Análise de cerimônias no sistema de votação Helios

Taciane Martimiano[1], **Jean Everson Martina**[1], **M. Maina Olembo**[2]

[1]Universidade Federal de Santa Catarina

[2]Technische Universität Darmstadt

{taciane.m,everson}@inf.ufsc.br, maina.olembo@cased.de

Abstract. *Helios is an online voting system, which allows its voters to verify whether their votes were correctly computed by the booth and stored for final tally. Usability improvements were proposed by Neumann, they are the use of a) several independent verifying institutes and b) smartphone app developed individually by these institutes, for verifying processes and correct storage of the submitted votes on the bulletin board. In this work, these improvements are analyzed as security ceremonies, based on the adaptive threat model proposed by Carlos et al.*
Key words: Ceremony analysis, threat models, usability, Helios

Resumo. *Helios é um sistema de votação online, que permite aos eleitores verificarem se seu voto foi corretamente computado pela cabine de votação e armazenado para contabilização final dos votos. Melhorias visando a usabilidade foram propostas por Neumann, são elas o uso de a) diversos independentes institutos de verificação e b) aplicativo para smartphone desenvolvido individualmente por tais institutos, para processos de verificação e armazenamento correto do voto no quadro de avisos. No presente trabalho, tais melhorias são analisadas como cerimônias de segurança, com base no modelo de ameaça adaptativo proposto por Carlos et al.*
Palavras-chave: Análise de cerimônias, modelos de ameaça, usabilidade, Helios

1. Introdução

Visando a confiança dos eleitores, sistemas criptográficos de votação online que oferecem verificabilidade, enquanto mantém o sigilo do voto, tem sido propostos e continuam sendo aprimorados. Nesse contexto, Helios[Adida 2008][Adida et al. 2009], um sistema de votação baseado na Internet, verificável e de código aberto, tem sido usado principalmente no meio acadêmico.

Levando a usabilidade do sistema em consideração, melhorias foram sugeridas como tentativa de estimular o uso correto e prático do sistema. Para tanto, o eleitor pode usar as páginas web dos institutos confiáveis participantes ou baixar e instalar um aplicativo em seu smartphone[Neumann et al. 2014]. Uma análise da segurança computacional de tais propostas é desenvolvida nesse trabalho, com foco em sigilo e integridade (propriedades importantes para a votação verificável). Para isso, as propostas foram modeladas como cerimônias de segurança, empregando o framework proposto por Carlos et al[Carlos et al. 2013], o qual é baseado no conjunto de capacidades do modelo de ameaça de Dolev-Yao[Dolev and Yao 1983].

O trabalho está estruturado da seguinte forma: o capítulo 2 aborda definições importantes para a compreensão das análises[1] apresentadas posteriormente. Nesse capítulo encontra-se a definição do conceito de cerimônias e seus meios de comunicação, modelos de ameaça, assim como suposições consideradas para realização das análises. O capítulo 3 apresenta a proposta ao sistema Helios versão web com uso de institutos de verificação. A seguir, apresenta a análise dessa proposta e os resultados alcançados. O capítulo 4 apresenta a proposta ao sistema Helios fazendo uso dos aplicativos para smartphone da eleição. A seguir, apresenta a análise dessa proposta, os resultados alcançados e vantagens em se empregar cerimônias. Por fim, o capítulo 5 traz as conclusões e trabalhos futuros.

2. Conceitualização

Nesta seção está definido o conceito de cerimônias e apresentado o modelo de ameaça adaptativo utilizado para análise das cerimônias.

Análise de cerimônias e modelo de ameaça adaptativo

Cerimônias e protocolos de segurança podem ser definidos como uma sequência de interações entre entidades com o intuito de atingir um certo objetivo, como por exemplo autenticação de entidades, distribuição de chaves, etc. A análise de cerimônias estende a análise de protocolos devido à inclusão de nodos humanos ao sistema[Ellison 2007]. Tal inclusão implica em um aumento de complexidade da análise, contudo proporciona resultados mais precisos e completos, sendo possível inclusive detectar falhas de segurança previamente não detectáveis[Carlos et al. 2013].

