**TECNOLOGIA ETHERVOLTZ PARA ELEIÇÕES AUDITÁVEIS.**

**1Alencar, M. F., 2Simoes, M.C., 3Silva, D.R.**

1,2,3Faculdade ETEP/Faculdade de Tecnologia de São José dos Campos, Avenida Barão do Rio Branco, 882, Jardim Esplanada – 12242-800 - São José dos Campos-SP, Brasil, 1[mtsalenc@gmail.com](mailto:mtsalenc@gmail.com) [2marize.simoes@etep.edu.br](mailto:2marize.simoes@etep.edu.br) [3diogenes.silva@etep.edu.br](mailto:3diogenes.silva@etep.edu.br)

**Resumo -** Este artigo apresenta uma prova de conceito de sistema eleitoral independente de software que remove do administrador a responsabilidade de garantir a disponibilidade, integridade e confiabilidade dos registros digitais dos votos. O código é escrito na linguagem *Solidity* e posteriormente compilado para *bytecode* que pode ser interpretado pela máquina virtual da rede *Ethereum*. O resultado é a criação de uma criptomoeda que existe para representar cada voto, onde cada registro digital de voto fica gravado num blockchain resistente a censura, ataques de negação de serviço e que está sempre disponível para auditorias.

**Palavras-chave:** tecnologia, blockchain, ethereum, criptomoeda, ethervoltz.

**Área do Conhecimento:** Engenharia da Computação

**Introdução**

Nos urnas eleitoras de primeira, segunda e terceira geração, todas as provas geradas pelos votos ficam sob controle do administrador e precisam ser levadas dos locais de votação de volta para as centrais de apuração dos votos. Além dos custos envolvidos para garantir que estes equipamentos não sejam alterados ou destruídos durante o transporte, o administrador também precisa guardar estes registros após a apuração de votos para auditorias.

Após o período eleitoral, caso um cidadão queira auditar os resultados das eleições, ele precisa interagir com o administrador do processo eleitoral para ter acesso os registros digitais, aos equipamentos e aos registros independentes de *software* (caso o administrador tenha optado por um sistema eleitoral independente de *software*). Cabe ao administrador decidir se ele tem ou não permissão para realizar a auditoria e quais são condições para a realização da mesma.

Os problemas deste tipo de centralização de poderes ficam evidentes nas eleições do Brasil. Embora não seja objetivo deste artigo discutir esses probemas, três casos são brevemente listados:

1. O Caso Marília, SP - 2004: Em auditoria, os Arquivos de Espelhos de Boletins de Urna da 400º Zona Eleitoral indicavam que muitas seções eleitorais tiveram seus resultados recebidos para apuração *antes* do início da votação. (SÉRVULO et al.,2010)
2. O Caso Alagoas - 2006: Diversas irregularidades nos arquivos gerados pelas urnas foram detectadas por auditores externos [4]. Frente as evidências, o administrador *negou* acesso aos arquivos solicitados pelos auditores e transferiu ao requerente uma cobrança antecipada no valor de R$ 2 milhões para que fosse desenvolvida uma perícia das urnas. Diante do não pagamento do valor proibitivo, o requerente foi multado e condenado por litigância de má-fé. (SÉRVULO et al.,2010)
3. O Caso Itajaí, SC - 2008: *Nenhuma* urna preparada para a votação passou pelo teste obrigatório prescrito pelo Art. 32 da Res. TSE 22.712/08. Um caso foi o da 97ª Zona Eleitoral onde a urna da seção 236 que foi sorteada para o teste obrigatório foi substituída por outra na hora do teste, preparada exclusivamente para este fim. A urna que foi utilizada para o teste foi posteriormente colocada à parte e recarregada, procedimento que destruiu eventuais provas nela gravadas. (SÉRVULO et al.,2010)

Em países como Alemanha, Argentina e Equador, que utilizam urnas mais modernas como as de segunda e terceira geração (BRUNAZO, 2014), o Princípio da Independência de Software em Sistemas Eleitoriais (RIVEST, 2016) é preservado, entretanto, todas as provas geradas no momento do voto são controladas pelo administrador. Auditorias ocorrem apenas com a autorização e sob condições impostas pelo mesmo. O caminho do voto desde a urna até a apuração funciona como uma caixa preta.

O objetivo do projeto é desenvolver sistema eleitoral que respeite o Princípio da Independência de Software em Sistemas Eleitorais e que decentralize o destino das provas geradas em cada voto, de forma que os registros físicos ficam sob controle do administrador e os registros digitais ficam sob controle de um programa autônomo que pode ser auditado de forma sem a necessidade de interagir com o administrador.

