ANÁLISE E ALTERAÇÃO DO PROTOCOLO CRIPTOGRÁFICO DE VOTAÇÃO DIGITAL LICHTLER

Rodrigo Schneider
Centro Universitário La Salle – UniLaSalle
Av. Victor Barreto, 2288 - Canoas/RS - Brasil
rodigo@yahoo.com

e

Vinicius Gadis Ribeiro
Centro Universitário Ritter dos Reis – UniRitter
Ru Orfanotrófio, 555 – Porto Alegre/RS - Brasil
vinicius@ritterdosreis.br

RESUMO

O protocolo criptográfico Lichtler, recentemente proposto, pretende atender os requisitos mínimos de segurança exigidos em sistemas de votação digital. O objetivo do presente trabalho foi estudar a análise desse protocolo, assim como sua implementação – no sentido de torná-lo realizável -, buscando tornar prática a sua a viabilidade, sem no entanto ferir as características de segurança propostas pelo autor. Para isso, o estudo e análise conduzidos foram baseados nos requisitos de segurança comuns a qualquer protocolo de votação digital, concentrando-se na implementação em um modelo tecnicamente viável. A análise do estudo realizado apresenta algumas falhas encontradas e o presente trabalho propõe algumas melhorias. A partir do protocolo modificado, implementou-se um protótipo que buscou validar o correto funcionamento e a viabilidade de seu uso prático.

Palavras-chave: Segurança Computacional, Protocolos Criptográficos, Votação Digital.

ABSTRACT

Lichtler cryptographic protocol intends to fulfill the minimal security requirements demanded on electronic voting systems. The aim of the present research was to study this protocol focusing a practical parameter, implementing and validating it as an avaliable protocol, without harming the security characteristics proposed by the author, thought. Furthermore, a study was conducted as well as an analysis of the protocol, basing it on security and implement requirements with the goal of reaching a tecnically viable model. The present article has demonstrated some errors comes up with some improvements. With this modified protocol, a prototitpe was implemented that demonstrated the working and viability of its practical usage.

Keywords: Computer Security, Cryptographic Protocols, Electronic Voting.

1 INTRODUÇÃO

A votação digital agrega as vantagens oferecidas pelas redes de computadores ao processo de votação. Apesar de o termo mais difundido na comunidade científica internacional ser votação eletrônica, no Brasil o termo votação digital é utilizado para caracterizar a diferença com relação ao processo de votação eletrônica - atualmente adotado pelo Tribunal Superior Eleitoral, cuja principal vantagem é a velocidade da

apuração dos resultados.

Considerando o valor das informações trafegadas em meio ao processo de votação, percebe-se facilmente a necessidade de mecanismos de segurança que garantam a confiabilidade e integridade do processo e de seus resultados. Pesquisadores da área têm definido e proposto requisitos específicos de segurança que devem ser observados no projeto desse tipo de sistema, como: correta autenticação do eleitor, sigilo e anonimato do voto, apuração exata, entre outros.

O recente protocolo criptográfico proposto por Lichtler [1] pretende atender aos requisitos de segurança de forma simples e semelhante a realidade do sistema eleitoral brasileiro. Com ele, o sistema proposto seria capaz de receber o voto de um eleitor corretamente autenticado, mantendo o anonimato do voto. Além disso, seria possível fornecer ao eleitor um comprovante de participação e um identificador de voto que permite a verificação da correta computação no resultado final, sem no entanto permitir que o eleitor deva comprovar a opção de seu voto à terceiros. Enfim, os requisitos mínimos desejados por um sistema de votação digital.

O referido protocolo foi escrito e proposto de forma conceitual e teórica, e até o início deste estudo não se encontrou nenhuma publicação a respeito de sua implementação. Portanto, não se conhece sua viabilidade de uso, nem o seu funcionamento prático em um sistema de votação. Assim, este trabalho objetivou estudar o protocolo sob o ponto de vista prático, implementando-o e validando-o como um protocolo de aplicação viável, sem no entanto ferir as características de segurança propostas pelo autor. Para isso, foi feita uma análise do protocolo a partir dos requisitos de segurança e de implementação para sistemas de votação encontrados na literatura.

O escopo deste estudo não abrange questões de disponibilidade de serviços computacionais, nem de escalabilidade e desempenho em votações de larga escala. O estudo limita-se à aplicação prática dos aspectos de confiabilidade e integridade do processo de votação digital.

