# Logs Transparentes: transparência e auditabilidade usando estruturas de dados verificáveis

**Leonardo T. Kimura**[1], **Stephanie M. Urashima**[1], **Guilherme Fumagali**[2], **Gustavo C. Bastos**[1], **Yeda R. Venturini**[2], **Marcos A. Simplício Jr.**[1]

[1]Escola Politécnica - Universidade de São Paulo (USP)

[2]Departamento de Computação - Universidade Federal de São Carlos (UFSCar)

lkimura@larc.usp.br, mihourashima@usp.br,
guilhermefumagali@estudante.ufscar.br, gustavocerq5@usp.br
yeda@ufscar.br, mjunior@larc.usp.br

***Abstract.*** *Transparency is one of the fundamental requirements of democracy and is a crucial aspect of processes such as elections and political donations. However, these processes often suffer from suspicion of misconduct and require complex auditing to dispel them. Thus, this paper presents Transparent Logs, a system that promotes greater transparency and efficiency in auditing through an architecture based on verifiable data structures. This paper applies the solution to Brazilian elections and presents a prototype with the main operations required to audit an election.*

***Resumo.*** *A transparência é um dos requisitos fundamentais da democracia e é um aspecto crítico de processos como eleições e financiamento de campanhas eleitorais. Entretanto, frequentemente esses processos sofrem suspeitas de manipulação e exigem uma auditoria complexa para mitigá-las. Desse modo, esse artigo apresenta Logs Transparentes, um sistema que promove maior transparência e eficiência na auditabilidade através de uma arquitetura baseada em estruturas de dados verificáveis. O artigo aplica a solução especificamente para as eleições brasileiras e apresenta um protótipo com as principais operações necessárias para auditar uma eleição.*

## 1. Introdução

A transparência é considerada como um dos requisitos fundamentais da democracia, necessária para que os cidadãos possam avaliar e responsabilizar os governantes pelas suas ações [Hollyer et al. 2011]. Essa importância fica ainda mais evidente nas ”eleições baseadas em evidências”, que devem fornecer evidências convincentes da veracidade do resultado divulgado [Stark and Wagner 2012]; e nos financiamentos de campanhas eleitorais, que devem processar cada doação de forma pública e auditável. Entretanto, frequentemente esses processos carecem de mecanismos transparentes para provar a sua integridade. Os financiamentos de campanhas eleitorais recebem denúncias de caixa 2 e corrupção [Speck 2012], enquanto os derrotados nas eleições repetidamente questionam os resultados divulgados [Pennycook and Rand 2021]. Essas críticas se estendem às eleições brasileiras, nas quais a sua legitimidade é abalada pelas acusações de falta de transparência e de pouca auditabilidade. [van de Graaf 2017] [Filho et al. 2015].

Uma das causas dessas críticas é a complexidade na auditoria desses processos, que exige a coleta, a análise e a agregação de uma grande quantidade de dados individuais. Por exemplo, para conferir o montante total das doações realizadas a um candidato é preciso analisar cada recibo individualmente e então somá-los, de modo a comparar com o valor declarado pelo candidato. Semelhantemente, para auditar o resultado de uma eleição, é preciso coletar e somar todos os votos agregados de cada sessão eleitoral. Isso acaba restringindo as auditorias para algumas organizações fiscalizadoras, tornando-as pouco viáveis para usuários com recursos limitados [Filho et al. 2015].

Para estimular a fiscalização pelo público, algumas entidades responsáveis por esses processos divulgam na Internet informações que consideram relevantes para a auditoria. Nas eleições brasileiras, por exemplo, os resultados dos votos de cada sessão eleitoral são divulgados através do BU na WEB [Tribunal Superior Eleitoral 2021]. No entanto, frequentemente esses repositórios não apresentam garantias de integridade das informações divulgadas e estão sujeitas à alteração de resultados ou à destruição de evidências. No caso do BU na WEB, a validação da integridade de suas informações exige mecanismos adicionais, como a comparação com uma versão impressa confiável. Além disso, essa validação deve ser realizada recorrentemente para detectar manipulações em momentos posteriores,

