Anais do Conic-Semesp. Volume 1, 2013 - Faculdade Anhanguera de Campinas - Unidade 3. ISSN 2357-8904

TÍTULO: PROTÓTIPO DE VOTAÇÃO CIVIS: AVALIAÇÃO DAS VULNERABILIDADES WEB

CATEGORIA: EM ANDAMENTO

ÁREA: ENGENHARIAS E TECNOLOGIAS

SUBÁREA: COMPUTAÇÃO E INFORMÁTICA

INSTITUIÇÃO: UNIVERSIDADE FEDERAL DO PARÁ

AUTOR(ES): CARLOS GUSTAVO RESQUE DOS SANTOS

ORIENTADOR(ES): ROBERTO SAMARONE DOS SANTOS ARAÚJO

COLABORADOR(ES): LEONARDO BARBOSA DA COSTA

1. RESUMO

Este estudo pretende avaliar a segurança do protótipo de votação CIVIS em relação às vulnerabilidades web conhecidas e documentadas seguindo um guia de revisão de código amplamente reconhecido o OWASP *Code Review Guide* [1]. Inicialmente foram identificadas todas as portas de entrada da aplicação, os níveis de acesso ao sistema e as áreas/itens de interesse. Foram desenvolvidos diagramas DFDs (*Data Flow Diagram*) para visualização do fluxo de dados. Futuramente será realizada a revisão de código com a finalidade de identificar e avaliar as vulnerabilidades web da aplicação e propor contramedidas.

Palavras-chave: Revisão Segura de Código Fonte, Votação Online Segura.

2. INTRODUÇÃO

Sistemas de votação frequentemente necessitam de requisitos de segurança, especialmente de confidencialidade e integridade, que além de serem chaves para a votação, também são requisitos conflitantes [2]. Quando o sistema de votação se trata de uma aplicação web o requisito de autenticidade eleva seu risco, devido à Internet não ser um meio de transmissão seguro, embora medidas possam mitigar tal problema.

CIVIS é um protótipo de votação seguro desenvolvido de acordo com o protocolo proposto por Araújo *et al*, 2010 [3]. Esse protocolo foi publicado em âmbito internacional e garante matematicamente que o voto será computado de forma correta e sigilosa. Porém, é necessário avaliar a segurança do protótipo em relação às vulnerabilidades web que podem existir pela má codificação ou configuração do protótipo.

3. OBJETIVOS

Geral: avaliar a segurança do protótipo de votação CIVIS de acordo com o guia de revisão de código OWASP *Code Review Guide* [1].

Específicos: desenvolver a modelagem de ameaças da aplicação (decompor e modelar a aplicação, determinar e ordenar ameaças); realizar a revisão de código, de acordo com as vulnerabilidades web encontradas.

4. METODOLOGIA

4.1. Tecnologias e Configurações da Aplicação

As principais tecnologias utilizadas para o desenvolvimento da aplicação foram: no servidor a linguagem Java em conjunto com o Framework JSF 2.0; para o proces-

samento no lado cliente principalmente Applets em Java e a linguagem de programação javascript com a utilização de AJAX.

4.2. Guia de Revisão de Código

Será utilizado o guia de revisão de código produzido pela organização OWASP para realizar a avaliação da segurança web do CIVIS. Esse guia divide a revisão em duas etapas distintas: modelagem de ameaças da aplicação e revisão de código. Esse guia está baseado no livro *Writing Secure Code* [4] que também será utilizado como apoio técnico.

5. DESENVOLVIMENTO

5.1. Modelagem de Ameaças da Aplicação

O processo de modelagem de ameaças está dividido em três etapas: decompor a aplicação; determinar e ordenar ameaças; e determinar contramedidas e mitigações.

Decompor a aplicação consiste em criar casos de usos para entender como a aplicação interage com entidades externas; identificar pontos de entrada para saber como um agente malicioso pode interagir com a aplicação; identificar itens/áreas de interesse de um agente malicioso; e identificar os níveis de acesso à aplicação.

Determinar e ordenar ameaças consiste em identificar as ameaças presentes na aplicação usando os pontos de entrada e itens/áreas de interesse identificados na decomposição da aplicação; classificar as ameaças de acordo com a metodologia STRIDE; e calcular qual o impacto delas na aplicação, com a metodologia DREAD, possibilitando uma ordenação das piores ameaças.