Nos protocolos, as ações humanas são meramente modeladas como suposições. Quando o protocolo é então implementado, essas suposições resultam em interações de usuário não realísticas (não compatíveis com situações cotidianas do mundo real). Em uma cerimônia de segurança, tem-se o fator humano projetado como parte integrante do sistema. Para isso, as cerimônias apresentam dois canais adicionais ao tradicional canal dispositivo-dispositivo (DD) (proveniente da estrutura dos protocolos). Esses canais adicionais são o canal humano-dispositivo (HD) e o canal humano-humano (HH), empregados para relacionar o 'nodo' humano aos demais nodos do sistema[Carlos et al. 2013]. Contudo, cerimônias de segurança ainda precisam de algumas suposições, tais como conhecimento inicial dos agentes humanos. Essas suposições tendem a ser mais detalhadamente descritas e realísticas em comparação às suposições dos protocolos.

Um dos desafios relativos à análise de cerimônias é a definição do perfil do atacante. O modelo mais conhecido e adotado é o modelo Dolev-Yao. Dolev e Yao[Dolev and Yao 1983] formalizaram o modelo de atacante introduzido por Needham and Schroeder[Needham and Schroeder 1978], onde o atacante tem total controle da rede, sendo capaz de copiar, replicar, alterar e criar mensagens. Ao atacante só não é permitido fazer criptoanálise. Porém, cada cerimônia deve ser analisada individualmente para obtenção do modelo mais adequado e realístico, dados os cenários que envolvem tal cerimônia. Caso as capacidades do atacante sejam exageradas, provavelmente a cerimônia será extremamente complexa e inutilizável. Entretanto, se as capacidades do atacante forem subestimadas, a cerimônia será falha. Para lidar com essa questão, o modelo de ameaça adaptativo proposto por Carlos et al é utilizado na análise das cerimônias apresentadas. Segundo esse modelo adaptativo, a cerimônia pode começar com um modelo de

[1] As análises contidas neste trabalho são de caráter semiformal, inclusive devido à limitação de espaço.

ameaça Dolev-Yao, a partir do qual removem-se capacidades (do conjunto de todas as capacidades que um atacante Dolev-Yao possui) a fim de tornar o atacante mais realístico. Entender o correto modelo de ameaça (ao qual o usuário estará sujeito ao interagir em uma cerimônia) evita sobrecarregar tal usuário com situações hipotéticas e garante que as propriedades de segurança estritamente necessárias sejam empregadas. Para cada canal será definido um modelo de ameaça, de acordo com as capacidades realísticas definidas sobre o conjunto total de capacidades de um atacante Dolev-Yao definidas em Carlos et al[Carlos et al. 2013].

O processo de análise começa com o estabelecimento dos canais presentes na cerimônia. Isso envolve listar os nodos humanos e dispositivos envolvidos, identificar quais desses nodos trocam mensagens entre si, e que tipo de canal essa comunicação caracteriza (isto é, HH, HD ou DD). Assim, é possível analisar o impacto das capacidades de um atacante em cada um dos canais. O atacante objetiva aprender o conhecimento trocado entre nodos. O modelo Dolev-Yao (DY) define habilidades que permitem ao atacante alcançar tal objetivo. Portanto, observa-se em cada mensagem quais abordagens o atacante pode usar para barrar ou modificar mensagens, criar e enviar mensagens de seu próprio conhecimento, etc. de forma a se obter um modelo de ameaça realístico que engloba os perfis dos atacantes associados a cada canal. Por exemplo, se o atacante tem acesso a uma dada chave criptográfica e intercepta mensagens cifradas com essa chave, ele será capaz de decifrar e aprender os conteúdos dessas mensagens, comprometendo a segurança das informações compartilhadas por tal canal[2]. É interessante ressaltar que seguindo o modelo de ameaça adaptativo de Carlos et al, temos um modelo de ameaça realístico e específico para cada cerimônia, dados seus participantes, canais e circunstâncias às quais estará sujeita.

Suposições

O presente trabalho utiliza algumas suposições relacionadas ao Helios e às entidades e análises das cerimônias, citadas a seguir: Considera-se que as entidades presentes nas cerimônias são confiáveis no que diz respeito à integridade do processo sendo executado (suposição já presente no Helios original[Adida 2008]). O atacante está presente nos canais de comunicação, sendo essa uma suposição típica do modelo Dolev-Yao. A cabine de votação do Helios é confiável e, assim, o eleitor tem motivação em usar o sistema para votar e verificar seu voto. Os institutos participantes são confiáveis, uma vez que qualquer comportamento malicioso pode conduzir à perda de reputação. O eleitor é um 'nodo' honesto na cerimônia, pois um eleitor desonesto pode facilmente corromper a conclusão correta da cerimônia em questão [3]. Além disso, o eleitor confia na cabine de votação quanto ao sigilo das informações. A cerimônia é considerada como tendo um único ponto de início e um único ponto de saída. O eleitor deve seguir todos os passos previstos na cerimônia que ele está executando.