2 VOTAÇÃO DIGITAL

O objetivo de um sistema de votação digital é conduzir o processo de votação sobre redes de computadores. Esse processo é composto por vários procedimentos, que podem ser divididos em quatro fases a saber: configuração do ambiente, credenciamento de eleitores, votação e apuração dos resultados.

Numa votação convencional, fraudes são prevenidas usando medidas físicas de segurança, envolvendo autoridades que fiscalizam o processo, garantindo o cumprimento dos devidos requisitos de segurança. Em votações digitais não existe o ambiente físico de votação, portanto, é necessária uma política de segurança que atenda aos mesmos requisitos de uma votação convencional, como autenticação e sigilo. Essa política baseia-se na identificação das possíveis ameaças ao sistema, tentando protegê-lo das mesmas.

Um sistema de votação digital é, portanto, uma aplicação distribuída, dotada de mecanismos e protocolos criptográficos que fornecem a devida segurança ao processo de votação, que se realiza sobre uma rede de computadores, considerando a possibilidade de seus próprios participantes legítimos assumirem um comportamento malicioso [2].

2.1 Requisitos

Os requisitos de segurança propostos por Riera [3] são:

- **Exatidão:** um sistema de votação é exato se, (1) não é possível alterar uma cédula válida; (2) toda cédula válida é contada na apuração; (3) nenhuma cédula inválida é contada na apuração.
- **Democracia:** um sistema de votação é democrático se, (1) apenas autorizados podem participar da votação; (2) cada votante pode votar apenas uma única vez.
- **Privacidade:** um sistema de votação garante privacidade se, (1) não há a possibilidade de associação entre voto e votante (anonimato); (2) nenhum votante pode provar qual foi seu voto (não-coação); (3) todos os votos permanecem em segredo até o fim da fase de votação (imparcialidade).
- **Verificabilidade:** há dois tipos de verificabilidade: universal e individual. Tem-se verificabilidade universal quando qualquer entidade pode certificar-se de que todas as cédulas foram contabilizadas corretamente na fase de apuração. Por outro lado, o sistema é individualmente verificável quando cada votante tem a possibilidade de verificar a contabilização do seu próprio voto.

Além dos requisitos de segurança, Cranor e Cytron [4, 5] e Riera [3] expõem três requisitos de implementação. Tais requisitos visam atender as expectativas dos usuários, no que se refere a utilização prática do sistema. São eles:

- **Conveniência:** um sistema de votação digital é conveniente se, (1) o voto é emitido de forma rápida; (2) o voto é emitido a partir de uma única sessão; (3) é exigido o mínimo de equipamento e habilidades especiais do votante.
- **Flexibilidade:** um sistema de votação digital é flexível, se é permitido uma variedade de formatos de cédula, permitindo inclusive perguntas de respostas abertas.
- **Mobilidade:** um sistema de votação digital é móvel, se não há restrições quanto ao local em que o votante emite seu voto, considerando-se as restrições impostas pela rede.
- **Escalabilidade:** um sistema de votação digital é escalável, se o mesmo permite a participação de uma quantidade indefinida de votantes.

Há ainda dois requisitos adicionais considerados a princípio opcionais, mas que se tornam importantes em votações onde o voto é obrigatório. São eles: (1) o sistema deve poder determinar quem votou e quem não votou; (2) o eleitor deve receber um comprovante de votação. O primeiro foi proposto por Schneier [6]. O segundo, é um requisito do atual sistema eleitoral brasileiro.

2.2 Protocolo Criptográfico Lichtler

O protocolo criptográfico de Lichtler [1] visa atender os requisitos de segurança. Este protocolo é fundamentado sobre três centrais eleitorais, a saber: Central de Credenciamento (CC), Central de Votação (CV) e Central de Apuração (CA).

Na ocasião do alistamento, faz-se necessário cadastrar os dados de identificação do votante, e principalmente a sua chave pública. A geração da chave pública fica a critério da implementação proposta - no presente trabalho, empregou-se o esquema RSA, dada a sua simplicidade, buscando apenas ilustrar a possível implementação de um protótipo. Além disso, o votante deve receber a chave pública da CV, que será usada durante a fase de votação. Lichtler [1] sugere que a identificação e o credenciamento do votante seja feito de forma pessoal, com base em documentos tradicionais.

