Subversão de contêiner criptográfico e alteração de dados na urna eletrônica brasileira

Galileu B. de Sousa[1,2], Ivo de C. Peixinho[1], Paulo C. H. Wanner[1]

[1]Polícia Federal
Brasília – DF – Brasil

{galileu.gbs, peixinho.icp, herrmann.pchw}@pf.gov.br

[2]Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte
Natal – RN – Brasil

Abstract. *This paper describes the results of the authors on the Security Public Trials – TPS, 2019 edition, offered by the Election Authority in Brazil. It was possible to subvert a cryptographic container on the computer that performs preparation of software used by the electronic ballots, including deciphering of private keys used by the GEDAI-UE software. The vote and the ballot secrecy were not violated; however, the name of the polling place was changed, and a valid card was generated and correctly executed by the electronic ballot, without any violation complaints. Changes made by the Election Authority technicians were assessed during a re-trial event, when the cryptographic container was again subverted, but the keys weren't deciphered because of additional protections.*

Resumo. *Este artigo descreve a participação dos autores nos Testes Púbicos de Segurança – TPS, edição 2019, oferecidos pelo TSE a cada eleição ordinária. Os autores foram bem-sucedidos em subverter um contêiner criptográfico na estação que realiza a preparação do software para instalação nas Urnas Eletrônicas, incluindo a decifragem de chaves privadas usadas pelo programa GEDAI-UE. O voto e o seu sigilo não foram violados, contudo, foi alterado o nome do local de votação e gerado um cartão válido que foi corretamente executado pela urna eletrônica, sem detecção da alteração. Correções realizadas pelos técnicos do TSE para sanar as vulnerabilidades foram avaliadas durante testes de confirmação, quando o contêiner criptográfico foi novamente subvertido, porém as chaves não foram decifradas por conta das proteções adicionais.*

1. Introdução

A urna eletrônica (UE) usada no Brasil completou vinte e cinco anos desde sua primeira aparição em 1996. Durante esses anos sofreu diversas melhorias, objetivando fundamentalmente a segurança e auditoria do processo. Evoluções significativas nessa área começaram a aparecer a partir do ano de 2009, ano em que começaram a ser organizados pelo Tribunal Superior Eleitoral (TSE) os Testes Públicos de Segurança (TPS).

Esses testes públicos têm o objetivo de fortalecer a confiabilidade, a transparência e a segurança da captação e da apuração dos votos, além de propiciar melhorias no processo eleitoral [TSE 2019]. A avaliação de sistemas de votação eletrônica ocorreu em diversos países, tais como Argentina [Amanto et al 2015], Estados Unidos [Calandrino et al 2007], Holanda [Gonggrijp e Hengeveld 2007] e Índia [Wolchok et al 2010].

No Brasil, os testes começaram a ser obrigatórios e a fazer parte do ciclo de desenvolvimento dos sistemas eleitorais de votação, apuração, transmissão e recebimento de arquivos das eleições brasileiras a partir de 2015 com a publicação da Resolução do TSE de nº 23.444/2015. O evento ocorre regularmente nos anos anteriores aos pleitos eleitorais e permite a participação de qualquer cidadão brasileiro, maior de 18 anos. Interessados submetem previamente planos de testes que são avaliados pelo TSE e, se aprovados, podem participar da edição do TPS nas dependências do tribunal.

Inicialmente, os testes foram direcionados à urna eletrônica e aos seus componentes propriamente e, conforme foram evoluindo, mais sistemas começaram a fazer parte do escopo dos testes. A Figura 1 mostra o ecossistema de votação brasileiro, contemplando os vários componentes e destacando aqueles que fazem parte dos testes públicos de segurança. Ainda na Figura 1, observa-se o Sistema Gerenciador de Dados, Aplicativos e Interface com a Urna Eletrônica (GEDAI-UE), que, simplificadamente, serve de interface entre os sistemas de cadastro de eleitores e candidatos do TSE e a urna. É função do GEDAI-UE gerar os cartões de memória que alimentam as urnas – nessas memórias estão tanto o software de controle do processo de votação, quanto os dados da seção eleitoral. Em geral, um cartão é usado para alimentar várias seções de uma mesma zona eleitoral. O GEDAI-UE ainda é responsável por armazenar os dados das seções e controlar as urnas que foram efetivamente carregadas, por meio de uma tabela de correspondência, para a correta vinculação durante o processo de apuração.

