Compartilhamento Seguro de Arquivos de Saúde usando Criptografia Baseada em Atributos e Redes Descentralizadas

Leonardo da Costa[1], **Billy Pinheiro**[1], **Roberto Araújo**[1], **Antônio Abelém**[1]

[1]Faculdade de Computação – Universidade Federal do Pará (UFPA)
Belém – PA – Brasil

{lbc,billy,rsa,abelem}@ufpa.br

Abstract. *Cloud computing allows on demand storage and sharing of records with high degree of availability. However, storing a health record in a cloud provider requires trusting it for the record security. By mitigating it, current approaches focus on confidentiality and access control. They do not adequately treat data integrity. This paper presents Decentralized Sharing of Health Records (DSHR) protocol, which employs attribute-based cryptography and decentralized networks for secure sharing of health records. The solution treats confidentiality, access control and integrity of records. A DSHR proof of concept was implemented and load tests were executed in order to demonstrate its feasibility.*

Resumo. *A computação em nuvem possibilita o armazenamento e o compartilhamento de arquivos sob demanda com alta taxa de disponibilidade para a área da saúde. Contudo, utilizar um provedor de nuvem para armazenar um arquivo de saúde significa confiar a ele a segurança do arquivo. Ao mitigar isso, as abordagens da literatura preocupam-se apenas com a confidencialidade e o controle de acesso, não tratando adequadamente a integridade dos dados. Esse trabalho apresenta o protocolo Decentralized Sharing of Health Records (DSHR), que utiliza criptografia baseada em atributos e redes descentralizadas para o compartilhamento seguro de arquivos de saúde, tratando a confidencialidade, o controle de acesso e a integridade dos arquivos. Uma prova de conceito do DSHR foi implementada e testes de carga foram executados para demonstrar a sua viabilidade prática.*

1. Introdução

A implantação e a manutenção de sistemas de armazenamento em grande escala apresentam altos custos. Por esse motivo, vários tipos de usuários fazem uso de sistemas de armazenamento terceirizados. A computação em nuvem é um exemplo de paradigma que provê esse tipo de serviço. O paradigma é amplamente popularizado e indispensável para muitas aplicações [Puthal et al. 2015]. A área da saúde tem usufruído dos benefícios da computação em nuvem [Abbas and Khan 2014]. Através de provedores de nuvem, arquivos de saúde de pacientes (e.g. prescrições, resultados de exames) são facilmente armazenados e compartilhados com colaboradores, e.g. médicas(os) e enfermeiras(os), aumentando a disponibilidade dos arquivos e reduzindo a carga de trabalho para administrá-los.

Arquivos de saúde armazenados em nuvem são comumente classificados em registros médicos eletrônicos e registros médicos pessoais [Serrão and Cardoso 2017]. Os

primeiros são arquivos oriundos de várias fontes (e.g. médicos, laboratórios) e fornecem uma visão ampla do estado médico do paciente. Por sua vez, os últimos contém os mesmos tipos de conteúdos que os registros médicos eletrônicos, porém são pensados para serem elaborados e administrados estritamente pelo paciente de modo privado e seguro, podendo incluir também informações fornecidas pelo próprio paciente.

Apesar de benéfica, a computação em nuvem introduz desafios de segurança quanto ao compartilhamento de arquivos de saúde [Dawoud and Altilar 2017]. Quando estes são enviados à um provedor de nuvem, o controle dos pacientes sobre os arquivos diminui. Se mecanismos para assegurar a confidencialidade e o controle de acesso não forem empregados, um provedor malicioso pode dar acesso aos arquivos à entidades não autorizadas. O acesso indevido aos arquivos dos pacientes pode levar ao uso de dados médicos em, por exemplo, propagandas e pesquisas de mercado [de Melo Silva et al. 2014]. A integridade dos arquivos na nuvem é uma outra questão de segurança. Um provedor malicioso pode indevidamente alterar ou apagar arquivos armazenados, o que poderia, por exemplo, prejudicar diagnósticos e o tratamento de pacientes.

Esse trabalho apresenta o *Decentralized Sharing of Health Records* (DSHR), um protocolo para compartilhamento seguro de arquivos de saúde que apresenta como principal contribuição a garantia da integridade dos arquivos, que envolve a alteração e a deleção indevida destes. Para isso, o DSHR substitui um provedor único de nuvem por redes descentralizadas. Emprega-se três tecnologias base para garantir segurança e escalabilidade de armazenamento: criptografia baseada em atributos (CBA), redes IPFS e *blockchain*. O DSHR também possibilita revogação de acesso, o que exclui o direito de acesso à um arquivo, e garante o não repúdio, impedindo que colaboradores neguem ter gerado um arquivo de saúde. Uma prova de conceito do DSHR foi implementada e testes de carga mostraram sua viabilidade prática. No melhor do conhecimento dos autores, essa é a primeira abordagem que utiliza em conjunto CBA, IPFS e *blockchain* para propiciar segurança em sistemas de informação de saúde.

Esse trabalho está organizado da seguinte forma. A Seção 2 introduz as tecnologias empregadas pelo DSHR. Após isso, a Seção 3 discute trabalhos relacionados. A Seção 4 apresenta o DSHR em detalhes, enquanto que a Seção 5 o avalia. Finalmente, a Seção 6 apresenta conclusões e aponta trabalhos futuros.

2. Preliminares

Para garantir a segurança do compartilhamento de arquivos de saúde, o DSHR utiliza três tecnologias. Criptografia é usada para que os arquivos sejam armazenados de modo confidencial e compartilhados com dados colaboradores. Redes descentralizadas de registro de transações (*blockchain*) são empregadas para registrar arquivos e impedir a alteração indevida destes. Redes descentralizadas de armazenamento em grande escala (IPFS) são empregadas para que os arquivos sejam armazenados em vários locais físicos, tornando-se inviável a deleção indevida dos mesmos. Essas tecnologias são apresentadas a seguir.