Feito isto, pode-se dar início ao processo de votação. Este protocolo é baseado em criptografia assimétrica, portanto, é necessário que cada participante do processo tenha

gerado o seu par de chaves para as funções de assinatura (S) e cifragem (E) de mensagens. O gerenciamento das chaves pode ficar por conta da aplicação. São também utilizados números pseudo-aleatórios para a validação (val) do voto, bem como para a verificação (ver) do mesmo por parte do eleitor. Sejam $S_X(M)$ o efeito de um ator X assinar a mensagem M, e $EK_X(M)$ o efeito de um remetente cifrar a mensagem M com a chave pública de X. Então os passos do protocolo são os seguintes:

Para cada votante V:

M1: V → CC $S_V(\text{idvotacao})$
M2: CC → V $EK_V(\text{cedula}, \text{val}_V)$
M3: CC → CV $S_{CC}(EK_{CV}(\text{dados}_V))$
M4: CC → CA $S_{CC}(EK_{CA}(\text{val}_V))$
M5: CV→ V $S_{CV}(EK_V(K_{CA}))$
M6: V → CV $S_V(EK_{CA}(\text{voto}_V, \text{val}_V, \text{ver}_V))$
M7: CV → V $S_{CV}(\text{comprovante}_V)$

Para cada eleição:

M8: CV → CA $S_{CV}(EK_{CA}(\text{voto}_V, \text{val}_V, \text{ver}_V))$

3 ANÁLISE DO PROTOCOLO LICHTLER

A análise mostrou que o protocolo não atende de forma satisfatória os requisitos de segurança. Além disso, notou-se que o protocolo pode ser otimizado para que se tenha um melhor desempenho e praticidade de uso em uma aplicação.

A seguir são listados os problemas levantados, bem como as soluções propostas:

- **Problema 1: CV deve ser confiável.** Esta central tem condições de quebrar a privacidade do eleitor e adulterar cédulas válidas, caso venha a comportar-se de forma maliciosa, informando ao eleitor uma chave da CA falsa.
- **Solução: CC passa a fornecer a chave da CA.** O eleitor passa a receber a chave pública da CA juntamente com a cédula em branco. Além de resolver o problema essa modificação diminui o número de mensagens na comunicação, vindo ao encontro do que propõe o requisito de conveniência.
- **Problema 2: CC deve ser confiável quanto a geração de cédulas.** Esta central pode vir a agir de forma maliciosa e gerar cédulas incompletas e incorretas. Isso pode comprometer o resultado final, uma vez que certas opções de voto pode não constar na cédula e conseqüentemente receber menos votos.
- **Solução: Auditar a CC.** Fiscalizar a central de credenciamento de forma rigorosa, auditando e assinando digitalmente o código fonte.
- **Problema 3: Falta de crença na CC por parte do eleitor.** A CC não assina digitalmente a cédula em branco quando a envia para o eleitor. O eleitor precisa crer que está recebendo a cédula de uma central de credenciamento confiável.
- **Solução: CC passa a assinar a cédula.** Com isso, a fase de credenciamento muda. Além de fornecer a chave pública da CV, passa-se a fornecer também a chave pública da CC, para que o eleitor possa autentica-la a fim de garantir a credibilidade necessária.
- **Problema 4: Falta de especificação de regras quanto a geração de cédulas.** A geração de mais de uma cédula com números de validação diferentes para o mesmo eleitor pode gerar erros na contabilização geral, prejudicando a exatidão por deixar de contar cédulas válidas.
- **Solução: Controlar a geração de números de validação por eleitor.** A CC pode enviar mais de uma vez a cédula para o mesmo eleitor, desde que haja um

controle rígido, armazenando o número de validação de forma cifrada na primeira solicitação, de forma que este seja reutilizado no caso de uma segunda solicitação.