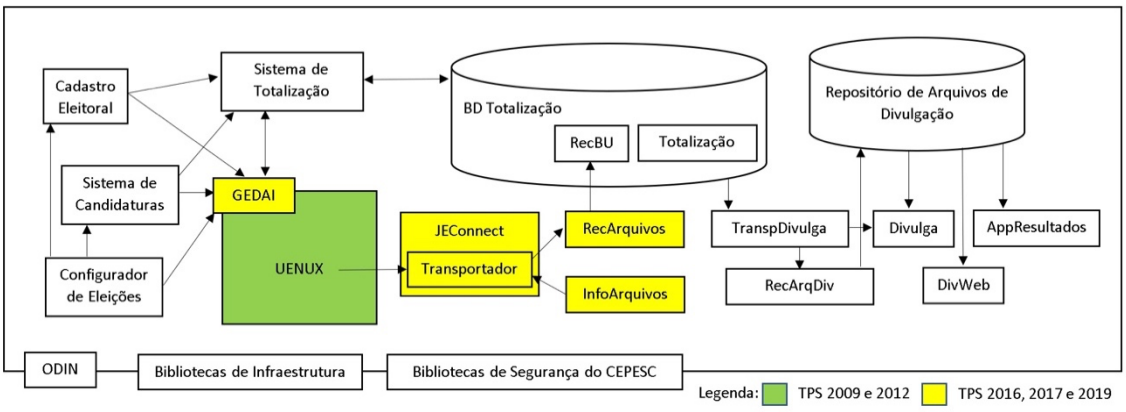

Figura 1. Ecossistema da Votação Eletrônica no Brasil

A segurança da urna eletrônica evoluiu substancialmente durante esses vinte cinco anos através de revisões do código-fonte por órgãos de controle como a Polícia Federal, partidos políticos [Filho et al 2014] e academia [Aranha et al 2014]. Durante os testes públicos de segurança mais contribuições da comunidade científica foram fundamentais para e evolução dos mecanismos de segurança utilizados pela urna eletrônica [Aranha et al 2018]. As soluções adotadas pelo TSE apoiam-se fortemente na criptografia e assinatura digital de seus programas [Monteiro et al. 2019].

Consequentemente, espera-se que nesse cenário os ataques passem a ocorrer em pontos mais frágeis e desguarnecidos do sistema, como exemplo, pode-se citar a pesquisa de Souza et al. (2017) e as recentes notícias de ataques aos sistemas do TSE que se tornaram públicas [G1 2018].

O presente artigo visa contribuir com melhorias na segurança do sistema da urna eletrônica relatando os resultados obtidos na edição de 2019 dos testes públicos de segurança e no respectivo teste de confirmação em 2020. Na ocasião, os autores conseguiram subverter as proteções na máquina que executa o GEDAI-UE e obtiveram o conteúdo do contêiner criptográfico, incluindo uma chave privada usada para assinar dados consumidos pela urna. Posteriormente, foi possível a modificação do programa próprio GEDAI-UE, de modo a assinar dados espúrios que foram aceitos pela urna.

Deve ser observado que nem o voto, nem o seu sigilo foram violados. De toda forma, os dados falsos passaram a constar no documento emitido no início da votação, a *zerézima*, e em um dos produtos gerados pela urna após o final da votação, o boletim de urna. Essas vulnerabilidades se não sanadas poderiam levar a desconfiança no processo eleitoral como um todo.

Organização. A seção 2 descreve os planos de testes propostos pelos autores, enquanto a seção 3 descreve a execução dos testes e os seus achados. Por fim, nas seções 4 e 5 são discutidos os resultados e as conclusões.

2. Planos de Testes

Visando ampliar a segurança do ecossistema como um todo, os autores apresentaram planos de testes direcionados à validação da segurança do gerenciamento das chaves de criptografia e da interface entre o GEDAI-UE e a urna eletrônica. Os esforços se concentraram no GEDAI-UE e no Subsistema de Instalação e Segurança (SIS), conforme detalhado adiante.

O GEDAI-UE é responsável por gerar os cartões de instalação das urnas eletrônicas denominados Flashes de Carga (FC), sendo executado em uma máquina Windows, com a segurança robustecida pelo SIS. Este subsistema faz inúmeras adequações no sistema operacional, incluindo a completa substituição do sistema de autenticação, o controle da execução de processos e a autoridade sobre o gerenciamento de permissões do sistema de arquivos.

Além do GEDAI-UE, outros programas usados pela Justiça Eleitoral para execução das rotinas da eleição executam em máquinas protegidas pelo SIS. A maior parte desses programas, arquivos de configuração e chaves criptográficas a serem copiados para a urna estão armazenados em um volume criptografado (nomeado como "`SE.pzm`") usando o aplicativo Truecrypt. O volume é gerenciado pelo SIS e aberto somente após a autenticação de um usuário, sendo fechado durante o *logout*. Essa sistemática impede, em princípio, ataques *offline* ao disco desses computadores.

Nesse contexto, a urna eletrônica somente torna-se operacional após a instalação do sistema operacional, softwares e demais dados de candidatos e eleitores, mediante a inserção do FC gerado pelo GEDAI-UE em procedimento repetido a cada nova eleição. Arquivos com candidatos e eleitores são importados pelo GEDAI-UE de outros aplicativos do TSE e copiados juntamente com o sistema operacional Linux modificado

pelo tribunal (UENUX) para o interior da urna [TSE 2021]. Usando os FC, os técnicos dos tribunais regionais configuram as urnas para a eleição.

A partir dessas informações, os autores executaram dois planos de testes que estão sumarizados na Tabela 1.

Tabela 1. Planos de testes executados pelos autores no TPS 2019

Título	Objetivos
Plano 1 - Extração do conteúdo do disco criptografado do SIS	1. obter acesso físico ao disco do computador com o GEDAI-UE instalado para retirar o disco criptografado e buscar a chave no registro do Windows; 2. inicializar o disco em uma máquina virtual para obter um *dump* de memória; 3. extrair a chave a partir do *dump* e comparar com as informações obtidas no registro para estabelecer o processo de formação da chave; 4. montar o disco cifrado e extrair dele os dados decifrados; 5. verificar, no disco cifrado, informações sensíveis para o processo eleitoral.
Plano 2 - Instalação e execução de código arbitrário em uma máquina do GEDAI-UE para implante de dados falsos na Urna Eletrônica.	1. obter acesso físico ao computador com o GEDAI-UE instalado para fazer uma imagem completa do disco; 2. inicializar o disco em uma máquina virtual; 3. subverter o sistema de inicialização para viabilizar o *boot* sem a carga do SIS; 4. acessar e modificar programas de criação e preparação de dados a serem gravados nas urnas eletrônicas; 5. criar um cartão de inicialização da urna com dados espúrios.