2.1. Criptografia Baseada em Atributos com Política no Texto Criptografado

Criptografia baseada em atributos com política no texto criptografado (CBA) [Bethencourt et al. 2007] é uma técnica que associa chaves criptográficas de usuários à atributos e usa uma política de acesso para criptografar dados. Essa política contém uma

estrutura de acesso composta por atributos descritivos relacionados através de operadores lógicos (e.g. AND, OR) que determinam quem pode acessar os dados em texto claro. Por exemplo, a estrutura (“Diretor”OR (“Médico”AND “Cirurgião”)) determina que apenas usuários com perfil de diretor ou de médico cirurgião têm acesso à conteúdos criptografados com tal política. CBA é nativamente planejada para a aplicação de controle de acesso.

Em CBA, uma autoridade de atributos (AA) confiável é responsável por gerar os parâmetros globais do sistema, e suas chaves pública e mestra. Esses dados são usados para gerar chaves secretas para usuários contendo atributos condizentes com seus perfis. Uma mensagem é criptografada com a utilização de uma política de acesso baseada em atributos. Para descriptografar um texto criptografado, os atributos de uma chave secreta devem satisfazer a política de acesso usada na criptografia.

2.2. *Blockchain*

Blockchain é uma tecnologia promissora, tendo como uma de suas aplicações mais populares o Bitcoin e abrindo uma gama de direções para outras aplicações. Uma *blockchain* é uma rede descentralizada que processa e armazena transações [Bonneau et al. 2015]. Uma transação é gerada através de assinaturas digitais e envolve o registro de um recurso para um endereço ou a transferência do recurso entre endereços. Um endereço na *blockchain* refere-se à uma chave pública de um usuário. Cada usuário gera um par de chaves de assinatura. Para emitir uma transação, um usuário deve assiná-la com a sua chave privada e, no caso de uma transferência, especificar o endereço de destino da transação.

Assim que uma transação é submetida à rede, os *miners*, nodos específicos, são encarregados de adicioná-la em uma estrutura de dados chamada bloco. Quando um bloco é formado, os *miners* tentam uni-lo ao último bloco anteriormente formado, criando-se uma cadeia. Para unir os blocos, os *miners* trabalham em uma tarefa computacionalmente difícil de se resolver. O primeiro a resolvê-la recebe uma recompensa (geralmente, em criptomoedas) e repassa a nova cadeia aos outros nodos. Todos os nodos conhecem a cadeia mais recentemente criada (isto é, a mais longa) e são capazes de validá-la. Para isso, basta verificar as ligações entre os blocos na cadeia. Estima-se que modificar um bloco não é praticável se a maioria do poder de processamento da rede for controlado por nodos honestos. Essa premissa garante a segurança da rede e a integridade das transações.

2.3. *InterPlanetary File System*

InterPlanetary File System (IPFS) [Benet 2014] é uma tecnologia que provê redes descentralizadas onde todos os nodos participantes possuem o mesmo sistema de arquivos. Dados armazenados em uma rede IPFS são mantidos localmente em vários nodos, eliminando a existência de pontos únicos de falha. Cada arquivo armazenado é endereçado pelo seu valor *hash*, o que o torna único e provê um processo de busca eficiente de arquivos.

Por ser uma rede descentralizada, o IPFS é usado nesse trabalho para permitir o armazenamento de arquivos de saúde em grande escala, já que armazenar grandes quantidades de dados em *blockchains* é inviável devido aos altos custos tanto de processamento e energia quanto de criptomoedas para o fazê-lo.

3. Trabalhos Relacionados

O emprego de CBA para o armazenamento e compartilhamento de arquivos de saúde na nuvem tem sido observado em vários trabalhos da literatura, diferentemente da utilização

conjunta de CBA e IPFS. Além disso, *blockchain* ainda encontra um número limitado de aplicações na área da saúde. De forma à introduzir uma visão geral do estado da arte, apresenta-se a seguir trabalhos que visam o compartilhamento seguro de dados de saúde.

No trabalho de [de Melo Silva et al. 2014], foi proposta uma arquitetura para compartilhamento de arquivos de saúde baseada em identidades federadas, nuvens e CBA. A arquitetura retira dos pacientes toda a complexidade para gerenciamento de atributos e chaves criptográficas, provendo fácil usabilidade. Contudo, um controle excessivo sobre a identidade e os atributos dos pacientes é dado aos provedores de identidade, o que torna a arquitetura vulnerável à ataques advindos destes. Ao contrário, o DSHR provê maior segurança ao delegar à autoridades de atributo apenas o gerenciamento inicial de atributos e depois atribuir aos pacientes total controle sobre seus atributos, chaves e arquivos.

A proposta de [Au et al. 2017] apresentou como principal contribuição um mecanismo para compartilhar arquivos de saúde na nuvem entre países com diferentes legislações. Para isso, CBA é utilizado. Diferentemente do DSHR, entretanto, a proposta de [Au et al. 2017] não considera ataques dos provedores de nuvem e de outras entidades maliciosas quanto à integridade dos arquivos. O protocolo para dispositivos móveis baseado em CBA, introduzido por [Liu et al. 2018], gera a maior parte de um texto criptografado em modo *offline*. Com isso, ao estarem *online* enviando arquivos à nuvem, os dispositivos executam computações criptográficas mais leves, economizando bateria. Contudo, a proposta introduz complexidade ao considerar que o paciente é responsável por administrar um conjunto de atributos distribuídos à colaboradores. Além disso, ela assume que a nuvem é confiável, não deletando arquivos e mantendo-os íntegros.