- **Problema 5: CC deve ser confiável quanto a integridade da lista de votantes.** Esta central pode vir a agir de forma maliciosa e adulterar a lista de votantes criando eleitores falsos, ou excluindo eleitores válidos.
- **Solução: Dividir a responsabilidade da integridade da lista com a CV.** A central de votação poderia agir como fiscal da CC se recebesse antes do início da votação uma lista íntegra e válida de todos os eleitores autorizados. Para isso, basta que a fase de credenciamento seja rigorosamente fiscalizada, e que ao final dessa fase seja enviada a lista completa dos eleitores à CV. Com isso, elimina-se a mensagem M3 da fase de votação, e cria-se uma mensagem no fim da fase de credenciamento onde todos os dados dos eleitores são enviados de uma única vez. Além de resolver a vulnerabilidade essa modificação torna o processo de emissão de voto mais rápido vindo ao encontro do que propõe o requisito de conveniência.
- **Problema 6: Conspiração entre as centrais.** Uma possível conspiração entre a CC e a CV comprometeria os requisitos de exatidão e democracia. A conspiração entre a CC e a CA comprometeria o anonimato do eleitor, e a conspiração entre CV e CA comprometeria o anonimato do eleitor e a imparcialidade dos resultados.
- **Solução:** A proposta original do protocolo já previu possibilidades de conspiração, e como solução sugere a fiscalização. Como o protocolo é fundamentado na comunicação restrita entre as centrais, o presente trabalho não tem o propósito corrigir essas vulnerabilidades.
- **Problema 7: Processamento desnecessário de criptografia em certas mensagens.** O protocolo original especifica que a cédula em branco bem como os dados dos eleitores devem ser totalmente cifrados. Isso pode comprometer o desempenho do sistema dependendo do tamanho desses pacotes.
- **Solução: Cifrar somente o necessário.** O único dado da cédula em branco que realmente deve ser cifrado é o número de validação, que por sinal não precisa ser conhecido pelo eleitor mas sim pela CA, portanto, deve ser cifrado a esta central. Além disso, os dados do eleitor não precisam ser obrigatoriamente cifrados.
- **Problema 8: Operação manual do sistema por seres humanos.** Para que a CA possa iniciar a apuração é necessário que esta receba os votos da CV e os números de validação da CC. Há portanto, a necessidade de operação manual nas três centrais para iniciar este processo.
- **Solução: Automatizar o processo de apuração.** Criar uma nova mensagem enviada pela CC à CV indicando o término do prazo de votação, assim, a CV encerra seus trabalhos enviando os votos válidos para a CA. Em seguida a CC envia os números de validação para a CA que finalmente pode iniciar a contabilização dos votos.

3.2 Protocolo modificado

O fluxo da comunicação entre os atores do sistema é apresentado na Figura 1.

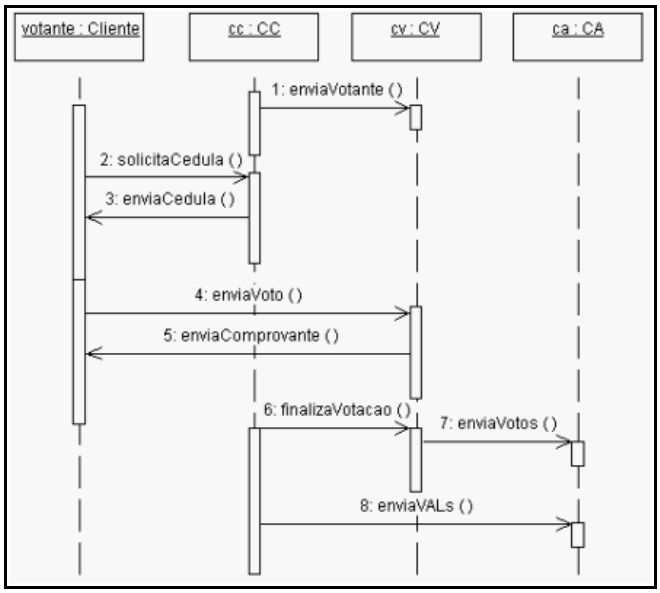

Figura 1 – Diagrama de Sequência – Protocolo de Lichtler modificado

Detalhamento das mensagens:

Fase de Credenciamento (cada votação):

M1: CC → CV $S_{CC}(dados_V, K_V)$ *(antiga M3)*

Fase de Votação (cada votante):

M2: V → CC $S_V(idvotacao)$ *(antiga M1)*

M3: CC → V $S_{CC}(cedula, K_{CA}, EK_{CA}(val_V))$ *(antiga M2)*

M4: V → CV $S_V(EK_{CA}(voto_V, val_V, ver_V))$ *(antiga M6)*

M5: CV → V $S_{CV}(comprovante_V)$ *(antiga M7)*

Fase de Apuração (cada votação):

M6: CC → CV $S_{CC}(idvotacao)$ *(nova)*

M7: CV → CA $S_{CV}(EK_{CA}(voto_V, val_V, ver_V))$ *(antiga M8)*

M8: CC → CA $S_{CC}(EK_{CA}(val_V))$ *(antiga M4)*

Este protocolo modificado pretende ser uma proposta de implementação simples e viável, mantendo as características de segurança da versão original e melhorando alguns pontos de falha detectados neste estudo. O Quadro 1 apresenta uma síntese da análise de segurança do protocolo modificado.