3. Execução

As severas restrições impostas pelo SIS exigiram que os testes fossem executados a partir de um ambiente de máquina virtual. A virtualização facilita, por exemplo, a realização de *dumps* de memória. Assim, a primeira ação foi a duplicação do disco rígido do computador contendo os sistemas SIS/GEDAI-UE. Em paralelo à duplicação, os autores desenvolveram outras atividades, ainda que a descrição no presente texto possa indicar um trabalho sequencial.

3.1. Extração da senha do volume cifrado

A análise do código-fonte do SIS evidenciou que a chave usada para abrir o volume cifrado é gerada no momento da instalação, usando uma função padrão do Windows que cria um CLSID – identificador único global. Trata-se de uma sequência de 32 caracteres hexadecimais no seguinte formato: {00000000-1111-2222-3333-444444444444}. A chave de criptografia é derivada a partir de uma série de inversões, permutações e substituições sobre esse CLSID. Ao final, ela é armazenada em um arquivo denominado

"SE.hsh" fora do drive criptografado e em uma chave do Registro do Windows presente em "HKLM\Software\Modulo\SisVolKsa\KsaVol".

Após a identificação do algoritmo que deriva a chave do volume cifrado, um programa que mimetiza a obtenção da senha do volume cifrado do SIS a partir do arquivo "SE.hsh" foi criado. A Tabela 2 apresenta o pseudocódigo do algoritmo. A senha obtida pelo programa desenvolvido, apesar de consistente com o esperado (32 bytes hexadecimais), não permitiu a decifragem e o acesso ao volume "SE.pzm", usando o programa Veracrypt em modo de compatibilidade com o Truecrypt.

A equipe de desenvolvimento do SIS foi consultada e informou que haveria um passo de ofuscação adicional realizado dentro de um processo servidor, parte do SIS, que teria acesso ao *device driver* do Truecrypt.

Por meio do espelhamento anteriormente realizado, foi possível inicializar o disco em uma máquina virtual usando o VirtualBox e realizar um *dump* da memória volátil (RAM), após a montagem do volume cifrado. Dessa forma, novas formas de obtenção da senha de acesso ao volume cifrado foram avaliadas, como as propostas em Halderman et al. (2009). Uma delas foi a busca pelas constantes "C" e "D" empregadas no algoritmo utilizado pelo programa de geração da senha.

Tabela 2. Algoritmo de geração da senha do volume cifrado

`A = gera_novo_CLSID()`	// Formato: {00000000-1111-2222-3333-444444444444}
`salva_hsh(A)`	// Salva o CLSID em SE.hsh e no registro
`B = inverte(A)`	// Rescreve a *string* A do final para o início
`C = "")(*&¨%$#@!@#$%¨&*()"`	// Constantes C e D usadas no processo de ofuscação
`D = "62BAAD7F2766FEFC"`	
`E = inverte(C)`	// Rescreve a *string* C do final para o início
`F = permuta_2_a_2(B)`	// Percorre B. Troca o byte corrente com o seguinte
`G = F xor C`	// C é repetido até ficar do mesmo tamanho de F
`H = subst_impares(D, G)`	// Substitui os caracteres nas posições ímpares de G // (iniciando em zero) pelos respectivos caracteres em D
`senha = conv_hexadec(H)`	// Escreve cada um dos caracteres em H na forma de // texto hexadecimal. Ex: "A5", gera "4135"

A pesquisa pelas constantes permitiu identificar um padrão de caracteres hexadecimais próximo a elas que mostrou ser efetivamente a senha de acesso ao volume criptografado. Como a senha encontrada na memória possuía 33 bytes, ao invés de 32, o último *byte* foi retirado e o disco criptografado "SE.pzm" foi montado com sucesso, utilizando-se o aplicativo Veracrypt.

3.2. Extração de chaves do volume decifrado

No volume decifrado, foram encontradas chaves privadas usadas pelo GEDAI-UE para assinar dados enviados ao cartão de carga da urna eletrônica. Em especial as chaves "osevin_GEDAI.ber.pri" e "ssevin_GEDAI.ber.pri" correspondentes à eleição oficial ("o") e simulada ("s"). Estas chaves estavam presentes tanto no diretório

"\Aplic\SEVIN\GEDAIUE\chaves\legal", quanto no diretório "\Aplic\SEVIN\GEDAIUE\chaves\comunitaria" de acordo com o tipo de eleição (a pasta comunitária se refere a eleições de entidades como OAB e CONFEA, cuja urna é emprestada pelo TSE[1]).

A ferramenta OpenSSL foi empregada para leitura dos arquivos, que estavam codificados em ASN1. Dessa forma, foi possível verificar o conteúdo das chaves em um campo de 140 *bytes*, com a identificação "SEVIN" – o nome do setor do TSE responsável por sua guarda (SEVIN/CSELE/STI/TSE).