A proposta de [Li et al. 2017] oferece significativa contribuição ao prover um sistema de informação de saúde que armazena textos criptografados de CBA com tamanho constante, não importando o tamanho do arquivo. Em contrapartida, além de utilizarem uma versão de CBA que introduz maior complexidade computacional aos dispositivos dos usuários, [Li et al. 2017] também não apresentam qualquer estratégia para impedir a alteração ou deleção de arquivos em provedores de nuvem maliciosos.

No melhor do conhecimento dos autores, [Rahulamathavan et al. 2017] apresentaram o único trabalho da literatura que usa CBA e *blockchain* para garantir o compartilhamento seguro de dados de saúde. Contudo, a proposta é limitada, tendo em vista que as atuais plataformas de *blockchain* não suportam o armazenamento de arquivos grandes ou proveem essa funcionalidade através da cobrança de altas taxas de recompensa para a inserção dos arquivos. A proposta de [Azaria et al. 2016] supre essa limitação ao usar *blockchain* e nuvens para compartilhar arquivos. Entretanto, a proposta é incompleta por não considerar a segurança dos arquivos na nuvem.

Apesar de existirem várias propostas que empregam CBA ou *blockchain* na área da saúde, a maioria apresenta limitações quanto à segurança, não consideram o problema da alteração/deleção indevida de arquivos, ou não apresentam escalabilidade de armazenamento. O presente trabalho introduz o DSHR, que considera todos esses aspectos. Na próxima seção, a solução é apresentada.

4. *Decentralized Sharing of Health Records* (DSHR)

Como descrito, as propostas da literatura para compartilhamento seguro de arquivos de saúde apresentam limitações, principalmente relacionadas à escalabilidade de armazena-

mento e integridade dos arquivos, que envolve a alteração e a deleção indevida destes. A fim de contornar essas limitações, apresenta-se a seguir o DSHR, um protocolo que emprega *blockchain*, IPFS e CBA para prover tanto confidencialidade e controle de acesso quanto integridade de arquivos de saúde, sem limitar a escalabilidade de armazenamento.

O DSHR é composto pelos seguintes participantes: um conjunto de autoridades certificadoras online (ACOs), que certificam chaves públicas de assinatura, um conjunto de autoridades de atributos (AAs), uma rede de *blockchain* e uma rede IPFS com um conjunto de nodos, e um conjunto de usuários, que podem desempenhar o papel de paciente ou colaborador. Para fins de simplicidade e melhor entendimento do DSHR, será considerado apenas uma ACO e uma AA. Entretanto, assume-se que esses elementos não necessitam ser confiáveis, tendo em vista que os serviços de ambos podem ser distribuídos entre várias entidades. Todos os participantes devem seguir o protocolo corretamente. Caso contrário, a detecção de ações maliciosas pelos métodos de segurança do DSHR podem levar os participantes a serem excluídos do sistema. Por fim, considera-se o uso de um canal de comunição seguro (e.g. TLS) para troca de mensagens entre os participantes.

Modelo de ataque. Considera-se a existência de nodos, da *blockchain* e da rede IPFS, e usuários maliciosos. Estes agem de maneira maliciosa ao tentar indevidamente ler, alterar ou apagar os arquivos de saúde armazenados na rede IPFS sem serem detectados. O objetivo do DSHR é garantir que tais comportamentos sejam detectados e/ou impedidos.

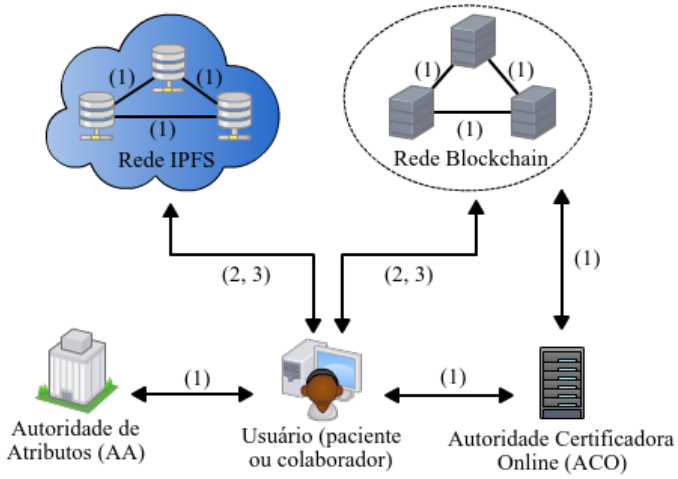


Figura 1. Visão geral do DSHR.

Visão geral. A Figura 1 ilustra a visão geral do DSHR com suas etapas entre parênteses. Ele inicia em uma etapa de configuração (etapa 1). Nela, a AA gera os parâmetros de CBA, uma rede de *blockchain* e uma rede IPFS são inicializadas, a ACO gera o seu par de chaves de assinatura e usuários cadastram-se para utilizar o sistema, obtendo suas respectivas chaves de CBA e de assinatura. Pacientes cadastrados podem então executar a etapa de armazenamento de um arquivo de saúde (etapa 2). Um arquivo de um paciente pode ser gerado por um colaborador ou pelo próprio paciente. No primeiro caso, o colaborador gerador assina o *hash* do arquivo e repassa-o ao paciente. O paciente criptografa o arquivo e a assinatura do colaborador, e armazena o texto criptografado resultante na rede IPFS. Além disso, a URL do arquivo na rede IPFS e os valores *hash* do arquivo e do texto criptografado são assinados pelo paciente e publicados como uma transação na *blockchain*.