Desse modo, esse trabalho apresenta uma solução que aumenta a transparência de processos críticos da democracia ao permitir uma auditoria eficiente das informações divulgadas. Particularmente, o sistema usa logs verificáveis para garantir eficientemente a integridade dos dados inseridos e usa mapas verificáveis para permitir a auditoria do resultado final a partir de amostras das informações. O sistema apresenta uma solução específica para as eleições brasileiras; entretanto, poderia ser adaptada para qualquer contexto que envolva a agregação de elementos individuais sensíveis, como financiamento de campanhas eleitorais. Por fim, esse artigo apresenta um protótipo com as principais operações necessárias para auditar uma eleição, como a verificação da integridade das informações registradas e a apuração paralela do resultado final da eleição.

## 2. Trabalhos relacionados

Semelhantemente ao Logs Transparentes, o Certificate Transparency [Laurie 2014] é um projeto que promove a transparência de certificados TLS usando estruturas verificáveis. Para isso, ele salva todos os certificados emitidos pelas CAs em um log verificável e permite que as entidades interessadas busquem e identifiquem certificados emitidos de forma fraudulenta. Entretanto, ele não possui algumas funcionalidades específicas de processos com contagem para auxiliar a sua auditoria. Por exemplo, Certificate Transparency não facilita a verificação do resultado das eleições através de algumas amostras de dados.

Já blockchain é proposto como uma estrutura alternativa para aumentar a transparência dos dados e garantir a integridade delas devido a sua natureza distribuída e capacidade de detecção de manipulações [Silva 2018]. Entretanto, blockchains tipicamente exigem mecanismos de consenso que acabam prejudicando o seu desempenho em relação aos sistemas centralizados. Além disso, blockchains usam extensivamente criptografias assimétricas, o que torna a solução mais custosa computacionalmente do que as estruturas de dados verificáveis, construídas majoritariamente pelo cálculo de hashes. O custo de sua auditoria também costuma ser maior, já que a sua estrutura em cadeia exige uma

verificação linear dos dados (O(n)), enquanto a verificação em árvore permite uma auditoria que cresce de forma logarítmica (O(log(n)). Por fim, o uso de blockchain em processos críticos como eleições é duramente criticado por introduzir oportunidades para problemas de segurança, por exemplo, uma maior complexidade no desenvolvimento e gerenciamento de seu software [Park et al. 2021].

## 3. Fundamentação teórica

Esta sessão apresenta os conceitos de estruturas de dados verificáveis e, em seguida, alguns aspectos do sistema eleitoral brasileiro.

### 3.1. Estruturas de Dados Verificáveis

As estruturas de dados verificáveis são construídas através de árvores de Merkle binárias [Merkle 2019], que salvam as informações nas folhas e calculam os nós intermediários através do hash de seus nós filhos. Dessa forma, elas garantem a integridade dos dados contidos na árvore através somente da monitoração da raiz. Além disso, essas estruturas permitem provar eficientemente a existência de um elemento na árvore através da prova de inclusão. Essa prova consiste em uma lista de nós necessária para recalcular a raiz da árvore a partir daquele elemento e seu custo computacional é proporcional ao logaritmo do número de elementos na árvore. Normalmente essas provas são assinadas, de modo que elas não podem ser forjadas por entidades externas. As estruturas de dados verificáveis e suas propriedades são descritas abaixo.

**Logs Verificáveis:** Um log verificável consiste em uma árvore de Merkle binária com a propriedade append-only [Eijdenberg et al. 2015], o que significa que as informações são somente adicionadas mas nunca removidas ou alteradas. Para provar essa propriedade, os logs verificáveis podem gerar uma prova de consistência entre duas versões da mesma árvore sempre que solicitado. Alternativamente, os logs verificáveis podem provar essa sua consistência ao publicar as alterações da árvore a cada nova adição de elementos. Nesse caso, alguns monitores interessados podem manter localmente uma cópia da árvore original para comparação.