A avaliação do código-fonte indicou que as chaves privadas armazenadas nesses arquivos tinham seus conteúdos protegidos com uma criptografia adicional, de modo que haveria um passo adicional para efetivamente obtê-las. Essa cifragem adicional utilizava o algoritmo AES256, no modo CBC, usando como chave e vetor de inicialização (IV) cadeias derivadas de um *hash* SHA512.

O *hash* é calculado sobre dados obtidos do "ou exclusivo" (XOR) de duas informações: uma cadeia de caracteres fixa: "STI – SEVIN/TSE/JUS/BR" (sem as aspas) e o conteúdo de um arquivo denominado "chaveiro.pri", encontrado dentro do volume criptografado do SIS.

No caso concreto, a operação XOR resultou em uma cadeia representando um ano seguido de um número (exemplo: "2019235") e uma série de espaços, retirados antes da aplicação do *hash* SHA512[2]. A partir do resultado do *hash*, o vetor de inicialização consiste nos 16 primeiros *bytes* (0-15) e a chave dos 32 *bytes* seguintes (16-47). O resumo desse algoritmo para decifragem das chaves privadas do "SEVIN" estão na Tabela 3.

Tabela 3. Algoritmo para decifragem das chaves privadas da SEVIN"

`A = "STI – SEVIN/TSE/JUS/BR"`	// Constante no código
`B = read("chaveiro.pri")`	// Presente em: // M:\Aplic\SEVIN\GEDAIUE\chaves

`C = A xor B`	// Observou-se que desse *XOR* resultou uma
	// string de identificação do pleito, com
	// espaços à direita.

`D = retira_espaco_direita(C)`	// Faz a remoção dos espaços (operação não // evidenciada na análise preliminar)[2].
`H = SHA512(D)`	// SHA512 da identificação do pleito.

`IV = H[0-15]`	// Usa os 16 primeiros bytes do *hash* como
	// vetor de inicialização para decifragem
	// da chave privada

`KEY = H[16-47]`	// Usa os próximos 32 bytes do *hash* como // chave para decifragem da chave privada.
`CHAVE = AES256_CBC_DECRYPT (KEY,IV, osevin_GEDAI.ber.pri)`	// A chave privada do SEVIN, // sem qualquer proteção

[1]https://www.tse.jus.br/imprensa/noticias-tse/2020/Fevereiro/voce-sabe-o-que-sao-eleicoes-comunitarias
[2]A remoção dos espaços só foi observada após análise do aplicativo GEDAI-UE, vide seção 3.4.

O algoritmo na Tabela 3 foi implementado em *Python* e a chave de decifragem da chave privada do "SEVIN" potencialmente obtida. Contudo, não foi possível validar a operação, pois as chaves (do "SEVIN") são empregadas em algoritmos de criptografia do CEPESC/ABIN, variantes dos algoritmos de curvas elípticas e do algoritmo El Gamal. Sob o argumento de serem "criptografia de Estado", nem seus modos de funcionamento, nem seus códigos-fontes foram disponibilizados para análise, contrariando princípios básicos de segurança [Kerckhoffs 1883] e transparência.

Adicionalmente, no volume cifrado, foram encontradas outras chaves, como, por exemplo, a chave privada da urna que é única por estado da federação ("`ue.ber.pri`"), dentre outras. O uso dessas chaves requer maior investigação, porém, após o TPS-2017, nos testes de confirmação de 2018, foi verificado [SEVIN, 2018] que estas chaves passaram a ser criptografadas a partir de um elemento presente fisicamente nas urnas eletrônicas (tabela criptográfica da BIOS – *Basic Input Output System*). Assim, em tese, não seria possível decifrar estas chaves sem acesso físico a componentes internos da urna eletrônica [Monteiro et al. 2019]. Diante da limitação de tempo, optou-se por não verificar esse processo.

3.3. Subversão do SIS

Os procedimentos de preparação de cartões de memória a serem inseminados na urna são complexos, de modo que mesmo com a disponibilidade do volume decifrado é inviável, em curto tempo, obter um cartão válido sem usar os sistemas do TSE. A máquina com o GEDAI-UE, contudo, é protegida pelo SIS, de modo que a execução de qualquer programa estanho ao ambiente (incluindo um *debugger*) é bloqueada.

O SIS substitui todo o subsistema de autenticação do Windows por um sistema próprio, por meio da implantação de novos *credential providers*, bibliotecas de validação de usuários e senhas para acesso à interface gráfica do Windows. Em adição, após a autenticação, os programas passíveis de execução são controlados pelo SIS e restritos àqueles previamente estabelecidos pelo TSE. Nesse contexto, foram adotados ataques *offline* ao SIS, objetivando:

a. Remoção do subsistema de autenticação definido pelo SIS;
b. Redefinição dos programas carregados após a autenticação;
c. Exclusão de serviços e *drivers* para controle de execução de programas e controle de acesso a arquivos e pastas.

O ataque *offline* foi realizado usando o "Hiren's BootCD" para inicializar uma máquina virtual com a cópia do disco da estação SIS como disco secundário. Após a carga foram alteradas diversas chaves de registro com o intuito de desabilitar os mecanismos de segurança implantados pelo SIS.

3.3.1 Remoção do subsistema de autenticação do SIS

As mudanças para restabelecer o subsistema de autenticação original do Windows demandaram a exclusão do provedor de autenticação do SIS, em "`HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Authentication\CredentialProviders`", cujos valores "`SISCP`" e "`SISCPLogoff`" indicavam

como responsáveis pela autenticação as bibliotecas "SISCP.dll" e "SISCPLogoff.dll" em "%WINDIR%\System32".