Em caso de o próprio paciente gerar um arquivo, segue-se os mesmos procedimentos para armazená-lo, com exceção da assinatura do colaborador no arquivo, que não existe.

Para acessar o arquivo na etapa de compartilhamento (etapa 3), um colaborador requisitante recupera da *blockchain* a transação referente ao arquivo, verifica a assinatura do paciente gerada nela e usa a URL contida na transação para transferir o texto criptografado da rede IPFS para o seu dispositivo. O colaborador então tenta descriptografar o texto criptografado usando a sua chave secreta de CBA, obtendo o arquivo original e a assinatura do colaborador gerador se os seu atributos satisfizerem a política de acesso usada na criptografia. Se a assinatura do colaborador gerador existe, o colaborador requisitante verifica-a e, em seguida, checa a integridade do arquivo. Para isso, ele extrai da transação recuperada da *blockchain* os valores *hash* referentes ao arquivo e os compara com os valores *hash* do texto criptografado e do arquivo original obtidos da rede IPFS. Se as duas comparações são verdadeiras, a integridade do arquivo foi mantida, indicando que o colaborador pode utilizá-lo. O restante da seção apresenta os detalhes do DSHR.

4.1. Configuração

Na etapa de configuração, os elementos do DSHR são configurados e o material criptográfico requerido é gerado. As partes interessadas em administrar em conjunto a ACO e a AA (e.g. provedores de plano de saúde, órgãos governamentais de saúde) devem, via contrato, estabelecê-las e determinar as entidades responsáveis por elas. Os atributos de CBA que a AA administrará são também definidos (e.g. um atributo pode referir-se à um plano de saúde ou à vários ao mesmo tempo). Com a ACO e a AA definidas, gera-se o par de chaves de assinatura PK_{ACO} e SK_{ACO} da ACO, chaves pública e privada, respectivamente, e executa-se o algoritmo CONFIGURAÇÃOAA, descrito a seguir.

CONFIGURAÇÃOAA(λ): Esse algoritmo é executado uma vez para uma AA e tem como entrada um parâmetro de segurança λ. Ele gera como saída os parâmetros globais PS do sistema de CBA, e as chaves pública PK_{CBA} e mestra MK_{CBA} de CBA da AA.

Após a execução do algoritmo, uma rede de *blockchain* e uma rede IPFS são inicializadas. Um mesmo nodo pode compor tanto a primeira quanto a segunda. Podem desempenhar o papel de nodo, por exemplo, os dispositivos de usuários, provedores de plano de saúde e provedores de nuvem. Como a *blockchain* irá registrar dados referentes à arquivos de saúde, considerados amplamente privados, uma *blockchain* privada é empregada. Dentre as características comuns dessa categoria de *blockchain*, está que apenas usuários autorizados podem participar da rede e ter acesso aos dados registrados. Os parâmetros PS e as chaves públicas PK_{ACO} e PK_{CBA} são dados públicos e compõem os primeiros blocos da *blockchain*, podendo ser acessados por todos os usuários cadastrados.

A etapa de configuração é finalizada com o cadastramento dos usuários, que pode, por exemplo, ser realizado por provedores de planos de saúde ou por órgãos governamentais. No cadastramento, um paciente tem o seu identificador ID_{pac} e o seu par de chaves de assinatura (PK_{pac} e SK_{pac}) gerados. A ACO assina a chave pública PK_{pac} usando SK_{ACO} e a publica na *blockchain* junto à ID_{pac}. O paciente também solicita à AA a execução do algoritmo GERAÇÃOCHAVECBA (descrito a seguir), que gera uma chave secreta K_{pac} de CBA para o paciente. Para um colaborador, segue-se o mesmo fluxo de cadastro de um paciente. Os dados de um colaborador são denotados por: ID_{col}, PK_{col}, SK_{col} e K_{col}.

GERAÇÃOCHAVECBA(ID_u, PS, Attr, PK_{CBA}, MK_{CBA}): Esse algoritmo recebe como

entrada o identificador ID_u de um usuário, os parâmetros globais PS de CBA, um conjunto de atributos Attr e as chaves PK_{CBA} e MK_{CBA} da AA. Ele gera como saída uma chave secreta de CBA para um usuário contendo o identificador ID_u e os atributos em Attr.

4.2. Armazenando um Arquivo de Saúde

Após a configuração, segue-se a etapa de armazenamento de um arquivo de saúde. Considera-se duas maneiras para a geração de um arquivo de um paciente: por um colaborador (arquivo terceiro) ou pelo próprio paciente (arquivo pessoal). Os procedimentos para armazenar um arquivo terceiro AS iniciam com um colaborador. Ao gerar AS para um paciente, o colaborador gerador computa o valor *hash* VH_{AS} de AS e uma assinatura SIG^{G}_{col} de VH_{AS} usando sua chave privada SK^{G}_{col}. O colaborador gerador envia os valores VH_{AS} e SIG^{G}_{col} para o paciente. Este dá sequência ao armazenamento executando o algoritmo CRIPTOGRAFIACBA, que criptografa AS e SIG^{G}_{col}. O algoritmo é descrito a seguir.

CRIPTOGRAFIACBA(AS, SIG^{G}_{col}, P_{CBA}, PS, PK_{CBA}): Esse algoritmo recebe como entrada um arquivo de saúde AS, uma assinatura SIG^{G}_{col} de um colaborador (parâmetro opcional), uma política de acesso baseada em atributos P_{CBA}, os parâmetros globais PS de CBA e a chave pública PK_{CBA} da AA. Ele criptografa AS e SIG^{G}_{col} (se passado como entrada) usando a política P_{CBA} e fornece como saída o texto criptografado CT.