3. Aplicação web vinculada a institutos de verificação

Esta seção aborda brevemente os processos que os eleitores desempenhariam utilizando verificação provida pelos institutos de confiança. Através do modelo adaptativo, é

[2]As capacidades do atacante DY aqui referidas encontram-se em [Carlos et al. 2013]

[3]Neste trabalho, não considera-se coerção. Assim, não foram incluídos casos em que o atacante é o próprio eleitor.

possível enfatizar como a presença humana limita as ações de possíveis atacantes do sistema.

1.	V	$\xrightarrow[HD_1]{}$	B	:	URL
2.	B	$\xrightarrow[HD_1]{}$	V	:	Instructions Page
3.	V	$\xrightarrow[HD_1]{}$	B	:	"*Proceed to ballot*"
4.	B	$\xrightarrow[HD_1]{}$	V	:	Ballot Page
5.	V	$\xrightarrow[HD_1]{}$	B	:	Vote
6.	V	$\xrightarrow[HD_1]{}$	B	:	"*Review vote*"
7.	B	$\xrightarrow[HD_1]{}$	V	:	Review Page
8.	V	$\xrightarrow[HD_1]{}$	B	:	"*Encrypt vote*"
9.	B	$\xrightarrow[HD_1]{}$	V	:	$H\{\{Vote + R\}_{Epk}\}$
10.	V	$\xrightarrow[HD_1]{}$	B	:	"*Check encrypted vote*"
11.	B	$\xrightarrow[HD_1]{}$	V	:	Verifying Options Page
12.	V	$\xrightarrow[HD_1]{}$	B	:	I
13.	B	$\xrightarrow[DD_1]{}$	I	:	Vote + R
14.	I	$\xrightarrow[HD_2]{}$	V	:	$H\{\{Vote + R\}_{Epk}\}$ + Vote

Figura 1. Voto de teste

1.	V	$\xrightarrow[HD_1]{}$	B	:	URL
2.	B	$\xrightarrow[HD_1]{}$	V	:	Instructions Page
3.	V	$\xrightarrow[HD_1]{}$	B	:	"*Proceed to ballot*"
4.	B	$\xrightarrow[HD_1]{}$	V	:	Ballot Page
5.	V	$\xrightarrow[HD_1]{}$	B	:	Vote
6.	V	$\xrightarrow[HD_1]{}$	B	:	"*Review vote*"
7.	B	$\xrightarrow[HD_1]{}$	V	:	Review Page
8.	V	$\xrightarrow[HD_1]{}$	B	:	"*Encrypt vote*"
9.	B	$\xrightarrow[HD_1]{}$	V	:	$H\{\{Vote + R\}_{Epk}\}$
10.	V	$\xrightarrow[HD_1]{}$	B	:	"*Log in to cast vote*"
11.	B	$\xrightarrow[HD_1]{}$	V	:	Login Page
12.	V	$\xrightarrow[HD_1]{}$	B	:	VoterID + Password
13.	B	$\xrightarrow[DD_3]{}$	BB	:	$\{Vote\}_{Epk}$
14.	B	$\xrightarrow[HD_1]{}$	V	:	Vote is submitted
15.	V	$\xrightarrow[HD_1]{}$	B	:	I
16.	B	$\xrightarrow[DD_1]{}$	I	:	$H\{\{Vote + R\}_{Epk}\}$
17.	I	$\xrightarrow[HD_2]{}$	V	:	$H\{\{Vote + R\}_{Epk}\}$ + VoterID

Figura 2. Voto final

Nessa proposta[4], o processo de votação é similar ao processo do Helios original. As diferenças surgem nos processos de verificação do voto. Para poder verificar se o voto está corretamente cifrado, o eleitor primeiramente precisa registrar o hash apresentado pela cabine de votação. O eleitor então expressa sua intenção de verificar o voto e seleciona um instituto no qual confia. A cabine de votação transmite as informações necessárias para verificação para o instituto. O instituto, por sua vez, irá computar o hash a partir das informações recebidas da cabine e apresentará o resultado de suas computações ao eleitor, juntamente com o voto recebido. O eleitor agora precisa checar e confirmar se os dois hashes são iguais e se o voto apresentado é de fato correspondente ao candidato escolhido.