As bibliotecas foram removidas e o CLSID do provedor invalidado. Com o sistema de autenticação do SIS inacessível, o restabelecimento do subsistema original ainda exigiu uma modificação para que a interface de *login* voltasse a oferecer outros usuários para autenticação: "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Authentication\LogonUI\UserSwitch\Enabled=1".

Ainda nesse processo, foram identificados usuários presentes no Windows, entre eles: "Administrador", "Suporte", "Instalador" e "101913" (usuário que o TSE criou para uso no TPS), alguns deles desabilitados. Os usuários foram habilitados e suas senhas redefinidas para "12345678", usando ferramenta de edição de senhas do *Hirens*.

3.3.2 Redefinição dos programas carregados após a autenticação

A lista de programas inicialmente executados após um *login* bem-sucedido, definido em "HKLM\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\Winlogon\Userinit", possuía o aplicativo "C:\seguranca\util\sisMessage.exe" na sua lista. Esse aplicativo teve que ser removido, pois realizava a finalização antecipada da inicialização do sistema operacional.

O programa inicial de interação com o usuário, presente em "HKLM\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\Winlogon\Shell" teve seu valor restabelecido de "C:\seguranca\util\erroCP.exe" para "explorer.exe". Os programas específicos do TSE iniciados com o sistema e presentes como valores na chave "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" foram removidos e outros três programas presentes nesta chave tiveram a sua inicialização automática cancelada.

Com a execução dessas modificações, foi possível inicializar o sistema. Dois usuários foram autenticados com sucesso: "Suporte" e "Administrador", sendo o primeiro percebido com mais direitos de acesso. Uma vez logado, foi possível criar um usuário, por meio do programa "C:\seguranca\util\manut.exe", parte do SIS. Optou-se por um usuário correspondente ao título de eleitor de um dos autores, com os níveis de credenciais mais elevados que o programa suportava. O usuário fez um acesso bem-sucedido, porém com restrição de acesso a algumas pastas.

3.3.3 Exclusão de serviços e *drivers* de controle de acesso e execução

Em virtude de restrições de acesso, nova avaliação foi realizada usando o *Hirens* e a carga de serviços e *drivers* associados ao SIS desabilitados, entre eles: "sisdrv", "sissrv", "sic_srvc" e "SisTarefas". Um suposto processo de estabelecimento de política de segurança, contido na chave do registro "HKLM\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\SeCEdit" foi igualmente removido. Por fim, as permissões de todos os arquivos relacionados ao TSE nos discos "C:" e "D:" da máquina do SIS tiveram permissão liberada para "Everyone".

Apesar de todas as modificações, após um *boot* regular do sistema, o usuário criado ainda não possuía acesso a pastas críticas do sistema, inclusive ao arquivo com o disco criptografado ("`SE.pzm`"). Também não era possível instalar novos programas.

O acesso irrestrito somente foi possível após inspeção do arquivo de *log* "`C:\seguranca\Util\Log\SIC.log`" que fazia referência ao *driver* "`PROTFXP.SYS`". Esse *driver* se interpunha a todo o sistema de permissões do NTFS. Após a sua remoção e reinicialização, o sistema passou a operar sem quaisquer restrições para o usuário criado. O volume cifrado não foi montado automaticamente, pois os serviços responsáveis pela montagem estavam desativados. De todo modo, a senha de decifragem do volume já era conhecida e a instalação de programas para montagem explícita já era permitida.

O volume cifrado foi, então, montado manualmente, usando o Veracrypt, e executou-se o aplicativo GEDAI-UE nesse novo ambiente. O aplicativo funcionou normalmente sem a presença do SIS. Observe-se que o SIS faz uma validação de assinatura digital de todos os programas que são executados na máquina, mas como não estava mais presente, a validação não foi realizada. A partir desse ponto tornou-se possível, inclusive, realizar alterações no executável do GEDAI-UE.

3.4. Engenharia reversa do GEDAI-UE

A execução do GEDAI-UE em um ambiente sem restrições e a partir do volume decifrado viabilizou a adoção de procedimentos para identificar se a chave privada do "SEVIN" era usada e o exato momento em que os arquivos eram assinados.

Extrações de memória volátil foram então realizadas enquanto o GEDAI-UE executava, mas a chave privada SEVIN não foi encontrada. Isso gerou dúvida se o algoritmo da Tabela 3 estava corretamente implementado ou se o aplicativo apagava, após assinar os dados, a área de memória onde a chave se encontrava. A forma de validar o algoritmo foi por meio de *debugging* do GEDAI-UE.

O aplicativo GEDAI-UE estava compactado com uma versão modificada do compactador UPX. Após a descompactação manual, o programa apresentava um erro de validação, alegando que o tamanho do arquivo executável fora violado. Foi possível contornar esse erro alterando o ponto de verificação no executável. Com o avanço da análise, foi possível encontrar a rotina que realiza a assinatura de arquivos, de modo que, a chave foi encontrada como um dos parâmetros dessa rotina. Infelizmente, a chave não coincidia com a do algoritmo da Tabela 3.