Ao obter CT, o paciente envia-o para a rede IPFS, que o armazena e retorna ao paciente a URL para obter o arquivo na rede. O paciente então usa VH_{AS}, CT, a URL e sua chave privada SK_{pac} no algoritmo CRIAÇÃOTX, descrito a seguir.

CRIAÇÃOTX(VH_{AS}, CT, URL, SK_{pac}): Esse algoritmo recebe um valor *hash* VH_{AS} de um AS, um texto criptografado CT, a URL de CT na rede IFPS e a chave privada de assinatura SK_{pac} do paciente. Ele computa o valor *hash* VH_{CT} de CT, cria uma transação TX, que inclui URL, VH_{AS} e VH_{CT} (onde VH_{CT} é o identificador do arquivo), e gera uma assinatura SIG_{pac} de TX usando SK_{pac}. O algoritmo fornece como saída TX e SIG_{pac}.

Com a execução de CRIAÇÃOTX, o paciente tem em posse os valores TX e SIG_{pac}. Ele então envia esses dados para os *miners* da *blockchain*, que verificam se SIG_{pac} é uma assinatura válida de um usuário cadastrado (através da chave pública PK_{pac} deste). Em caso afirmativo, os *miners* inserem os dados em um novo bloco. Note-se que são publicados na *blockchain* tanto o valor *hash* VH_{AS} do arquivo original quanto o *hash* VH_{CT} do arquivo criptografado. Isso é necessário para manter a integridade do arquivo na rede IPFS e prover revogação de acesso, como será explicado mais adiante.

Os procedimentos para armazenar um arquivo pessoal são idênticos aos realizados para um arquivo terceiro. Contudo, como o primeiro é gerado pelo paciente, ele não é assinado por uma colaborador e, por isso, a assinatura SIG^{G}_{col} não é dada como entrada à CRIPTOGRAFIACBA. Assim, o texto criptografado CT, armazenado na rede IPFS, conterá apenas o arquivo AS. Armazenar CT nessa rede impede um único nodo malicioso de deletar o arquivo da rede, diferentemente de abordagens de nuvens, que são gerenciadas por uma única entidade. Se um nodo IPFS honesto mantém CT, este pode ser recuperado.

4.3. Compartilhando um Arquivo de Saúde

Criptografar o seu arquivo com uma política de CBA garante ao paciente confidencialidade e controle de acesso. A relação de atributos definida na política impõe segurança no

próprio arquivo, dispensando-se a delegação da função do controle de acesso. No DSHR, o colaborador precisa de uma chave de CBA apropriada para acessar o arquivo. Os procedimentos para acessar arquivos terceiros e pessoais são os mesmos, diferenciando-se pela existência ou não da assinatura SIG_{col}^{G} do colaborador gerador. Eles são descritos a seguir.

Com as informações do novo arquivo publicadas na *blockchain*, colaboradores podem solicitar acesso a ele. Um colaborador requisitante primeiramente realiza uma busca na *blockchain* da transação TX que contém as informações mais atualizadas do arquivo criptografado de interesse através do *hash* VH_{CT} do mesmo. Esse *hash* poderia, por exemplo, ser informado ao colaborador pelo próprio paciente. O paciente deve informar também a sua chave pública PK_{pac} e a chave pública PK_{col}^{G} do colaborador que gerou o arquivo (em caso de um arquivo terceiro), para que o colaborador requisitante possa verificar as assinaturas do arquivo geradas. O colaborador requisitante deve verificar a assinatura da ACO em PK_{pac} e PK_{col}^{G} usando PK_{ACO} para validar as chaves.

Ao obter TX e sua respectiva assinatura SIG_{pac} gerada pelo paciente, o colaborador requisitante primeiramente verifica SIG_{pac} usando PK_{pac}, o que indica se a transação foi de fato gerada pelo paciente. Em caso afirmativo, o colaborador extrai de TX a URL que aponta a localização de CT na rede IPFS e transfere CT para o seu dispositivo. Em seguida, ele tenta descriptografar CT empregando a sua chave secreta K_{col}^{R} de CBA no algoritmo DESCRIPTOGRAFIACBA, descrito a seguir.

DESCRIPTOGRAFIACBA(PS, CT, K_{col}^{R}): Esse algoritmo tem como entrada os parâmetros PS de CBA, um texto criptografado CT e uma chave secreta K_{col}^{R} de CBA. Ele descriptografa CT usando K_{col}^{R}, retornando o arquivo original AS e a assinatura SIG_{col}^{G} de VH_{AS} do colaborador gerador (em caso de um arquivo terceiro) se a chave secreta satisfizer a política usada em CRIPTOGRAFIACBA. Caso contrário, ele retorna um valor aleatório.

Se o colaborador requisitante não for capaz de obter AS e SIG_{col}^{G}, a etapa finaliza com um erro. Em caso de sucesso, ele verifica a assinatura SIG_{col}^{G} usando a chave pública PK_{col}^{G} do colaborador gerador (em caso de um arquivo terceiro). Isso garante que o arquivo de saúde foi gerado pelo colaborar informado pelo paciente. Além disso, essa verificação garante o não repúdio (que o colaborador gerador do arquivo não pode negar tê-lo gerado). Em seguida, o colaborador requisitante executa o algoritmo VERIFICAÇÃOAS (descrito a seguir) para verificar a integridade do arquivo AS obtido da rede IPFS.

VERIFICAÇÃOAS(AS, CT, TX): este recebe como entrada o arquivo AS obtido com DESCRIPTOGRAFIACBA, o texto criptografado CT obtido da rede IPFS e a transação TX obtida da *blockchain*. O algoritmo extrai de TX os valores *hash* VH_{AS} e VH_{CT}, computa os valores *hash* VH'_{AS} de AS e VH'_{CT} de CT e verifica se $\text{VH}_{AS} = \text{VH}'_{AS}$ e $\text{VH}_{CT} = \text{VH}'_{CT}$. Ele retorna um valor booleano indicando se as duas igualdades são verdadeiras.