**Mapas verificáveis apoiados em log:** Um mapa verificável é uma árvore de Merkle na qual os dados são armazenados em uma posição calculada a partir de seu próprio conteúdo [Eijdenberg et al. 2015]. Desse modo, os mapas verificáveis permitem provar tanto a existência quanto a não existência de um dado da árvore através da prova de inclusão na posição correspondente ao conteúdo do dado. Entretanto, ele não possui a propriedade append-only. Desse modo, para permitir a sua auditoria, ele deve ser apoiado em um log verificável que armazena todas as mudanças realizadas. A integridade de ambas as árvores são garantidas através de um novo log verificável que armazena a raiz do mapa, a raiz do log, e o tamanho delas.

### 3.2. O sistema eleitoral brasileiro

O Brasil realiza as suas eleições dividindo o pais em sessões eleitorais, cada uma com uma urna eletrônica para receber os votos dos eleitores. Ao final da votação, a urna contabiliza digitalmente os votos recebidos e gera o Boletim de Urna (BU), que contém o resultado daquela sessão. Esse BU é impresso, assinado pelos fiscais presentes na sessão eleitoral, e fixado no local da votação para conferência pública. A versão digital do BU é então enviada para o sistema central que coordena a totalização [TSE 2019].

Então, o sistema eleitoral central verifica a autenticidade e a integridade de cada BU recebido e trata quaisquer irregularidades, tais como duplicatas ou má formatações. Ele também trata os votos que devem ser considerados nulos, como os votos que foram atribuídos a um candidato cassado. Por fim, o sistema contabiliza os votos recebidos. Após o término da apuração, o resultado final é divulgado, assim como uma versão digital dos BUs recebidos [TSE 2019].

O processo eleitoral também gera outras informações, como a tabela de correspondência, que correlaciona as urnas eletrônicas com as sessões que as usaram, o hash do código-fonte usado nas urnas, e os logs das urnas.

## 4. Visão Geral do Logs Transparentes

Conforme mostra a Fig. 1, o sistema é composto por diferentes árvores que registram as informações eleitorais. A árvore pré-eleição contém as informações geradas e publicadas antes do dia das eleições, como o código de correspondência e o código-fonte, enquanto a árvore pós-eleição contém as informações obtidas depois do dia das eleições, como os logs da urna. A árvore de BUs registra os votos de cada sessão eleitoral, enquanto o mapa de BUs contém as informações adicionais que facilitam a recontabilização do resultado da eleição. Por fim, a árvore de raízes registra periodicamente a raiz de cada árvore e gera uma única raiz principal que deve ser fiscalizada pelos monitores.

O sistema prevê dois tipos de participantes que auditam o funcionamento da eleição: (1) os eleitores, que verificam individualmente a integridade das informações divulgadas; e (2) os monitores, que verificam o funcionamento do sistema como um todo. Os monitores são tipicamente compostos por organizações com maior interesse nas eleições, como partidos políticos ou o ministério público; entretanto, qualquer pessoa interessada pode participar. As tarefas dos monitores podem ser subdivididas em três operações principais: (1) armazenar localmente a raiz principal do Logs Transparentes, de modo a não permitir alterações arbitrárias nas árvores do sistema; (2) verificar a consistência das árvores sempre que novas informações forem inseridas; e (3) recontabilizar o resultado das eleições a partir das informações do sistema, comparando-o com o resultado oficial divulgado.

### 4.1. Verificação de BU e outras informações

Os eleitores podem verificar a integridade dos BUs e de outras informações registradas no sistema através da prova de inclusão. Para isso, os eleitores escolhem o dado a ser verificado e solicitam ao sistema a prova de inclusão correspondente através de uma API pública. Se a prova estiver correta, os eleitores podem confiar não só que o dado está íntegro naquele momento, mas também que ele não será manipulado posteriormente pelo sistema. Isso é útil para verificar os BUs digitais, por exemplo, já que eles exigem recorrentemente uma comparação com uma versão impressa confiável. Note que os eleitores devem realizar a verificação em um ambiente de sua confiança, visto que uma verificação fornecida pela entidade eleitoral tem pouco valor para auditar a ação dela mesma. Para isso, essa verificação pode ser feita através de um aplicativo próprio ou através de um código-aberto já inspecionado. Além disso, os eleitores devem realizar as verificações usando uma raiz confiável; assim, eles devem solicitá-la à pelo menos um monitor de sua escolha.