Verificou-se que para o algoritmo funcionar bastava remover os espaços do resultado do XOR. A chave resultante era idêntica à encontrada dentro do GEDAI-UE, com exceção dos bytes finais (posições 48-63, não usados). Na decifragem, eles eram "0x0C" e na rotina eles eram "0xAE". Ou seja, o programa na Tabela 3 foi capaz de decifrar as chaves privadas da "SEVIN" referentes as eleições "Oficial" e "Simulada", tanto do diretório "LEGAL" quanto do diretório "COMUNITARIA", de modo que quatro diferentes chaves privadas foram obtidas, de acordo com o tipo de pleito.

A partir dessas chaves, buscou-se verificar quais os dados são assinados pelo GEDAI-UE utilizando-as para assinar alterações no banco de dados do pleito da eleição. Essas mudanças foram detectadas pela urna eletrônica.

Aprofundando-se a análise constatou-se que a chave decifrada assinava uma série de arquivos a serem copiados para o cartão de memória utilizado para inseminar a urna eletrônica, entre eles:

a. "`scueconf.dat`": arquivo de configuração do programa SCUE (Sistema de Carga da Urna Eletrônica). Este sistema realiza a instalação do cartão de carga para o cartão interno da urna. Não foram feitos mais testes neste arquivo por falta de tempo, mas foram consideradas promissoras as possibilidades na alteração deste arquivo;

b. "`imageinfo`": informações da mídia como número serial e outras;

c. "`avgm.vsc`": arquivo no formato ASN1 contendo as assinaturas dos arquivos "`local.jez`" (informação dos locais de votação) e "`eleitor.jez`" (informações dos eleitores, incluindo impressões digitais);

d. "`<nnnnnnn>-pu.dat`": arquivo com uma máscara do Boletim de Urna (BU), contendo diversos dados que são impressos como: o nome do tribunal, cabeçalhos, entre outros;

e. "`local.jez`": Este arquivo é construído dinamicamente a partir do banco de dados da eleição que é alimentado no momento que os dados dos sistemas de candidaturas e eleitores são inseridos no GEDAI-UE. Dentro deste arquivo "JEZ" (Justiça Eleitoral Zip), existem outros arquivos contendo as descrições das localidades em arquivos com o formato "`<nnnnn>-lo.dat`".

As alterações nos bancos de dados foram todas detectadas pelo GEDAI-UE. Contudo, os autores identificaram, por meio de interceptação das chamadas à função "`CreateFileA`" dentro do *debugger* o momento exato no executável em que o GEDAI-UE gera e assina os arquivos "`<nnnnn>-lo.dat`", que são inseridos no "`local.jez`". Neste ponto, foram realizadas alterações nos arquivos, antes da assinatura. Alguns parâmetros foram modificados, mas novamente a urna detectou as mudanças.

Concluiu-se que a assinatura não seria dos dados do arquivo, mas de um *buffer* de memória que daria origem ao arquivo ("`<nnnnn>-lo.dat`"). Interceptando a chamada a "`WriteFile`" e realizando varredura da memória imediatamente antes da gravação do arquivo, foi possível alterar o conteúdo do *buffer* da seguinte forma:

a. unidade da Federação: foi alterada de "TO" para "PF" (Polícia Federal);

b. nome da Cidade: foi alterada de "PALMAS" para "NATAAL";

c. nome por extenso do Estado: foi alterado de "TOCANTINS" para "PERITOSPF".

As alterações foram realizadas de modo a preservar o tamanho das cadeias de texto e assim evitar eventuais verificações de processamento pelo GEDAI-UE. Após a alteração o cartão foi gerado com sucesso e carregado na Urna, que imprimiu "PF" e "NATAAL" tanto na zerézima quanto no boletim de urna. A urna também exibiu o nome modificado da cidade na sua tela. A Figura 2, veiculada em telejornais com autorização do TSE, permite visualizar uma dessas alterações.

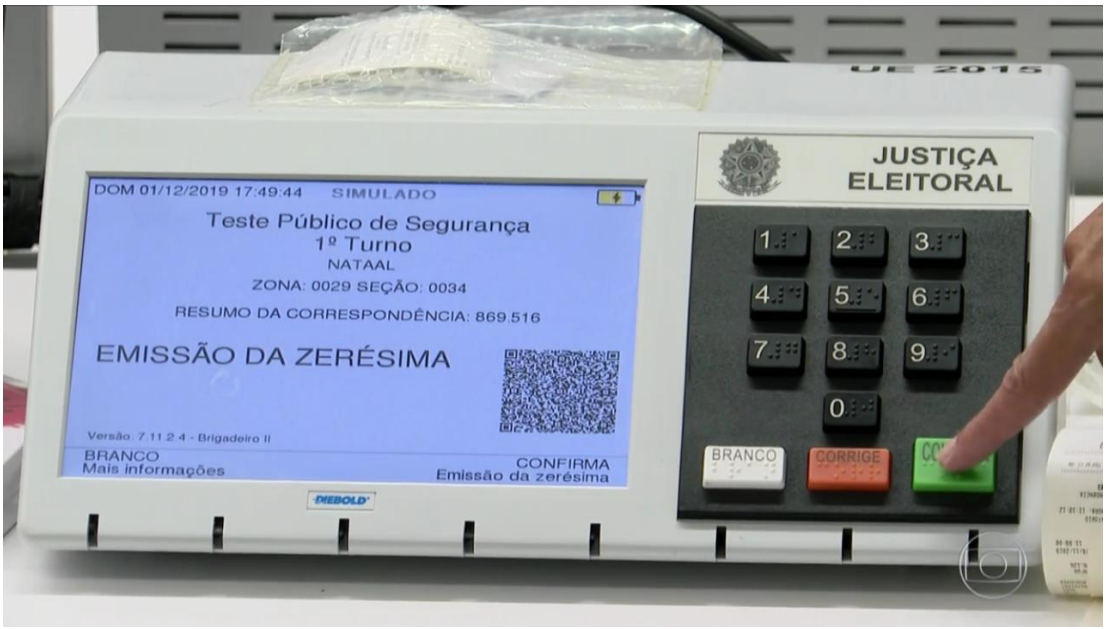

Figura 2. Quadro de reportagem televisiva, com nome da cidade modificado.