A igualdade $\text{VH}_{AS} = \text{VH}'_{AS}$ indica ao colaborador requisitante que o arquivo original enviado pelo paciente para a rede IPFS é o mesmo arquivo obtido no seu dispositivo. Já a igualdade $\text{VH}_{CT} = \text{VH}'_{CT}$ informa que o texto criptografado recebido da rede IPFS é o que contém a política mais nova usada pelo paciente para criptografar o arquivo. Assim, o colaborador pode identificar se o nodo que disponibilizou CT está entregando a versão mais nova do texto criptografado. Em caso negativo, o colaborador pode alertar os outros nodos da rede sobre tal ação maliciosa, o que poderia levar à exclusão do nodo. Se a integridade do arquivo foi mantida, o colaborador então utiliza o arquivo para os devidos fins.

4.4. Revogação de Acesso

A revogação de acesso é uma funcionalidade altamente desejável em sistemas de compartilhamento. Ela permite que o dono do arquivo retire a permissão de acesso de determinados usuários. Em sistemas de saúde, isso aplica-se, por exemplo, à quando um paciente muda de clínica e dá permissão de acesso à novos colaboradores, removendo a de antigos.

No DSHR, visando permitir acesso à um arquivo AS à novos colaboradores e retirar o acesso de colaboradores antigos, um paciente elabora uma nova política de acesso baseada em atributos que não seja satisfeita pelos atributos dos colaboradores antigos, porém que corresponda corretamente aos atributos de colaboradores novos e remanescentes. O paciente então realiza os procedimentos descritos na Seção 4.2, isto é, executa os algoritmos CRIPTOGRAFIACBA e CRIAÇÃOTX usando os mesmos AS e VH_{AS} e uma nova política P_{CBA}, o que resulta em um novo CT e em uma nova TX contendo VH_{CT}, VH_{AS} e URL. Ele envia o novo CT à rede IPFS e a nova transação TX à *blockchain*.

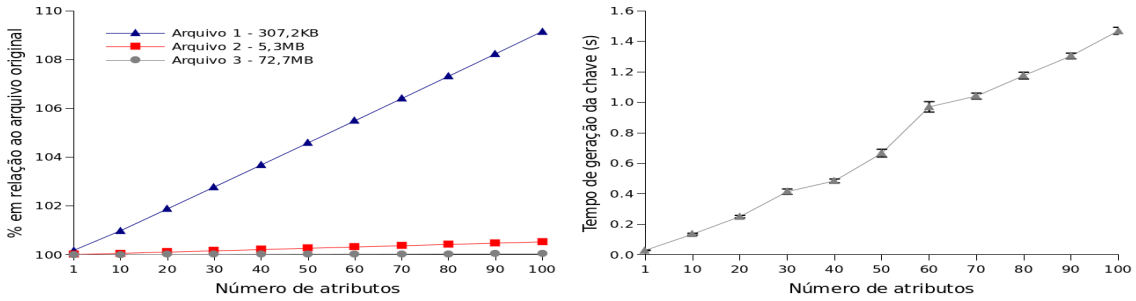
A partir de então, o paciente já passa a informar o novo VH_{CT} aos colaboradores. Estes apenas conseguem obter da rede IPFS o arquivo criptografado com a nova política de acesso, tendo em vista que não conhecem o *hash* antigo. Apenas colaboradores revogados, mas que tenham armazenado localmente o arquivo anteriormente, conseguem acessá-lo (algo difícil de ser impedido). Como medida de proteção contra colaboradores que detém o VH_{CT} antigo e tentarem recuperar da rede IPFS a versão antiga ligada à este *hash*, os pacientes podem informar aos nodos IPFS o *hash* VH_{AS} do arquivo de saúde, solicitando que arquivos criptografados antigos relacionados à VH_{AS} sejam excluídos.

5. Avaliação

O DSHR emprega CBA, *blockchain* e IPFS para garantir segurança. Para mostrar a viabilidade prática do DSHR, uma prova de conceito foi implementada em C++ e uma *blockchain* foi criada com a plataforma de código aberto Ethereum. Testes foram executados em uma máquina virtual com processador Intel Core i7 e 1GB de RAM. As etapas de armazenamento e compartilhamento e a geração de chaves de CBA foram testadas. Três arquivos de saúde com a extensão PDF de tamanhos 307,2KB, 5,3MB e 72,7MB foram usados para avaliar o impacto do tamanho do arquivo no DSHR. Eles são denominados Arquivo 1, Arquivo 2 e Arquivo 3, respectivamente. Três métricas foram avaliadas: o impacto da CBA no tamanho dos arquivos, o tempo de execução e o uso de memória pelo usuário nas etapas. Foi considerada apenas a carga das operações de segurança do DSHR, visto que o tempo para troca de mensagens depende da banda da rede disponível para o usuário. Cada teste foi executado 50 vezes. Os resultados são apresentados a seguir.

Na etapa de armazenamento, a CBA tem impacto no tamanho do arquivo que é armazenado na nuvem. Este tem relação com o tamanho do arquivo original e com o número de atributos contidos na política de acesso, como ilustrado na Figura 2(a). O Arquivo 1 apresentou um aumento de tamanho alto em comparação aos outros arquivos. Seu maior aumento foi de 9% quando 100 atributos foram empregados. Entretanto, isso representa um aumento de apenas 27,6KB. Já os outros dois arquivos apresentaram aumentos menores que 1%. Sendo assim, o custo de armazenamento de arquivos criptografados em detrimento de suas respectivas versões não criptografadas não é significativo.