Requisito de segurança	Análise
A cédula não pode ser alterada	Atende, desde que não haja **conspiração entre CC-CV**.
Toda cédula válida é contada na apuração	Atende, por meio do mecanismo de verificação individual de voto.
Nenhuma cédula inválida é contada na apuração	Atende, desde que a **CC seja confiável** e gere cédulas corretamente.
Apenas votantes autorizados participam da votação	Atende, desde que haja fiscalização durante a fase de credenciamento, e que não haja **conspiração entre CC-CV**.
Cada votante emite somente um voto	Atende. A CV faz esse controle.
Anonimato	Atende, desde que não haja **conspiração entre CC-CA e CV-CA**.
Não-Coação	Atende. Não há como o eleitor comprovar o resultado do seu voto.
Imparcialidade	Atende, desde que não haja **conspiração entre CV-CA**.
Verificabilidade	Atende a verificabilidade individual, pois cada votante pode verificar a correta contabilização do seu voto.

Quadro 1 – Síntese da análise de segurança da versão modificada.

4 PROTÓTIPO DO PROTOCOLO MODIFICADO

O protótipo abrange implementa as fases de votação e apuração, por serem as fases mais relevantes e complexas do processo. Entretanto, não seria interessante apresentá-las sem antes configurar o ambiente e credenciar eleitores. Por isso, essa sessão apresenta os passos para a configuração e o credenciamento a serem executados manualmente.

4.1 Desenvolvimento

O protótipo é composto por três módulos servidores e um módulo cliente que funcionam exatamente conforme a especificação do protocolo modificado. O módulo cliente é a interface do votante com o sistema distribuído, interagindo com os servidores CC e CV.

Todos os módulos foram desenvolvidos na linguagem Java utilizando o IDE Eclipse por ser uma ferramenta livre. Java foi adotada devido a sua portabilidade. Além disso, Essa plataforma oferece muitas bibliotecas que facilitam o desenvolvimento de aplicações distribuídas e de rotinas de segurança. O J2SDK 1.5 possui o provedor de rotinas criptográficas *SunJCE*, que implementa o algoritmo RSA adotado pelo protótipo.

Para o gerenciamento de chaves foi utilizada a ferramenta *keytool* do Java, que permite gerar, armazenar e manipular as chaves e os certificados digitais em uma keystore que funciona como um repositório seguro de chaves.

A comunicação entre os módulos foi implementada com a tecnologia de RMI do Java, onde o remetente invoca um método remoto do destinatário passando a mensagem como parâmetro do método. Para isso, cada central define e implementa um método remoto para cada tipo de mensagem que ela deve receber.

Cada central eleitoral possui um banco de dados MySQL, por ser um sistema gerenciador de banco de dados livre e amplamente utilizado.

Os serviços criptográficos exigidos pelo protocolo foram definidos em uma interface de objeto, para que possam ser implementados de diversas formas com diferentes técnicas e algoritmos criptográficos. O Quadro 2 apresenta o código da interface Seguranca, definida no pacote comum.

```
public interface Seguranca {
          public MensagemAssinada signObj(Serializable obj,
                              String alias, char[] pass);
          public String checkSign(MensagemAssinada m);
          public String cifra(String msg, Key chavePublica);
          public String decifra(String msg, String alias,
                                                 char[] pass);
          public Certificate getCertificate(String certAlias);
          public boolean setCertificate(String certAlias,
                                        Certificate cert);
          public double geraPseudoAleatorio();
}
```

Quadro 2 – Interface dos serviços de segurança.

A classe `SegurancaRSA` do pacote `comum` implementa essa interface utilizando os recursos oferecidos pelo componente SunJCE. Os métodos `getCertificate` e `setCertificate` manipulam a *keystore* do Java armazenando ou recuperando um certificado digital com base num nome identificador.