3.5. Teste de Confirmação

O teste de confirmação, realizado em 2020 pelo TSE, permitiu aos autores nova execução dos planos de testes que obtiveram sucesso em 2019, a fim de validar as correções efetuadas pelo TSE. Segundo breve apresentação, as seguintes mudanças foram realizadas:

a. SIS: detecção de execução em máquina virtual usando driver de *kernel*;

b. SIS: retirada de criação de *dumps* de memória (hibernação e *crash*);

c. SIS: superação de controles de acesso e melhor validação do SIS pelas aplicações;

d. GEDAI-UE: fim dos arquivos "`<nnnnn>-lo.dat`" e "`audio.jez`"; redução "`scueconf.dat`" e "`infomidia.dat`"; mudança na assinatura dos arquivos "`<nnnnnnn>-pu.dat`", entre outros;

e. GEDAI-UE: detecção de execução em máquina virtual e sob *debugger*;

f. GEDAI-UE: proteção de acesso a chaves com uso do chip TPM.

Durante o teste de confirmação, o SIS voltou a ser comprometido e ter o seu volume cifrado montado com sucesso, usando uma técnica diferente. Desta vez o disco foi inicializado usando o VirtualBox com rotinas que impedem a detecção de execução virtualizadas pelo sistema *guest* (*VirtualBox Hardened*). Para obtenção e montagem do volume cifrado foram realizadas buscas por chaves AES no *dump* de memória, usando a ferramenta "Bulk Extractor". Por meio de combinação de fragmentos de 256 bits, formou-se supostas *master keys* de 512 bits. Estas chaves candidatas foram submetidas à ferramenta MkDecrypt, que montou com sucesso o volume "`SE.pzm`", usando uma delas. A chave do SEVIN não foi decifrada, pois não foi possível subverter a proteção do *chip* TPM. Desta forma o primeiro plano foi novamente concluído com sucesso,

variando a técnica empregada, porém o segundo foi contido com sucesso pelas mudanças realizadas pelo TSE.

4. Discussão

Os autores consideram fundamental a existência dos testes públicos como elemento de transparência do sistema eleitoral. Reconhecem, também, o esforço do TSE na criação de um sistema eleitoral moderno e seguro. A agilidade na apuração e os elementos de segurança inseridos como votação paralela e lacração dos códigos, entre outros, são importantes mecanismos para a confiabilidade do processo eleitoral.

No entanto, tecnicamente, a urna eletrônica tem a segurança fortemente baseada na criptografia e na assinatura digital de dados e informações. Enquanto esses mecanismos conferem segurança na detecção de alterações maliciosas, também significam que o sistema é tão forte quanto a proteção e o gerenciamento das chaves criptográficas utilizadas. Conforme foi descrito neste artigo, o Subsistema de Instalação e Segurança (SIS) foi subvertido expondo chaves privadas relevantes, de modo que os autores puderam alterar informações que foram aceitas e exibidas pela urna eletrônica. Nesse sentido, recomenda-se que o SIS e o próprio sistema operacional Windows sejam substituídos no ambiente de geração dos cartões de memória por um ambiente customizado e mais seguro. As chaves de criptografia, por exemplo, poderiam ser armazenadas usando equipamentos de específicos para armazenamento de chaves como os denominados *Hardware Security Module* (HSM).

Entende-se, igualmente, que é recomendável a utilização de algoritmos públicos e amplamente testados pela comunidade científica. O uso de uma "criptografia de Estado", a que não se tem acesso, representa uma "segurança por obscuridade", prática não recomendada para a proteção de informação [Kerckhoffs 1883]. Acredita-se que a criptografia de Estado pode ser usada para confidencialidade de informações dentro do governo, mas não deveria ser usada na urna eletrônica por falta de transparência e validação do algoritmo.

Por fim, como pode ser visto na Figura 1, a superfície de ataque possível ao sistema eleitoral é muito maior que o disponível para os Testes Públicos de Segurança. Assim, entende-se que os demais sistemas deveriam fazer parte do escopo dos próximos testes públicos, além do próprio TPS adotar um modelo mais simplificado e que estimule uma maior participação. Ademais, o código-fonte e o produto das memórias de resultado (MR) deveriam ser públicos, como já fazem outros países [ACT Electoral Commission 2020]. Essas ações submeteriam o ecossistema do sistema de votação eletrônico no Brasil a maior escrutínio por parte da sociedade, levando a maior transparência e aumentando a qualidade do processo eleitoral.

5. Conclusões

Os resultados dos procedimentos aqui descritos mostram a importância do escrutínio externo do ecossistema de votação eletrônica no Brasil. Espera-se que as sugestões apresentadas sirvam para aumentar a transparência e a segurança do processo.