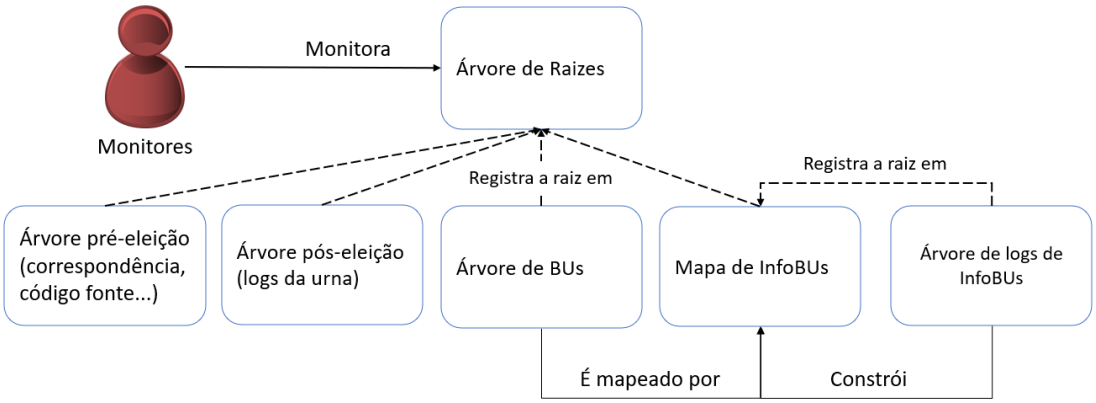


**Figura 1. As árvores do sistema e o relacionamento entre elas. Cada árvore é responsável por armazenar informações em diferentes etapas da eleição. O mapa de infoBUs contém informações relacionadas a cada BU e é apoiado pela árvore de logs de infoBUs. A raiz de cada árvore é registrada na árvore de raízes, que é fiscalizada pelos monitores.**

### 4.2. Monitoração da consistência das árvores

Um log verificável só garante a integridade dos dados enquanto a sua raiz for confiável. Desse modo, os monitores devem sempre armazenar a última raiz assinada do Logs Transparentes para prevenir qualquer alteração indevida. Além disso, os monitores devem fiscalizar a consistência da árvore e garantir que nenhuma informação anterior foi removida ou alterada. Eles fazem essa fiscalização sempre que uma nova raiz for publicada, por exemplo, durante o recebimento dos BUs no dia das eleições. Nessas ocasiões, os monitores devem verificar a prova de correspondência entre a nova raiz e a última raiz publicada, que será fornecida pelo sistema.

Alguns monitores podem desejar validar a consistência da árvore depois da publicação de várias raízes do sistema. Nesse caso, eles não devem solicitar as raízes antigas para o sistema, já que ele pode lhes fornecer uma versão adulterada. Ao invés disso, os monitores devem obter as raízes em uma fonte de sua confiança. Uma opção é solicitá-las a um ou mais monitores; entretanto, isso exige que os outros monitores armazenem todas as versões já publicadas da raiz. Outra opção é o sistema registrar as raízes em um repositório público fora do controle da entidade eleitoral, como em uma blockchain pública de baixo custo [Mendonça and Matias 2021].

### 4.3. Recontabilização dos votos

A integridade dos dados inseridos no Logs Transparentes, por si só, não garante a veracidade do resultado da eleição; é preciso também conferir se os BUs registrados foram considerados no resultado final. Assim, os monitores podem realizar essa conferência através da recontabilização total dos votos, que envolve baixar toda a árvore de BUs, verificar cada prova de inclusão, e somar todos os votos registrados.