Para o eleitor conferir se o seu voto final foi corretamente armazenado no quadro de avisos, ele registra o hash apresentado. O eleitor, então, faz log in para submeter seu voto. Após autenticação bem sucedida, a cabine de votação submete o voto ao quadro de avisos. O eleitor seleciona um instituto dos vários disponíveis, após ter submetido seu voto. Uma nova página web abre, onde o eleitor entra com o hash registrado anteriormente e confere o resultado apresentado pelo instituto. O instituto também precisa apresentar o ID do eleitor, para evitar problemas de colisão de hashes[Kusters et al. 2012]. Assim, mesmo que para dois eleitores seja apresentado o mesmo hash, pelo ID (que é único para cada eleitor) tal problema pode ser identificado.

3.1. Análise

Para o conhecido modelo de ameaça Dolev-Yao (DY), todos os canais de comunicação estão sob um atacante DY. Já no modelo de ameaça adaptativo, considera-se que apenas

[4]As cerimônias que ilustram a proposta estão nas figuras 1 e 2. As mensagens estão em inglês para fazer jus ao sistema real ao qual se referem. As entidades presentes são o eleitor (representado pela letra V, *Voter* em inglês), cabine de votação (letra B, *Booth* em inglês), instituto (letra I, *Institute* em inglês) e quadro de avisos (letras BB, *Bulletin Board* em inglês). As letras abaixo das setas representam o canal pelo qual a mensagem (apresentada ao lado direito de cada imagem) é transmitida.

o canal dispositivo-dispositivo (DD) está sob um atacante Dolev-Yao (DY), enquanto que os canais humano-humano (HH)[5] e humano-dispositivo (HD) estão sob um atacante DY-E. DY-E significa que tal atacante possui todas as capacidades de um atacante DY, exceto a capacidade Escuta (*Eavesdrop*). Essa capacidade é excluída, pois consideram-se ambientes controlados onde o eleitor não precisa checar ao seu redor e assegurar-se de que não há alguém espionando-o. A respeito do canal HD, assume-se que existe um ser humano (e não uma máquina fingindo ser um humano) lidando com um dispositivo (por exemplo, olhando para a tela e digitando algo no teclado). Assim, um atacante DY-E não é capaz de comprometer o sigilo das mensagens enviadas por canais HD uma vez que o eleitor está no domínio do dispositivo, limitando as ações do atacante. Por exemplo, mesmo que o atacante possua as capacidades de Fabricar (*Fabricate*) e Criptografia (*Crypto*), ele só pode aplicar tais capacidades em mensagens e conhecimento que ele já possua. Portanto, não há ameaças nas cerimônias estudadas dado que o atacante não é capaz de aprender nenhuma informação no que diz respeito às ações do eleitor.

Considerando o modelo de ameaça DY, o atacante tem total controle de todos os canais de comunicação e é capaz de manipular o eleitor durante todo o processo. Em tais cenários, o atacante intercepta todas as mensagens trocadas entre os pares de nodos do sistema, e envia mensagens de seu próprio conhecimento no lugar das originais. Simultaneamente, para o eleitor são apresentados dados corretos, onde o atacante se faz passar pelas entidades legítimas. Portanto, o eleitor é levado a acreditar que seu voto foi cifrado, submetido e armazenado apropriadamente, quando isso não é verdade. Contudo, essa situação é altamente improvável de acontecer nas cerimônias apresentadas, pois o canal HD limita as ações do atacante.

Já um cenário realístico e factível é o atacante interceptar mensagens apenas no canal DD. Nesse caso, o instituto recebe informações alteradas, calculando um hash diferente do esperado pelo eleitor. O eleitor passa a não confiar mais no instituto. Esse resultado ressalta a necessidade de se ter vários institutos disponíveis, provendo serviços de verificação para os eleitores. Portanto, o eleitor tem total liberdade de verificar usando vários outros institutos. Se as tentativas seguintes também falharem, então o eleitor pode contatar a comissão eleitoral.