Na realidade, observou-se que as contribuições já resultaram em melhorias. O TSE implementou mudanças importantes para aumentar a segurança do ambiente e

reduzir a superfície de ataque, especialmente a proteção das chaves de criptografia usando o recurso TPM presente nos computadores modernos. A mudança implementada no SIS para detecção de máquinas virtuais e remoção dos elementos de formação das chaves se mostraram insuficientes para mitigar a vulnerabilidade que permite montar o contêiner criptográfico externamente, visto que um despejo de memória pode revelar a chave mestra e permitir acessar o volume. Importante ressaltar que se trata de um problema de difícil solução, pois programas de cifração automática de arquivos necessitam manter a chave mestra em memória, para viabilizar as operações de leitura ou gravação cifradas.

Referências

ACT Electoral Commission (2020). Development of the system. Disponível em: https://www.elections.act.gov.au/elections_and_voting/electronic_voting_and_counting/development_of_the_system. Acesso em: 29 jun. 2021.

Amato, F., Oro, I. A. B., Chaparro, E., Lerner, S. D., Ortega, A., Rizzo, J., Russ, F., Smaldone, J., and Waisman, N. (2015). Vot.Ar: una mala eleccion. Disponível em: https://github.com/HacKanCuBa/informe-votar/blob/master/Informe/informe.pdf. Acesso em: 08 set. 2021.

Aranha, D. F., Karam M. M., de Miranda A., Scarel F. (2014). (In)segurança do voto eletrônico no Brasil, Cadernos Adenauer 1/2014: Justiça Eleitoral, 117-133.

Aranha, D. F.; Barbosa, P., Cardoso, T. N. C., Luders, C., Matias, P. (2018). Execução de código arbitrário na urna eletrônica brasileira, SBSeg 2018.

Calandrino, J. A., Feldman, A. J., J. A. Halderman, D. W., and H. Yu, W. P. Z. (2007). Source Code Review of the Diebold Voting System. Disponível em: https://jhalderm.com/pub/papers/diebold-ttbr07.pdf. Acesso em: 08 set. 2021.

Coimbra, R. C., Monteiro, J. R. M., Costa, G. S. (2017). Registro impresso do voto, autenticado e com garantia de anonimato, SBSeg 2017.

Filho, A. B., Carvalho, M. A. M., Teixeira, M. C., Simplicio, M. A. Jr., Fernandes, C. T. (2014). Auditoria Especial no Sistema Eleitoral 2014, SBSeg 2015.

G1 (2018). PF investiga suposta invasão hacker a sistemas do TSE. Disponível em: https://g1.globo.com/politica/noticia/2018/11/09/policia-federal-investiga-suposta-invasao-hacker-a-sistemas-do-tse.ghtml. Acesso em: 29 jun. 2021.

Gonggrijp, R., Hengeveld, W. (2007). Studying the Nedap/Groenendaal ES3B Voting Computer: A Computer Security Perspective. In USENIX/ACCURATE Electronic Voting Technology Workshop. Disponível em: http://usenix.org/events/evt07/tech/full_papers/gonggrijp/gonggrijp.pdf. Acesso em: 08 set. 2021.

van de Graaf, J. (2017). O mito da Urna: Desvendando a (in)segurança da urna eletrônica. Disponível em: https://www.urantiagaia.org/social/eleicao/mito-da-urna.pdf. Acesso em: 29 jun. 2021.

Halderman, A, et al. (2009). Lest We Remember: Cold Boot Attacks on Encryption Keys. Communications of the ACM, Volume 52, Issue 5.

Kerckhoffs, A. (1883). La cryptographie militaire. J sci militaires IX:5–38, 161–191. Disponível em: http://www.petitcolas.net/fabien/kerckhoffs. Acesso em: 29 jun. 2021.

Monteiro, J., Lima, S., Rodrigues, R., Alvarez, P., Meneses, M., Mendonça, F., Coimbra, R. (2019). Protegendo o sistema operacional e chaves criptográficas numa urna eletrônica do tipo T-DRE, SBSeg 2019.

SEVIN (2018). Respostas as vulnerabilidades e sugestões de melhorias encontradas no Teste Público de Segurança 2017. Disponível em: http://www.justicaeleitoral.jus.br/arquivos/relatorio-tecnico-tps-2017-1527192798117. Acesso em: 29 jun. 2021.

de Souza, W. S., de Azevedo, D. G., de Souza, S. X. (2017). Do back-end para o front-end: Uma Avaliação de Segurança de Aplicações Móveis da Justiça Eleitoral, SBSeg 2017.

TSE (2019). Você sabe o que é o Teste Público de Segurança do Sistema Eletrônico de Votação? http://www.tse.jus.br/imprensa/noticias-tse/2019/Outubro/voce-sabe-o-que-e-o-teste-publico-de-seguranca-do-sistema-eletronico-de-votacao.

TSE (2021). TSE entra para o seleto grupo de incorporador de funcionalidades no Linux. Disponível em: https://www.tse.jus.br/imprensa/noticias-tse/2021/Maio/tse-entra-no-seleto-grupo-de-incorporador-de-funcionalidades-no-linux. Acesso em: 29 jun. 2021.

Wolchok, S., Wustrow, E., Halderman, J. A., Prasad, H. K., Kankipati, A., Sakhamuri, S. K., Yagati, V., and Gonggrijp, R. (2010). Security analysis of India's electronic voting machines. In ACM Conference on Computer and Communications Security, pages 1–14.