O método `signObj` assina um objeto, para isso deve-se informar a senha e o

identificador da chave privada armazenada na *keystore*. O método `checkSign` verifica um objeto assinado e retorna um valor booleano indicando a autenticidade da assinatura. Ambos os métodos trabalham com a classe `MensagemAssinada` que define a estrutura de um objeto assinado. Essa mensagem assinada é o pacote final que deve ser enviado pela rede em todas as trocas de mensagens no protocolo. A mensagem contém o objeto assinado e o próprio certificado digital de quem o assinou.

O método `cifra` recebe o texto e a chave pública a ser usada na cifragem, retornando o texto cifrado. O método `decifra` recebe o identificador da chave privada e uma senha que permite recuperar essa chave da *keystore*, a fim de usá-la na decifragem.

O método `geraPseudoAleatorio` gera os números de validação e de verificação definidos no protocolo. Esses números devem ser secretos, e por isso foi utilizada a classe `SecureRandom` da plataforma Java juntamente com o algoritmo SHA1PRNG implementado no componente SunJCE. O método generateSeed gera uma semente aleatória do tamanho indicado no argumento.

4.2 Funcionamento do protótipo baseado na alteração proposta

Na fase de configuração, todo o ambiente é instalado e configurado para atender o processo de votação. Os passos são:

1. Definir centrais eleitorais e seus endereços de rede, gerando o par de chaves e emitindo o certificado digital para cada;
2. Instalar e configurar o módulo da central de credenciamento, cadastrando a votação, as opções de voto e os dados das demais centrais eleitorais que atuarão no processo.
3. Instalar e configurar o módulo da central de votação, cadastrando os dados das demais centrais eleitorais que atuarão.
4. Instalar e configurar o módulo da central de apuração, cadastrando os dados das demais centrais eleitorais que atuarão.

Todas os passos dessa fase devem ser acompanhados por fiscais, a fim de garantir a correta configuração de cada central. Além disso, a CC deve ser confiável quanto a geração de cédulas, por isso, faz-se necessário uma auditoria no código fonte, seguido de sua compilação e assinatura digital do código compilado.

Na fase de credenciamento, os votantes são credenciados a participar de determinada votação. A responsabilidade por este processo é da central de credenciamento, e os passos são os seguintes:

1 Autenticar o eleitor de forma pessoal, através de documentos tradicionais de identificação.
2 Gerar o par de chaves, emitir o certificado digital e cadastrar os dados do eleitor.
3 Fornecer ao eleitor os certificados digitais da CV e da própria CC.

Ao fim do período de credenciamento, os dados dos eleitores e seus certificados digitais são assinados e enviados à central de votação correspondente da seguinte forma: M1: CC → CV $S_{CC}(dados_V, K_V)$

Os certificados digitais fornecidos ao eleitor inclusive seu próprio par de chaves gerado é armazenado em um arquivo de chaves de forma cifrada, e fornecido ao eleitor através de um disquete. O passo 4 do credenciamento é a primeira mensagem de comunicação entre centrais, sendo de fundamental importância para o início da fase de votação. Por isso, esse passo foi implementado no protótipo.

Todos os passos devem ser acompanhados por fiscais, a fim de garantir a integridade da lista de eleitores recebida pela central de votação, e ao final dessa fase, todos atores participantes devem possuir uma *keystore* configurada com suas chaves privadas e com os certificados digitais dos demais atores que se comunicarão nas fases de votação e apuração.

A fase de votação é composta basicamente por duas rotinas, a de solicitação de cédula em branco (M2 e M3), e a de depósito de voto (M4 e M5). A votação inicia quando o eleitor solicita a cédula em branco de determinada eleição assinando o pedido. Em seguida a CC verifica a assinatura e o credenciamento do eleitor e gera uma cédula assinada, composta pela questão da votação e suas opções de votos, pela chave pública da CA e pelo número de validação cifrado à CA. Sendo a primeira solicitação desde eleitor, o número cifrado é armazenado para que em caso de futuras solicitações seja reutilizado para o mesmo eleitor.

O eleitor verifica a assinatura da CC e faz seu voto gerando um número pseudo-aleatório de verificação. É então gerado um pacote assinado contendo o voto, e os números de validação e verificação cifrados à CA, para emissão à CV.