Analisando as cerimônias apresentadas acima, constata-se que a mensagem 13 da cerimônia Voto de teste (figura 1) apresenta Voto + R sendo transmitido como texto plano (sem criptografia) através de um canal DD. O mesmo não acontece com a cerimônia Voto final (figura 2), onde tal mensagem contém informações cifradas com a chave pública da eleição (E_{pk}). Logo, o sigilo não está presente para a cerimônia de voto de teste, estando presente apenas na cerimônia para voto final. Tal conclusão se deve ao fato de que o atacante não aprende o voto ou a informação randômica a partir do hash.

[5]Neste trabalho, nenhuma das cerimônias abordadas utiliza canal humano-humano (HH), assim apenas o canal humano-dispositivo(HD) será abordado.

4. Aplicativo para smartphone

Nessa proposta[6], o eleitor verifica o voto usando um dispositivo diferente do usado para votar, assim não há mais a necessidade de que o eleitor confie no dispositivo que utiliza para votar. Tal dispositivo para verificação é o próprio smartphone do eleitor, que já está em seu domínio e com o qual ele já está acostumado. Esta seção aborda brevemente os processos que os eleitores desempenhariam utilizando verificação provida pelo aplicativo da eleição no smartphone.

1.	V	$\xrightarrow[HD_1]{}$	B	:	URL
2.	B	$\xrightarrow[HD_1]{}$	V	:	Instructions Page
3.	V	$\xrightarrow[HD_1]{}$	B	:	"*Proceed to ballot*"
4.	B	$\xrightarrow[HD_1]{}$	V	:	Ballot Page
5.	V	$\xrightarrow[HD_1]{}$	B	:	Vote
6.	V	$\xrightarrow[HD_1]{}$	B	:	"*Review vote*"
7.	B	$\xrightarrow[HD_1]{}$	V	:	Review Page
8.	V	$\xrightarrow[HD_1]{}$	B	:	"*Encrypt vote*"
9.	B	$\xrightarrow[HD_1]{}$	V	:	Check-code + $QRCode^*$
10.	V	$\xrightarrow[HD_3]{}$	A	:	"*Open/Scan*"
11.	B	$\xrightarrow[DD_2]{}$	A	:	QRCode
12.	V	$\xrightarrow[HD_1]{}$	B	:	"*Check encrypted vote with app*"
13.	B	$\xrightarrow[HD_1]{}$	V	:	QRCode Page
14.	V	$\xrightarrow[HD_3]{}$	A	:	"*Check that vote is correctly encrypted*"
15.	B	$\xrightarrow[DD_2]{}$	A	:	$QRCode^{**}$
16.	A	$\xrightarrow[HD_3]{}$	V	:	"*Check-codes match. Is this your vote?*" + Vote
17.	V	$\xrightarrow[HD_3]{}$	A	:	Confirm (Yes)
18.	V	$\xrightarrow[HD_3]{}$	A	:	"*Close*"

*Check-code is the hash $H\{\{Vote + R\}_{Epk}\}$. In messages 9 and 11, the QRCode has the check-code contents.

**In messages 13 and 15, the QRCode has $(Vote + R + E_{pk})$ information.

Figura 3. Voto de teste

1.	V	$\xrightarrow[HD_1]{}$	B	:	URL
2.	B	$\xrightarrow[HD_1]{}$	V	:	Instructions Page
3.	V	$\xrightarrow[HD_1]{}$	B	:	"*Proceed to ballot*"
4.	B	$\xrightarrow[HD_1]{}$	V	:	Ballot Page
5.	V	$\xrightarrow[HD_1]{}$	B	:	Vote
6.	V	$\xrightarrow[HD_1]{}$	B	:	"*Review vote*"
7.	B	$\xrightarrow[HD_1]{}$	V	:	Review Page
8.	V	$\xrightarrow[HD_1]{}$	B	:	"*Encrypt vote*"
9.	B	$\xrightarrow[HD_1]{}$	V	:	Check-code + $QRCode^*$
10.	V	$\xrightarrow[HD_3]{}$	A	:	"*Open/Scan*"
11.	B	$\xrightarrow[DD_2]{}$	A	:	QRCode
12.	V	$\xrightarrow[HD_1]{}$	B	:	"*Log in to cast vote*"
13.	B	$\xrightarrow[HD_1]{}$	V	:	Login Page
14.	V	$\xrightarrow[HD_1]{}$	B	:	VoterID + Password
15.	B	$\xrightarrow[DD_3]{}$	BB	:	$\{Vote\}_{Epk}$
16.	B	$\xrightarrow[HD_1]{}$	V	:	Vote is submitted
17.	V	$\xrightarrow[HD_3]{}$	A	:	"*Check that vote is on bulletin board*"
18.	A	$\xrightarrow[DD_4]{}$	BB	:	$H\{\{Vote + R\}_{Epk}\}$
19.	BB	$\xrightarrow[DD_4]{}$	A	:	Confirmation + VoterID

*Check-code is the hash $H\{\{Vote + R\}_{Epk}\}$. In messages 9 and 11, the QRCode has the check-code contents.