Como essa recontabilização total dos votos possui um custo computacional elevado, os monitores podem optar por realizar uma verificação parcial do resultado. Nesse cenário, os monitores podem para auditar diferentes partes da eleição e assim combinar os seus esforços para verificar o resultado total divulgado. Para isso, o sistema utiliza o mapa de infoBUs, que contém em cada nó a soma dos votos dos nós abaixo deles. Nesse

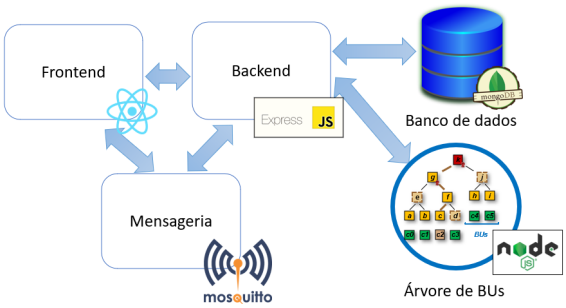


**Figura 2. Principais módulos do protótipo, com as tecnologias utilizadas**

mapa, a raiz de cada subárvore contém o resultado parcial dos BUs abaixo dele, e a raiz principal do mapa contém o resultado final da eleição. Assim, a prova de inclusão desse mapa envolve não só o cálculo dos hashes do caminho até a raiz, mas também o hash dos resultados parciais incluídos em cada nó intermediário. Consequentemente, caso algum monitor tenha verificado o resultado de algum nó intermediário, outros monitores não precisam fazê-lo novamente.

Além da soma dos votos dos BUs, o cálculo do resultado das eleições envolve a aplicação de regras específicas (e.g., anulação de votos de candidatos cassados). Assim, essas regras aplicadas também são registradas no mapa de infoBUs, na posição correspondente ao BU sendo processado. Então, os votos desse infoBUs são recalculados e os resultados armazenados em seus nós superiores são atualizados. Desse modo, os monitores interessados podem conferir o resultado de uma sessão eleitoral ao verificar os votos de um BU e aplicar as regras registradas no infoBU correspondente. Como essas regras podem ser aplicadas mesmo depois da construção do mapa de infoBUs, todas as novas alterações do mapa são registradas na árvore de logs de infoBUs. Desse modo, o mapa de infoBUs é apoiado pela árvore de histórico de infoBUs, de maneira semelhante ao mapa verificável apoiado por log.

## 5. Protótipo

O protótipo implementa Logs Transparentes usando a árvore de BUs e apresenta as principais funcionalidades para verificar uma eleição. Através dele, o eleitor pode verificar a integridade dos BUs inseridos e os monitores podem recontabilizar o resultado final da eleição. Além disso, o protótipo permite aos monitores fiscalizar a consistência da árvore durante a sua construção comparando-a com uma cópia da árvore construída localmente. Por fim, o protótipo está disponível em https://github.com/larc-logs-transparentes/logs-transparentes, que inclui uma documentação sobre a instalação, requisitos do sistema e as operações possíveis.

A arquitetura do protótipo pode ser vista na Fig. 2. Conforme mostra a figura, o Frontend é uma interface WEB construída em React, enquanto o Backend é uma API pública construída em Express.js. O BU é armazenado em um banco de dados MongoDB, enquanto a árvore de BU armazena o hash dos BUs em um log verificável usando uma biblioteca em NodeJS [Mota 2022]. Por fim, o serviço de mensageria é um servidor MQTT usando Mosquitto, no qual o Frontend se subscreve para receber as atualizações publicadas pelo Backend durante a construção da árvore.

O protótipo toma o cuidado de executar todas as verificações através de um código


**Figura 3. Consulta e verificação de um BU. As provas de auditoria são fornecidos no card a direita, e o resultado da verificação é evidenciado pela cor do cadeado, que pode ser verde (correto) ou vermelha (incorreto)**

Javascript executado no Frontend, na máquina do usuário. Assim, o próprio usuário pode verificar a veracidade do resultado. Se desejar, o usuário pode usar a prova de inclusão disponibilizada no Frontend e realizar as suas próprias validações utilizando um script de sua autoria. Fig. 3 ilustra a tela de consulta e verificação de um BU com sua respectiva prova de inclusão.

## 6. Demonstração

A demonstração será realizada através de um computador conectado à rede interna do LARC-USP, utilizando uma instância do protótipo no servidor do laboratório. As funcionalidades apresentadas incluem as principais operações de verificação da eleição, como a consulta e a validação de um BU, o monitoramento da árvore de BUs durante a sua construção, e a recontabilização do resultado. Uma tentativa de manipulação será simulada através de uma modificação direta no banco de dados e será detectada de maneira perceptível ao usuário.