**Metodologia**

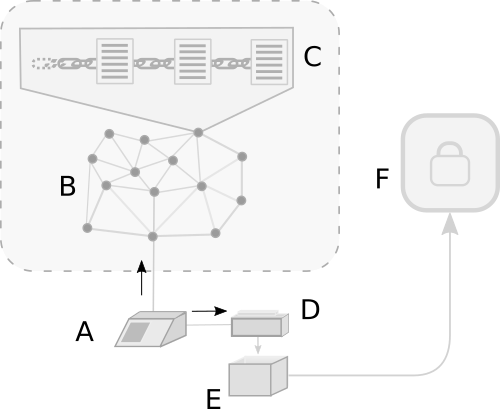
Para o desenvolvimento do projeto, foram estabelecidos os seguintes requisistos:

1. Velocidade de Apuração – O sistema precisa entregar velocidade de apuração igual ou superior aos sistemas eleitorais atuais.
2. Disponibilidade – Os registros digitais de votos precisam estar armazenados em um sistema que seja resistente a ataques de negação de serviço e devem estar sempre disponíveis para auditorias.
3. Integridade – Os registros digitais dos votos precisam ser imunes a alteração não autorizada e logs de alterações executadas devem imutáveis e permantentes.
4. Decentralizado e Autônomo – Parte do software e infraestrutura utilizados não devem estar sob controle de uma autoridade central.
5. Independência de Software – Erros ou alterações não detectados no software do sistema não devem poder causar modificações ou erros indetectáveis no resultado final.

Para garantir os requisitos 1,3 e 4, foi adotada a estratégia de transformar cada voto em uma criptomoeda, chamada *VoltToken*, cujas regras de emissão e transferência são definidos no *bytecode* gerado da compilação de um contrato inteligente imutável (ETHEREUM, 2016). Desta forma, cada voto é uma transferência de uma carteira associada a uma determinada urna, para uma carteira associada a um determinado candidato.

Para garantir o requisito 2, a plataforma e infraestrutura escolhida para hospedar o contrato inteligente é a máquina virtual Ethereum e a *rede peer-to-peer* formada pelos milhares de nós que possuem cópias do *blockchain* utilizado como banco de dados dos votos (WOOD, 2015). Como o código do contrato é aberto e as transações são asseguradas por criptografia de chaves assimétricas, um auditor pode verificar as regras de negócio do sistema e verificar a origem de todas as transações realizadas. De maneira similar a outras criptomoedas como o Bitcoin, a infraestutura não possui autoridade central com poder de emitir ou realizar transferências de VoltTokens de forma indetectável ou fora das regras definidas no contrato inteligente (ETHEREUM, 2016).

Figura 1 – Arquitetura do sistema



Fonte: O autor.

Na Figura 1, as letras representam respectivamente:

* Um computador a ser utilizado como urna sob controle do adminstrador, e que possui uma chave privada capaz de assinar transferências de *VoltTokens*.
* A rede *peer-to-peer* formada pelos milhares de nós e mineradores da rede *Ethereum*
* A cópia do *blockchain* *Ethereum* em um nó na rede.
* Uma impressora para realizar a impressão de Votos Conferídos Pelo Eleitor (*VICE*).
* Uma urna para a coleta dos *VICEs*
* Um cofre sob controle do adminstrador para armazenamento dos *VICE*.

Como as chaves privadas que assinam cada voto estão sob controle do administrador, elas estão sujeitas a ataques internos e externos que podem causar o vazamento das mesmas. De maneira similar, o *frontend* do utilizado para interagir com a aplicação hospedada na rede *Ethereum* pode conter erros ou estar infectado com código malicioso capaz de fraudar votos.

Para garantir que não seja possível este tipo de vulnerabilidade ser explorada para a emissão de votos fraudulentos e válidos, é necessário que o sistema atenda ao Princípio da Independencia de Software em Sistemas Eleitorais, que é o item 5 dos requisitos. A estratégia utilizada é a emissão de uma versão modificada das provas auditáveis pelo eleitor utilizadas em urnas de 2ª geração.

Figura 2 – VICE do Ethervoltz



Fonte: O autor.

Cada transação confirmada e incluída em um bloco na rede *Ethereum*, produz um *hash* que a identifica unicamente (ETHEREUM,2016). Uma busca em um explorador de blocos com este *hash*, permite saber a chave pública que a assinou a transação e a chave pública da carteira destino. Consequentemente, este *hash* identifica a urna de onde um determinado voto saiu e qual candidato recebeu o voto. O sistema Ethervoltz explora este fato para criar um laço entre as provas impressas e aos registros digitais, ao incluir este hash nos votos impressos. A Figura 2 mostra informações que devem estar incluídas no *VICE* utilizado pelo sistema *Ethervoltz*.

**Resultados**

Um sistema eleitoral independente de software e que grava os registros digitais de votos numa base de dados autônoma e resistente a censura. Foi escrito um contrato inteligente escrito na linguagem *Solidity* utilizando o *framework* *Truffle* e testes *javascript* para cada função criada. Os testes são desenvolvidos utilizando as suites de teste e asserção *mocha.js* e *chai.js.*

O contrato inteligente define as seguintes regras, que são imutáveis após hospedagem na rede:

* A transferência de *VoltTokens* só pode ocorrer durante o período eleitoral, portanto, fraudes também só podem ocorrer neste horário.
* Apenas endereços de carteiras que representam candidatos podem receber *VoltTokens* no período eleitoral.
* O número total de moedas em circulação é definido no momento da criação do sistema.
* Nenhuma nova unidade da moeda pode ser emitida após a criação do sistema.
* Cada urna recebe precisamente o número de *VoltTokens* correspondente ao número de eleitores que devem votar naquela urna.