Na etapa de configuração, chaves secretas de CBA são geradas. O tempo de geração de uma chave cresce de acordo com o seu número de atributos (Figura 2(b)).



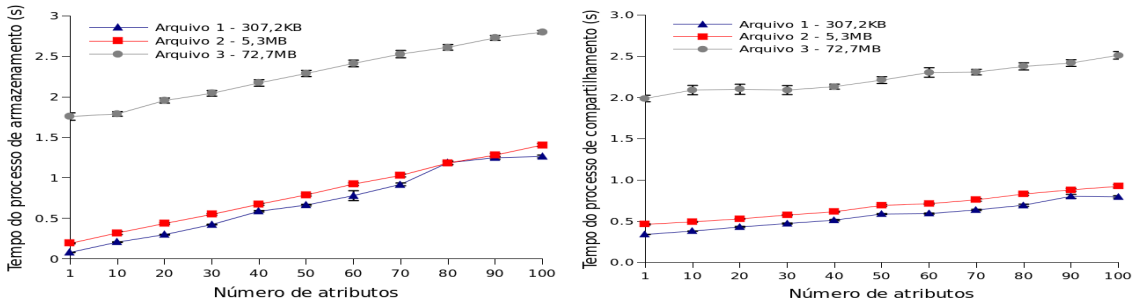
(a) Impacto da CBA no tamanho original dos arquivos de saúde.

(b) Tempo para gerar chaves de CBA.

Figura 2. Resultados experimentais da geração de chaves e criptografia de CBA.

Esse processo mostrou-se eficiente, apresentando o maior tempo médio (1,5s) para 100 atributos, enquanto que para menos que 40 atributos, o tempo médio foi menor que 0,5s.

A etapa de armazenamento é composta pela execução de: criptografia de CBA (uma vez), uma função de *hash* por duas vezes (SHA-256 foi usado) e duas assinaturas digitais (ECDSA foi empregado). A Figura 3(a) apresenta os tempos para completá-la. A diferença entre a execução dessa etapa com os arquivos 1 e 2 foi mínima. Para uma política de acesso com 50 atributos, por exemplo, o tempo médio dessa etapa com o Arquivo 1 foi de 0,65s, enquanto que com o Arquivo 2 foi de 0,8s. Há um aumento significativo do tempo quando o Arquivo 3 é usado, apresentando 2,3s com uma política com 50 atributos e 2,8s com 100 atributos. Esse aumento é consequência dos algoritmos de criptografia de CBA e de *hash*. O tempo de execução das assinaturas é negligenciável.



(a) Tempo para completar a etapa de armazenamento de um arquivo de saúde.

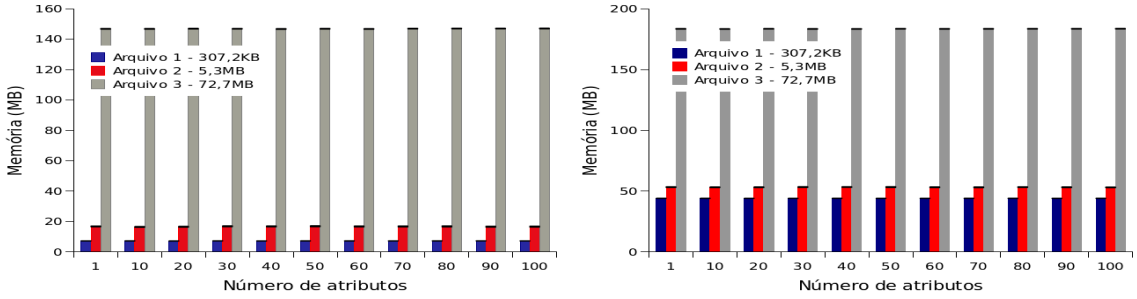
(b) Tempo para completar a etapa de compartilhamento de um arquivo de saúde.

Figura 3. Resultados experimentais do tempo de execução.

A Figura 3(b) ilustra os tempos para completar a etapa de compartilhamento, composta pela execução de: uma busca na *blockchain*, uma descriptografia de CBA, uma função de *hash* por duas vezes e duas verificações de assinatura digital. Comparada à etapa de armazenamento, a de compartilhamento apresenta um maior tempo de execução quando quantidades menores de atributos são usadas. Isso ocorre devido à busca na *blockchain* executada no início da etapa, que leva em média 0,3s (uma *blockchain* com 500.000 transações foi usada). A etapa de compartilhamento, entretanto, não apresenta um crescimento significativo do tempo de execução à medida que o número de atributos aumenta.

Então, com o uso de quantidades maiores de atributos, essa etapa é mais eficiente que a de armazenamento. Para 50 atributos, por exemplo, o tempo médio para o Arquivo 1 foi de 0,59s, enquanto que para os arquivos 2 e 3 o tempo foi de 0,7s e 2,21s, respectivamente.

O uso de memória também mostra a viabilidade do DSHR. Como ilustram as Figuras 4(a) e 4(b), o uso de memória nas duas etapas está diretamente relacionado ao tamanho do arquivo empregado. Na etapa de armazenamento, a quantidade de memória usada é aproximadamente igual ao dobro do tamanho do arquivo mais alguns MB, o que totaliza 7MB para o Arquivo 1, 17MB para o Arquivo 2 e 147MB para o Arquivo 3. A etapa de compartilhamento apresenta as mesmas características que a etapa de armazenamento e mais um uso adicional de memória relacionado à busca de uma transação na *blockchain* (aproximadamente 37MB), o que justifica o fato de os resultados obtidos serem maiores (44MB para o Arquivo 1, 53MB para o Arquivo 2 e 183MB para o Arquivo 3). A variação apresentada em relação ao número de atributos é negligenciável.



