

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

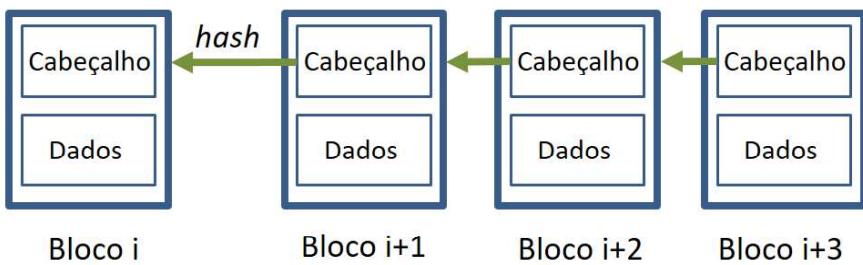


Figura 2.1. Estrutura básica de uma Blockchain

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 válido, 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. Segundo 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 - 1^a geração

Os primeiros estudos sobre 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 surgiiram 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 - 2^a 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.

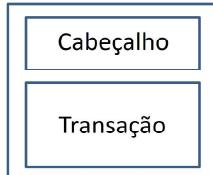


Figura 2.2. Bloco

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.

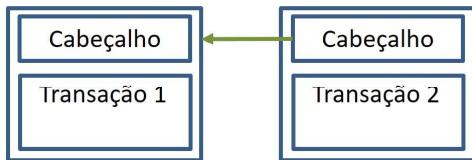


Figura 2.3. Cadeia de Blocos

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

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

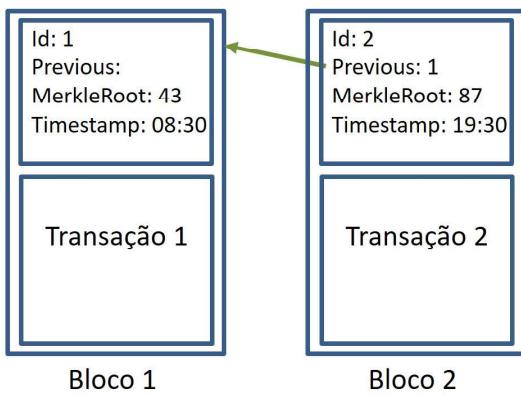


Figura 2.4. Cadeia de Blocos com cabeçalho

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 gênese.

Em um sistema Blockchain, o primeiro bloco da cadeia é denominado de bloco gênese 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 gênese.

Até aqui, o exemplo do segundo cenário poderá ser complementado com o bloco gênese, 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,

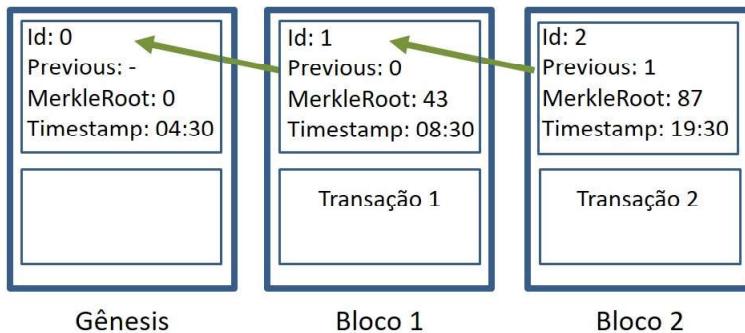


Figura 2.5. Cadeia de Blocos com gênesis

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 vindos 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 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”.

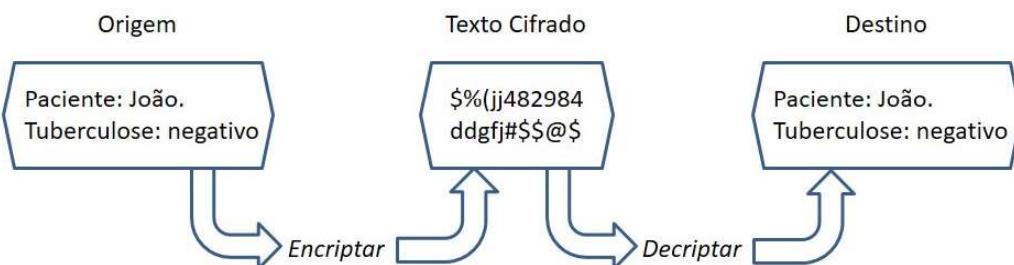


Figura 2.7. Aplicação da Criptografia

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.

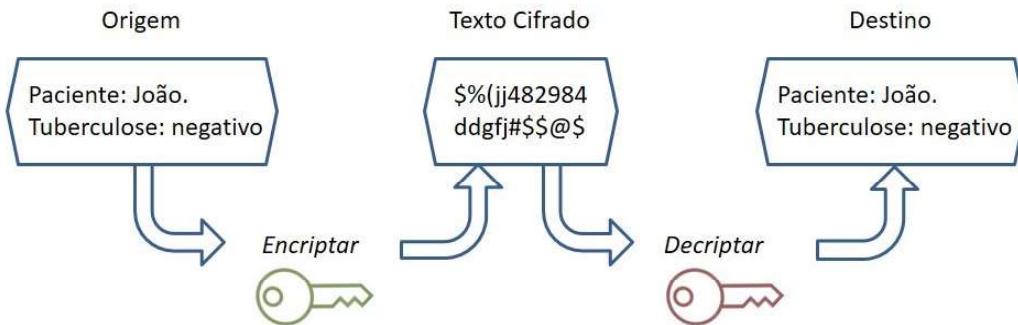


Figura 2.8. Fluxo de criptografia assimétrica

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.