As regras anteriores são auditáveis por qualquer um através do código fonte do contrato disponibilizado pelo administrador. Todos os *VoltTokens* são rastreáveis desde o momento de sua emissão, portanto, roubos de votos ficam registrados permanentemente no blockchain assim como a origem e destino dos mesmos.

**Discussão**

Em sistemas eleitorais de primeira, segunda e terceira geração, todo o processo de auditoria precisa necessariamente envolver o administrador, já que este tem custódia de todas as provas do processo eleitoral. No sistema proposto, apesar do administrador ser o responsável pela produção do código fonte, o software e banco de dados que gerenciam os registros digitais de voto, não estão sob o controle do mesmo e consequentemente, parte da auditoria pode ocorrer sem a necessidade do envolvimento do administrador. De fato, utilizando apenas um navegador de blocos, qualquer pessoa pode analisar as transferências de *VoltTokens*, em busca de endereços e carteiras que não foram anunciados em cerimônia oficial pelo administrador.

Já uma auditoria mais aprofundada pode ser conduzida da seguinte forma:

1. O auditor solicita à maquina virtual uma lista com todas as transações realizadas pela urna em questão, passando a chave pública da urna.
2. O auditor solicita ao administrador da eleição, a caixa contendo os votos impressos conferíveis pelo eleitor, da mesma urna.
3. O auditor compara as duas provas, em busca de provas inconsistentes.

Alguns exemplos de inconsistências nas provas, e que caracterizam fraudes são listados:

* O número de *VICE's* na caixa entregue pelo auditor é diferente do número de registros de voto digital retornados pela máquina virtual Ethereum
* Algum registro digital de voto retornado pela máquina virtual Ethereum não possui seu VICE associado na caixa que o administrador entregou ao auditor.
* Algum VICE não possui ter um hash válido.

Um *hash* impresso no *VICE* é considerado válido se:

* Ele existir no blockchain Ethereum
* O endereço do remetente for igual ao da urna que está sendo auditada
* O endereço da carteira do candidato que recebeu o voto for o mesmo que o impresso no VICE
* O endereço do contrato da criptomoeda for o mesmo publicado na cerimônia oficial.

**Conclusão**

EtherVoltz é uma proposta que visa decentralizar o processo eleitoral e de auditorias ao transferir a responsabilidade de gerenciar dos registros digitais de votos a um programa que executa em uma máquina virtual que não possui autoridade central, é imutável e resistente a censura. Ao transformar o voto do eleitor em uma criptomoeda, o sistema imediatamente ganha todas as propriedades de segurança do protocolo de consenso e de disponibilidade da rede *peer-to-peer* utilizado na plataforma.

Ao transformar o voto do eleitor em uma criptomoeda, o sistema imediatamente ganha todas as propriedades de segurança do protocolo de consenso. Votos são finitos e cada voto é rastreável do candidato até a urna de onde partiu, garantindo o anonimato ao eleitor.

Embora ainda exista a necessidade da emissão e controle das provas impressas para garantir o Princípio da Independência de Software ao sistema, a separação do destino das duas provas produzidas no momento do voto dá aos eleitores poder de auditoria com limitada necessidade do envolvimento de intermediários.

**Referências**

BRUNAZO, A. F. Modelos e Gerações dos Equipamentos de Votação Eletrônica. Disponível em: <<http://www.brunazo.eng.br/voto-e/textos/modelosUE.htm>>.Acesso em 27 jul.2017.

CHOHAN, U. W. Criptocurrencies: A Brief Thematic Review. Disponível em: <<https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3024330>>.Acesso em 13 ago.2017.

ETHEREUM, Ethereum WhitePaper. Disponível em: <<https://github.com/ethereum/wiki/wiki/White-Paper>>. Acesso em 09/07/2017.

RIVEST, R. L. et al. On the notion of “software independence“ in voting systems. Disponível em: <<https://people.csail.mit.edu/rivest/pubs/RW06.pdf>>. Acesso em 22 jul.2017.

SÉRVULO, S S. et al. 1º Relatório do Comitê Multidisciplinar Independente, 2010 Disponível em: <<http://www.votoseguro.org/textos/CMind-1-Brasil-2010.pdf>>. Acesso em 15 jul.2017.

WOOD, G. Ethereum: A Secure Decentralised Generalised Transaction Ledger. Disponível em: <<http://www.cryptopapers.net/papers/ethereum-yellowpaper.pdf>>. Acesso em 13 jun.2017.