(a) Quantidade de memória usada para completar a etapa de armazenamento de um arquivo de saúde.

(b) Quantidade de memória usada para completar a etapa de compartilhamento de um arquivo de saúde.

Figura 4. Resultados experimentais da quantidade de memória usada.

Apenas [Liu et al. 2018] e [Rahulamathavan et al. 2017] realizaram testes experimentais entre os trabalhos relacionados, variando o número de atributos, porém usando arquivos na ordem de KB. A fase de armazenamento de [Rahulamathavan et al. 2017] apresentou um tempo médio de 0,5s quando 10 atributos foram usados (maior número de atributos testado pelos autores), enquanto que o DSHR foi mais eficiente, levando em média 0,2s. Já a fase de compartilhamento de [Rahulamathavan et al. 2017] foi mais eficiente que a do DSHR. O primeiro levou em média 0,23s para ser completado com 10 atributos, enquanto que o segundo levou 0,38s. [Liu et al. 2018], por outro lado, testaram sua proposta variando os atributos de 1 à 100. A sua fase de armazenamento foi sempre mais eficiente, apresentando um tempo médio constante (0,01s), visto que as operações de maior custo para armazenamento são realizadas de antemão (*offline*). Já a sua fase de compartilhamento apresentou um tempo de aproximadamente 100s no pior caso (100 atributos), menos eficiente comparada à mesma fase do DSHR, que levou em média 0,79s.

6. Conclusões e Trabalhos Futuros

Esse trabalho apresentou o DSHR, um protocolo para compartilhamento seguro de arquivos de saúde que usa CBA, IPFS e *blockchain*. Foi mostrado que, diferentemente do estado da arte, o DSHR garante a integridade dos arquivos quando são armazenados fora do controle do paciente, não permitindo modificações e deleções indevidas. Além disso, o DSHR provê revogação de acesso e o não repúdio referente à geração dos arquivos.

Os testes de carga mostraram a viabilidade prática do DSHR mesmo quando um número alto de atributos de CBA e um arquivo grande foram usados. O DSHR é tão eficiente quanto propostas anteriores, visto que os tempos para armazenamento e compartilhamento diferenciam-se em poucos milissegundos para mais ou para menos quando comparados aos tempos das outras propostas. No futuro, pretende-se testar o DSHR com outros tamanhos de *blockchain* e melhorar a eficiência do seu mecanismo de revogação.

Agradecimentos

Essa pesquisa teve suporte parcial da 4ª Chamada Conjunta EU-BR H2020, através do contrato no. 777067 (NECOS - *Novel Enablers for Cloud Slicing*), financiada pela Comissão Europeia e pelo Ministério da Ciência, Tecnologia, Inovações e Comunicações (MCTIC) através da RNP e do CTIC.

Referências

Abbas, A. and Khan, S. U. (2014). A review on the state-of-the-art privacy-preserving approaches in the e-health clouds. *IEEE J. Biomedical and Health Informatics*, 18(4):1431–1441.

Au, M. H., Yuen, T. H., Liu, J. K., Susilo, W., Huang, X., Xiang, Y., and Jiang, Z. L. (2017). A general framework for secure sharing of personal health records in cloud system. *J. Comput. Syst. Sci.*, 90:46–62.

Azaria, A., Ekblaw, A., Vieira, T., and Lippman, A. (2016). Medrec: Using blockchain for medical data access and permission management. In *2nd International Conference on Open and Big Data, OBD 2016, Vienna, Austria, August 22-24, 2016*, pages 25–30.

Benet, J. (2014). IPFS - content addressed, versioned, P2P file system. *CoRR*, 1407.3561.

Bethencourt, J., Sahai, A., and Waters, B. (2007). Ciphertext-policy attribute-based encryption. In *2007 IEEE Symposium on Security and Privacy*, pages 321–334.

Bonneau, J., Miller, A., Clark, J., Narayanan, A., Kroll, J., and Felten., E. (2015). Sok: Research perspectives and challenges for bitcoin and cryptocurrencies. In *SP 2015*.

Dawoud, M. and Altilar, D. T. (2017). Cloud-based e-health systems: Security and privacy challenges and solutions. In *2017 International Conference on Computer Science and Engineering (UBMK)*, pages 861–865.

de Melo Silva, L., Araujo, R., da Silva, F. L., and Cerqueira, E. (2014). A new architecture for secure storage and sharing of health records in the cloud using federated identity attributes. In *16th IEEE Healthcom, Natal-RN, Brazil, October 15-18*, pages 194–199.

Li, Y., Liang, K., Su, C., and Wu, W. (2017). DABEHR: decentralized attribute-based electronic health record system with constant-size storage complexity. In *Green, Pervasive, and Cloud Computing - 12th International Conference*, pages 611–626.

Liu, Y., Zhang, Y., Ling, J., and Liu, Z. (2018). Secure and fine-grained access control on e-healthcare records in mobile cloud computing. *Future Generation Comp. Syst.*, 78:1020–1026.

Puthal, D., Sahoo, B. P. S., Mishra, S., and Swain, S. (2015). Cloud computing features, issues, and challenges: A big picture. In *2015 International Conference on Computational Intelligence and Networks*, pages 116–123.

Rahulamathavan, Y., Phan, R., Misra, S., and Rajarajan, M. (2017). Privacy-preserving blockchain based iot ecosystem using attribute-based encryption. In *IEEE International Conference on Advanced Networks and Telecommunications Systems*.

Serrão, C. and Cardoso, E. (2017). Handling confidentiality and privacy on cloud-based health information systems. *Journal of Information Privacy and Security*, 13(2):51–68.