A CV recebe o voto e verifica a assinatura do eleitor certificando-se se o mesmo já não votou anteriormente. Sendo um eleitor credenciado, o voto é armazenado na central para futuro envia à CA, e em seguida é emitido um comprovante ao eleitor que vem a ser o próprio voto cifrado à CA e assinado pela CV.

Finalmente o eleitor recebe o comprovante de participação e verifica a autenticidade do mesmo mediante a verificação de assinatura da CV. Sendo um comprovante válido, ele deve ser armazenado em disco. Nesse momento é apresentado uma tela de finalização da votação com o número de verificação de voto, para que o eleitor certifique-se da correta contabilização no resultado final. A Figura 2 apresenta as janelas do protótipo no momento em que o eleitor recebe o comprovante de participação.

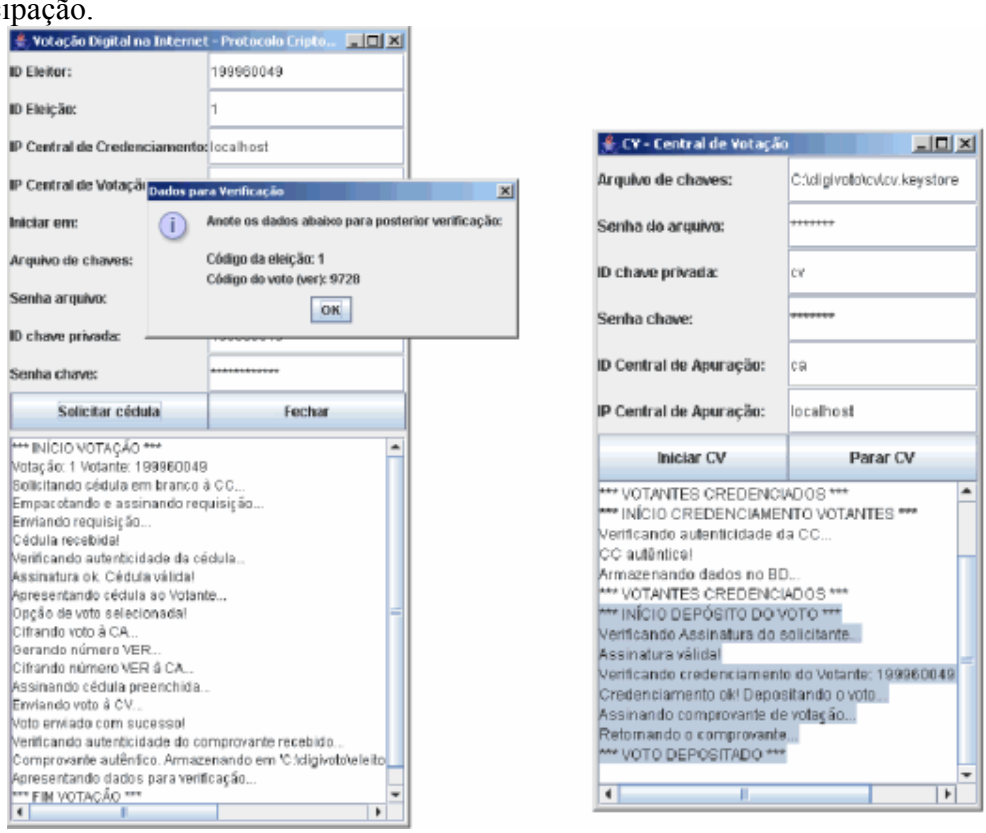

Figura 2 – Votação – Depositando voto.

A fase de apuração inicia com o encerramento do período de votação, quando a CC notifica à CV, enviando o identificador da eleição de forma assinada. A CV verifica a autenticidade da mensagem e em seguida assina e repassa à CA os votos armazenados. Em seguida a CC assina e envia todos os números de validação para a CA começar a apurar os votos.

5 CONCLUSÕES

A partir da análise do protocolo de Lichtler feita neste trabalho foi possível identificar oito problemas, sendo quatro vulnerabilidades que comprometem a segurança do processo, uma falha que compromete a crença na autenticidade da central de credenciamento, um ponto de ocorrência de erros que compromete a exatidão dos resultados, e duas questões que podem dificultar a aplicação prática do protocolo.