## 7. Conclusão

Esse artigo descreveu uma solução baseada em árvores de Merkle para aumentar a transparência e a auditabilidade de processos críticos à democracia. Foi apresentada uma solução específica para as eleições brasileiras, porém ela pode ser adaptada para qualquer processo que envolva a verificação e a agregação de resultados individuais sensíveis. Nessa solução, os eleitores podem verificar a integridade dos dados registrados de maneira eficiente, enquanto os monitores podem conferir o resultado final das eleições. Foi implementado um protótipo com as principais operações do sistema, como a verificação de BUs, monitoração da raiz da árvore, e a recontabilização do resultado da eleição. Em trabalhos futuros, pretende-se explorar outros mecanismos para melhorar a auditoria, como a fiscalização de apenas uma parte da árvore de maior interesse aos monitores e o uso de smart contracts para automatizar a monitoração de forma publicamente verificável.

***Agradecimentos***: Esse trabalho foi em parte financiado pelo CNPq (bolsa de produtividade 304643/2020-3) e pela Ripple através da *University Blockchain Research Initiative*. Também agradecemos Paulo Matias (UFSCar), Roberto Samarone S. Araújo (UFPA), Diego F. Aranha (Aarhus U.) pelos comentários feitos ao trabalho.

## Referências


Eijdenberg, A., Laurie, B., and Cutter, A. (2015). Verifiable data structures. *Google Research, Tech. Rep*.

Filho, A. B., Carvalho, M. A., Teixeira, M. C., Simplicio Jr, M. A., and Fernandes, C. T. (2015). Auditoria especial no sistema eleitoral 2014. In *XV Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais*. SBC.

Hollyer, J. R., Rosendorff, B. P., and Vreeland, J. R. (2011). Democracy and transparency. *The Journal of Politics*, 73(4):1191–1205.

Laurie, B. (2014). Certificate transparency. *Communications of the ACM*, 57(10):40–46.

Mendonça, B. d. A. and Matias, P. (2021). Auditchain: a mechanism for ensuring logs integrity based on proof of existence in a public blockchain. In *11th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, pages 1–5.

Merkle, R. C. (2019). Protocols for public key cryptosystems. In *Secure communications and asymmetric cryptosystems*, pages 73–104. Routledge.

Mota, M. (2022). Merkletreejs GitHub repository. `https://github.com/miguelmota/merkletreejs`. Acessed on 18-07-2022.

Park, S., Specter, M., Narula, N., and Rivest, R. L. (2021). Going from bad to worse: from internet voting to blockchain voting. *Journal of Cybersecurity*, 7(1).

Pennycook, G. and Rand, D. (2021). Examining false beliefs about voter fraud in the wake of the 2020 presidential election. *The Harvard Kennedy School Misinformation Review*.

Silva, M. P. (2018). A segurança da democracia e a blockchain (securing democracy through blockchain). *Revista Projeção, Direito e Sociedade*, 9(1).

Speck, B. W. (2012). O financiamento político e a corrupção no brasil. *Temas de corrupção política. São Paulo: Balão Editorial*, pages 49–97.

Stark, P. B. and Wagner, D. (2012). Evidence-based elections. *IEEE Security & Privacy*, 10(5):33–41.

Tribunal Superior Eleitoral (2021). Resultados de eleições e boletins de urna estão disponíveis para consulta no Portal do TSE. `https://bit.ly/3Pyw0cg`. Acessed on 25-07-2022.

TSE (2019). Resolução nº 23.611, 19/12/2019. Dispõe sobre os atos gerais do processo eleitoral para as Eleições 2020. `https://www.tse.jus.br/legislacao/compilada/res/2019/resolucao-no-23-611-de-19-de-dezembro-de-2019-1`. Acessed on 14-07-2022.

van de Graaf, J. (2017). O mito da urna: Desvendando a (in) segurança da urna eletrônica. https://www.urantiagaia.org/social/eleicao/mito-da-urna.pdf. Acessed on 13-07-2022.