Figura 4. Voto final

A cabine de votação apresenta um QR code contendo o hash do voto, juntamente com o próprio hash. O eleitor utiliza seu smartphone para escanear o QR code que contém esse hash. Tal hash será armazenado pelo aplicativo para uso posterior no processo de verificação do voto. O eleitor expressa sua intenção de verificar o voto para a cabine de votação. A seguir, ele escaneia um segundo QR code e o aplicativo computa o hash e o compara com o armazenado anteriormente. Em um caso de comparação bem sucedida, o aplicativo informa que os hashes são iguais e pede ao eleitor para confirmar que o voto apresentado na tela é o voto correto.

Para verificar que o voto foi corretamente armazenado para contagem final no quadro de avisos (cerimônia apresentada na figura 4), o eleitor escaneia o primeiro QR code que contém o hash. O eleitor faz log in e a cabine de votação submete seu voto após autenticação bem sucedida. O eleitor utiliza o aplicativo para checar o quadro de avisos procurando pelo hash do seu voto. O aplicativo realiza essa checagem consultando o quadro de avisos pelo valor do hash. Em caso bem sucedido, o aplicativo mostra uma mensagem para o eleitor afirmando que o hash foi armazenado no quadro de avisos. Para

[6]As cerimônias que ilustram a proposta estão nas figuras 3 e 4. As mensagens estão em inglês para fazer jus ao sistema real ao qual se referem. Além das entidades que já apareceram nas cerimônias anteriores, tem-se a presença da entidade aplicativo (representado pela letra A, *App* em inglês).

prevenir problemas de colisão [Kusters et al. 2012], o aplicativo também retorna o ID do eleitor. Em caso mal sucedido, o aplicativo informa o eleitor de que o hash não foi encontrado no quadro de avisos. O eleitor pode usar outros aplicativos para verificação. Em caso de múltiplos hashes falharem, o eleitor pode entrar em contato com a comissão eleitoral.

4.1. Análise

Apesar de os canais DD geralmente estarem sob um atacante DY, isso não é realístico para o canal DD_2 (por exemplo, na mensagem 11 da figura 3). Esse canal é um 'canal visual' uma vez que não há bluetooth ou conexão de qualquer modo entre os dispositivos envolvidos. Nesse específico cenário, o eleitor usa seu smartphone para escanear o QR code apresentado pelo computador. Considera-se que ambos os dispositivos estão no domínio do eleitor e não sob controle do atacante. Para o canal DD_2, tem-se situações similares aos canais HD (descritas na seção 3). Por exemplo, o atacante não pode bloquear os conteúdos passando por esse canal pois isso implicaria o atacante bloquear a tela do computador do eleitor e seu smartphone. Situações similares acontecem se o atacante tenta aplicar qualquer outra de suas capacidades. Assim, para um ataque ser bem sucedido, o atacante precisa estar no domínio dos dispositivos do eleitor. Tal cenário só seria factível se o eleitor deixasse os dispositivos abandonados no meio do processo de votação. Portanto, considera-se que o canal DD_2 também é DY-E, assim como os canais HD. Qualquer combinação enfraquecida de capacidades do atacante DY (qualquer combinação de capacidades, não envolvendo Escuta) continuará não sendo efetiva em tais canais. Isso se deve ao fato de que Escuta é a única capacidade que pode comprometer o sigilo do voto do eleitor.