As vulnerabilidades mais graves detectadas envolvem as próprias entidades legítimas do sistema. Além do problema da conspiração entre as centrais – já previsto pelo autor do protocolo – há outras vulnerabilidades que permitem a violação do processo por uma única entidade. Nesse sentido, tanto a central de credenciamento como a de votação poderiam violar o sistema de forma independente. Considerando a possibilidade destas centrais assumirem um comportamento malicioso, conclui-se que o protocolo estudado não cumpre os requisitos de segurança de forma satisfatória.

No que se refere aos problemas de aplicação prática, conclui-se que o protocolo é de aplicação inviável. Sua implementação seria demasiadamente complexa, e como resultado teria-se um sistema não conveniente aos usuários participantes do processo. Isso ocorre porque os requisitos de implementação não foram considerados no projeto do protocolo.

Com o intuito de aumentar a segurança e viabilizar a implementação, foram propostas cinco modificações no protocolo. Além disso, foi proposta uma melhoria que detalha a especificação referente a rotina de geração de cédulas. Esse detalhamento visa evitar a ocorrência de erros de contabilização que afetam a exatidão dos resultados.

Um dos grandes avanços que as modificações contribuem ao protocolo é a redução do nível de confiança nas centrais eleitorais. No caso da central de votação, não há mais possibilidades dessa entidade violar o processo de forma independente. Porém, a central de credenciamento continua com uma vulnerabilidade que permite a ela a geração de cédulas falsas ou incompletas aos eleitores. Entretanto este problema pode ser contornado facilmente com uma auditoria na central. Esta solução funciona como um requisito para que o protocolo trabalhe de forma segura.

Aplicando as modificações à proposta de Lichtler chegou-se a uma nova versão de protocolo, o qual foi implementado em um protótipo de votação digital. Como o objetivo do protótipo foi apresentar, na prática, o funcionamento e a viabilidade de uso do protocolo, destacamos que as limitações deste estudo não consideraram definições formais matemáticas de algoritmos criptográficos, nem com restrições e mecanismos de segurança oferecidos pelos sistemas gerenciadores de banco de dados. Assim, foi empregado o RSA sem as preocupações referentes a ataques a ele deferidos. Além disso, o protótipo não implementou os processos de fiscalização das centrais, sugeridos como alternativa de solução às vulnerabilidades não resolvidas pelas modificações propostas. Entretanto, essas limitações não comprometem o funcionamento do protótipo para fins acadêmicos, dentro que que prevê o protocolo Lichtler – a exemplo de outros, assume-se que as centrais não devem integrar informações com o intuito de

determinar origem do voto, identificar eleitores de determinado voto etc. Essa última situação constitui verdadeiro problema de ordem real: a conspiração entre as centrais eleitorais. Este problema é colocado como uma restrição ou condição mínima no projeto do protocolo, e a solução proposta pelos autores foi utilizar a fiscalização como um requisito para garantir que a comunicação entre as centrais seja restrita ao que foi especificado pelo protocolo.

Esse protótipo não pretende ser uma proposta de solução definitiva, pois o protocolo modificado ainda não atende a alguns detalhes dos requisitos de segurança. Apesar disso, houve progressos observáveis, pois a versão modificada agrega maior confiabilidade, e o protótipo implementado mostrou na prática sua viabilidade de uso em sistemas de votação digital.

REFERÊNCIAS

[1] LICHTLER, R. L. **Um Estudo Sobre Protocolos Criptográficos para Votações Digitais.** Trabalho de Diplomação, 2000. Porto Alegre: Instituto de Informática -UFRGS.

[2] RIERA, A. **An Introduction to Electronic Voting Schemes.** Universitat Autònoma de Barcelona, 1999.

[3] RIERA, A. **Design of Implementable Solutions for Large Scale Electronic Voting Schemes.** Universitat Autònoma de Barcelona, 1999.

[4] CRANOR, L.; CYTRON, R. **Design and Implementation of a Practical Security-Conscious Electronic Polling System**. *Washington University Report WUCS-96-0*2, Washington University.

[5] CRANOR, L.; CYTRON, R. **Sensus: A Security-Conscious Electronic Polling System for the Internet.** *Proceedings of the Hawai I International Conference on System Sciences* (IEEE).

[6] SCHNEIER, Bruce. **Applied Cryptography**. New York: John Wiley & Sons, 1995.