang Raymond Choo, Charles Zhechao Liu, and Debiao He. Blockchain in healthcare applications: Research challenges and opportunities. *Journal of Network and Computer Applications*, 2019.

[51] Deepak K Tosh, Sachin Shetty, Xueping Liang, Charles A Kamhoua, Kevin A Kwiat, and Laurent Njilla. Security implications of blockchain cloud with analysis of block withholding attack. In *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pages 458–467. IEEE Press, 2017.

[52] Asaph Azaria, Ariel Ekblaw, Thiago Vieira, and Andrew Lippman. Medrec: Using blockchain for medical data access and permission management. In *Open and Big Data (OBD), International Conference on*, pages 25–30. IEEE, 2016.

[53] Matthias Mettler. Blockchain technology in healthcare: The revolution starts here. In *2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom)*, pages 1–3. IEEE, 2016.

[54] Alex Roehrs, Cristiano André da Costa, and Rodrigo da Rosa Righi. Omniphr: A distributed architecture model to integrate personal health records. *Journal of biomedical informatics*, 71:70–81, 2017.

[55] Jie Zhang, Nian Xue, and Xin Huang. A secure system for pervasive social network-based healthcare. *IEEE Access*, 4:9239–9250, 2016.

[56] Mayra Samaniego and Ralph Deters. Hosting virtual iot resources on edge-hosts with blockchain. In *2016 IEEE International Conference on Computer and Information Technology (CIT)*, pages 116–119. IEEE, 2016.

[57] Muhammad Siddiqi, Syed Taha All, and Vijay Sivaraman. Secure lightweight context-driven data logging for bodyworn sensing devices. In *2017 5th International Symposium on Digital Forensic and Security (ISDFS)*, pages 1–6. IEEE, 2017.

[58] QI Xia, Emmanuel Boateng Sifah, Kwame Omono Asamoah, Jianbin Gao, Xiaojiang Du, and Mohsen Guizani. Medshare: Trust-less medical data sharing among cloud service providers via blockchain. *IEEE Access*, 5:14757–14767, 2017.

[59] Zonyin Shae and Jeffrey JP Tsai. On the design of a blockchain platform for clinical trial and precision medicine. In *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, pages 1972–1980. IEEE, 2017.

[60] Xiao Yue, Huiju Wang, Dawei Jin, Mingqiang Li, and Wei Jiang. Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control. *Journal of medical systems*, 40(10):218, 2016.

[61] Liam Bell, William J Buchanan, Jonathan Cameron, and Owen Lo. Applications of blockchain within healthcare. *Blockchain in Healthcare Today*, 2018.

[62] Hilarie Orman. Blockchain: the emperors new PKI? *IEEE Internet Computing*, 22(2):23–28, 2018.

[63] Maria José Amaral Salomi and Rafael Fabio Maciel. Gestão de documentos e automação de processos em uma instituição de saúde sem papel. *Journal of Health Informatics*, 8(1), 2016.

[64] Mehdi Benchoufi and Philippe Ravaud. Blockchain technology for improving clinical research quality. *Trials*, 18(1):335, 2017.

[65] Alexandre Marinho. A crise do mercado de planos de saúde: devemos apostar nos planos populares ou no SUS? *Planejamento e Políticas Públicas*, (49), 2017.

[66] Patrick Kierkegaard. E-prescription across europe. *Health and Technology*, 3(3):205–219, 2013.

[67] J. Kreps, N. Narkhede, and J. Rao. Kafka: A distributed messaging system for log processing. In *Proceedings of 6th International Workshop on Networking Meets Databases (NetDB), Athens, Greece*, 2011.

[68] Patrick Hunt, Mahadev Konar, Flavio P. Junqueira, and Benjamin Reed. Zookeeper: Wait-free coordination for internet-scale systems. In *Proceedings of the 2010 USENIX Conference on USENIX Annual Technical Conference*, pages 1–14, 2010.