No que diz respeito ao modelo de ameaça DY, o atacante pode manipular o eleitor através da manipulação das informações apresentadas a ele. Tal situação pode ser considerada realística para a cerimônia do voto final usando aplicativo (figura 4). Contudo, é altamente improvável de acontecer devido ao fato dos canais HD estarem seguros sob a suposição de que o ambiente é controlado. Adicionalmente, foi demonstrado ser irrealista para a cerimônia de voto de teste usando aplicativo (figura 3). Tal cerimônia é mais segura por possuir o canal visual, o qual limita as ações do atacante, pois apresenta o mesmo comportamento que os canais HD. Uma contribuição muito importante da proposta usando o aplicativo constitui-se de que ambos os votos de teste e final são secretos, quando comparados com a proposta que faz uso dos institutos (onde o voto de teste é enviado sem uso de criptografia por um canal DY). Tal contribuição significa que essa cerimônia possui a propriedade do sigilo e, como as mensagens não são interrompidas e não são modificadas, conclui-se que tal cerimônia também garante integridade.

5. Conclusão

Neste trabalho foi analisada a segurança das propostas feitas para o sistema de votação Helios. Para tal análise foi utilizado o framework proposto por Carlos et al[Carlos et al. 2013], aplicando o modelo adaptativo de ameaça. Para esse fim, os processos de votação e verificação do voto foram abordados como cerimônias, integrando a interação humana na análise. O modelo de ameça Dolev-Yao foi usado para comparação com o modelo adaptativo, onde foi possível ressaltar os ganhos em se empregar um modelo que reflita as necessidades de segurança para cada específico cenário sem sobrecarregar o usuário.

Na primeira proposta para verificação, usando institutos confiáveis, os resultados mostram a possibilidade de violações de sigilo quando o eleitor verifica se seu voto está corretamente cifrado, e violações de integridade quando ele verifica se seu voto foi corretamente submetido no quadro de avisos. As violações de integridade tomam forma de 'ataques de reputação', resultando na perda de confiança no instituto por parte do eleitor (ao receber informações incorretas). Para as cerimônias envolvendo o aplicativo, o sigilo é mantido devido à presença de um canal visual e ao fato da informação ser enviada cifrada (e não em forma de texto plano). Os resultados também mostram que nenhum ataque significativo pode ocorrer quando o eleitor verifica se seu voto está cifrado de forma correta. Violações de integridade acontecem através dos 'ataques de reputação', cuja estratégia de mitigação é a existência de diversos institutos, ou aplicativos mantidos por esses, disponíveis para o eleitor. Através dessa solução, a propriedade da integridade pode ser mantida em ambas as propostas abordadas. No caso do processo de verificação falhar em algum dos casos, o eleitor pode verificar fazendo uso de outras fontes.

Os resultados desse trabalho ressaltaram várias melhorias que podem ser feitas ao protocolo de votação do Helios. Uma futura proposta envolve o eleitor entrar com uma informação única conhecida apenas por ele. Verificar a presença dessa informação em um estágio posterior do sistema garantiria ao eleitor a integridade do voto submetido. Propostas serão desenvolvidas com o objetivo de equilibrar os aspectos de segurança e as expectativas e habilidades dos nodos humanos na cerimônia.

Referências

Adida, B. (2008). Helios: Web-based Open-Audit Voting. In *Proceedings of the 17th Symposium on Security*, pages 335 – 348. Usenix Association.

Adida, B., De Marneffe, O., Pereira, O., and Quisquater, J.-J. (2009). Electing A University President using Open-Audit Voting: Analysis of Real-World Use of Helios. In *Proceedings of the 2009 Conference on Electronic Voting Technology/Workshop on Trustworthy Elections*, pages 10–10. Usenix Association.

Carlos, M. C., Martina, J., Price, G., and Custodio, R. F. (2013). An Updated Threat Model for Security Ceremonies. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, SAC '13, pages 1836–1843, New York, NY, USA. ACM.

Dolev, D. and Yao, A. C. (1983). On the Security of Public Key Protocols. *IEEE Transactions on Information Theory*, 29(2):198–208.

Ellison, C. (2007). Ceremony Design and Analysis. Cryptology ePrint Archive, Report 2007/399. `http://eprint.iacr.org/`.

Kusters, R., Truderung, T., and Vogt, A. (2012). Clash Attacks on the Verifiability of E-Voting Systems. In *Security and Privacy (SP), 2012 IEEE Symposium on*, pages 395–409.

Needham, R. M. and Schroeder, M. D. (1978). Using encryption for authentication in large networks of computers. *Commun. ACM*, 21(12):993–999.

Neumann, S., Olembo, M. M., Renaud, K., and Volkamer, M. (2014). Helios verification: To alleviate, or to nominate: Is that the question, or shall we have both? In *3rd International Conference on Electronic Government and the Information Systems Perspective*. Springer. To appear.