Por fim, identificar contramedidas e mitigações consiste em listar quais são as possíveis alternativas para as ameaças detectadas de acordo com sua categoria. Essas alternativas devem levar em consideração o impacto que a ameaça pode trazer em nível organizacional e técnico e qual a abrangência desse impacto.

5.2. Revisão de Código

A revisão de código está dividida em duas partes: revisão por controle técnico e revisão de código. A primeira visa verificar as configurações do ambiente da aplicação em busca de vulnerabilidades relacionadas à autenticação, autorização, gerenciamento de sessão, validação de entrada e tratamento de erro. A revisão por código visa verificar as boas práticas de programação em busca de vulnerabilidades

relacionadas à *Buffer Overflows*, *OS Injection*, SQL Injection, Validação de dados, XSS, CSRF, problemas de Log, integridade da sessão e concorrência de *Threads*;

6. RESULTADOS PRELIMINARES

O projeto está em etapas iniciais e em pleno desenvolvimento. Como resultado preliminar foi concluído a primeira etapa da modelagem de ameaças da aplicação, ou seja, foram identificados e documentados as portas de entrada, os itens/áreas de interesse e os níveis de acesso à aplicação. Foram construídos DFDs (*Data Flow Diagrams*) para desmembrar a aplicação e permitir visualização do fluxo dos dados na aplicação. A **figura 1** mostra o DFD geral da aplicação. Os DFDs específicos já foram desenvolvidos para cada item/área de interesse identificado e eles abrangem também as portas de entrada da aplicação e os níveis de acesso.

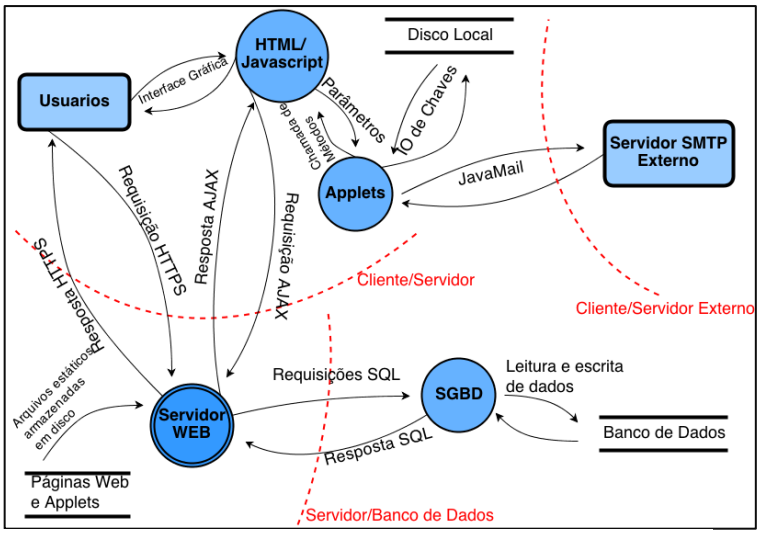

Figura 1 – DFD geral do protótipo de votação CIVIS.

Como trabalhos futuros serão utilizados os DFDs para identificação das possíveis ameaças web da aplicação; categorização das ameaças de acordo com o STRIDE; cálculo do impacto de cada aplicação de acordo com o DREAD; e identificação de contramedidas para as ameaças.

7. FONTES CONSULTADAS

[1] WILLIAMS, J. **OWASP Code Review Guide V1.1**. OWASP Foundation 2008.

[2] CLARKSON, M. R.; CHONG, S.; MYERS, A. **Civitas: Toward a Secure Voting System**. Symposium on Security and Privacy, IEEE Computer Society, 2008.

[3] ARAÚJO, R.; RAJEB, N.; ROBBANA, R.; TRAORÉ, J.; YOUSSFI, S. **Towards Practical and Secure Coercion-Resistant Electronic Elections**. CANS, páginas 278-297. http://dx.doi.org/10.1007/978-3-642-17619-7_20. 2010.

[4] HOWARD, M.; LeBlanc, D. **Writing Secure Code**. Microsoft Press 2nd ed. 2003.