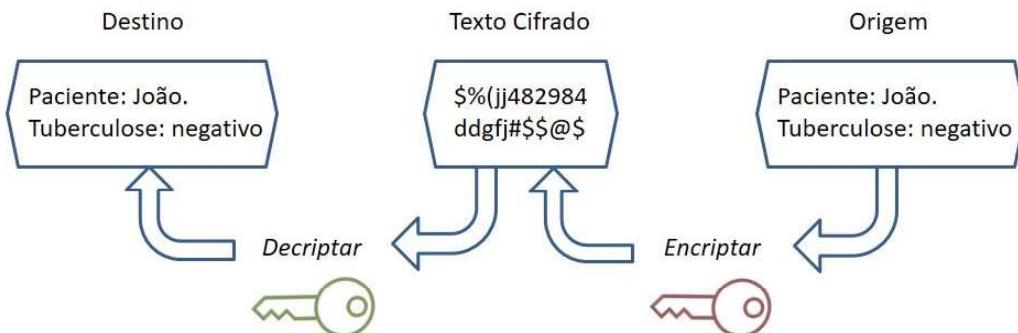


Figura 2.9. Fluxo inverso de criptografia assimétrica

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-la 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).

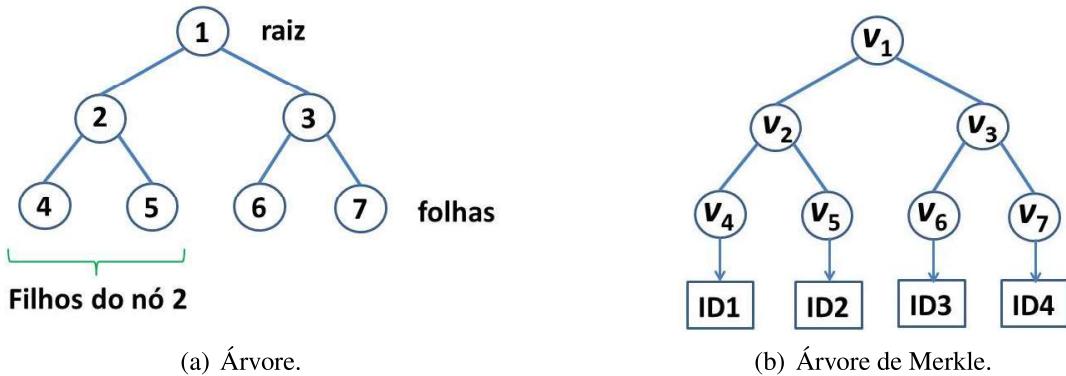


Figura 2.10. Exemplos de Árvores.

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 Blockchain 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

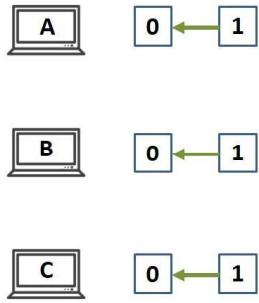


Figura 2.11. Consenso inicial

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).

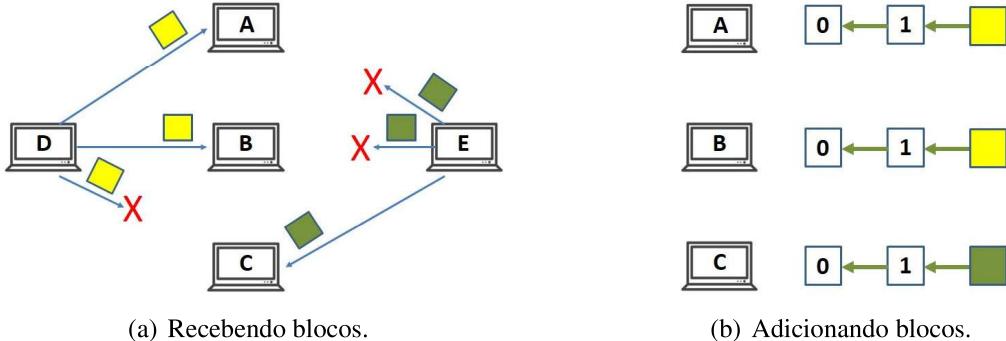


Figura 2.12. Consenso intermediário.

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 possíveis 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).

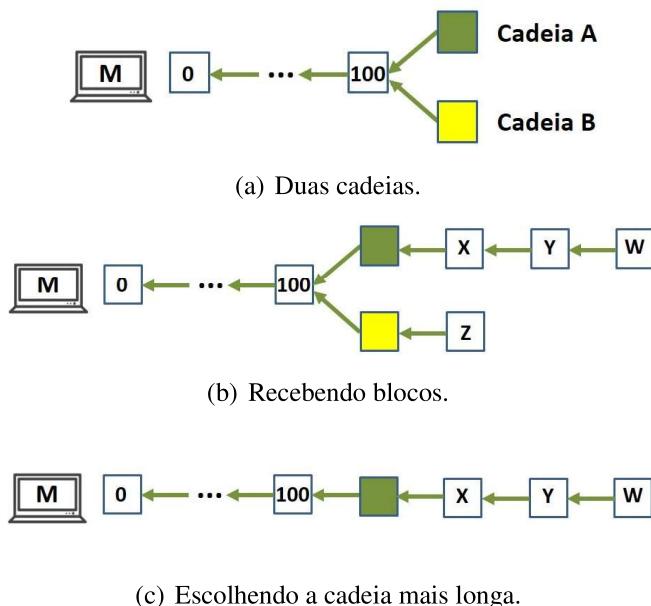


Figura 2.13. Consenso de qual bloco inserir.

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 Bitcoin³ e Ethereum⁴ 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.

³Vide <https://bitnodes.earn.com/>

⁴Vide <https://ethstats.net/>