Um Processo Seguro para Desenvolvimento de Software

Francisco José Barreto Nunes, Arnaldo Dias Belchior

Universidade de Fortaleza (UNIFOR) – Mestrado em Informática Aplicada

fcojbn@yahoo.com.br, belchior@unifor.br

Abstract. *Information Security aims to guarantee information integrity, availability and confidentiality. This work identifies a set of activities, derived from the information security, which can be aggregated in a software process; contributing to have a development secure process.*

Resumo. *A Segurança da Informação objetiva garantir que a informação seja integra, disponível e confidencial. Este trabalho identifica um conjunto de atividades derivadas da segurança da informação, que podem ser agregadas ao processo de software, auxiliando na condição de que se tenha um processo seguro de desenvolvimento.*

1. Introdução

Os processos de desenvolvimento de software em geral não garantem que os sistemas sejam imunes a ataques ou deixem de apresentar problemas de segurança. CLASP (*Comprehensive, Lightweight Application Security Process*) é uma iniciativa que objetiva aplicar a segurança dentro do processo de desenvolvimento de software (CLASP, 2006). Dias (2001) afirma que a segurança da informação envolve a manutenção da confidencialidade, a disponibilidade e a integridade de informações, variando de acordo com o tipo de ambiente computacional e com o grau de importância do sistema.

Este trabalho propõe um conjunto de atividades de segurança da informação, que compõem um processo seguro de desenvolvimento de software, baseadas no SSE-CMM (2003), no OCTAVE (Alberts, 2001), na ISO/IEC 15408 (2005a, 2005b, 2005c) e na ISO/IEC 17799 (2005). Essas atividades foram validadas através de uma pesquisa de campo realizada com especialistas em segurança da informação e em processo de desenvolvimento de software.

Este trabalho está organizado como se segue: a seção 2 descreve padrões e normas de segurança que estruturam atividades do processo seguro proposto; a seção 3 apresenta as atividades do processo seguro de desenvolvimento de software; a seção 4 analisa os resultados obtidos em uma pesquisa de campo; a seção 5 apresenta as principais conclusões deste trabalho.

2. Padrões e Normas de Segurança da Informação

O conjunto de atividades de segurança da informação do processo seguro de desenvolvimento de software proposto foi baseado em padrões e normas de segurança a seguir.

O SSE-CMM (*System Security Engineering – Capability Maturity Model*) é usado em um contexto de se aumentar garantias na segurança de um sistema através do fornecimento de um meio de traduzir as necessidades de segurança do cliente em um processo de segurança, que gere produtos que satisfaçam tais necessidades. Neste trabalho, somente foi considerada a dimensão "domínio" relacionada com a engenharia de segurança, que contém um conjunto de boas práticas de segurança, que podem ser usadas no ciclo de vida do software (SSE-CMM, 2003).

O OCTAVE (*Operationally Critical Threat, Asset, and Vulnerability Evaluation*) é uma técnica estratégica de planejamento e avaliação baseada em riscos para segurança. Os riscos dos ativos mais críticos são usados para priorizar áreas que necessitam de melhoria e para definir a estratégia da organização (Alberts, 2001)

Para a ISO/IEC 15408 (2005a, 2005b, 2005c), o desenvolvimento seguro de software envolve segurança tanto do ambiente de desenvolvimento quanto da aplicação desenvolvida. As necessidades de segurança devem ser tratadas em todo o ciclo de vida, passando pela gerência de requisitos de segurança, especificação funcional, projeto de alto nível, projeto de baixo nível, até a implementação final do sistema em seu ambiente de produção.

A NBR ISO/IEC 17799 (2005) objetiva preservar a confidencialidade, integridade e disponibilidade das informações, através da implementação de controles, através da implementação de políticas, práticas ou processos. Esses controles garantem que os objetivos estabelecidos para a segurança serão atendidos satisfatoriamente. O essencial para uma organização é identificar os requisitos de segurança.

3. O Processo Seguro para Desenvolvimento de Software

A partir dos padrões e das normas da seção anterior, foi proposto o conjunto de atividades do processo seguro para desenvolvimento de software (Tabela 1).

Tabela 1. Atividades do Processo Seguro de Desenvolvimento

MACRO-ATIVIDADE	ATIVIDADE
Planejar Segurança	Definir objetivos de planejamento de segurança e identificar seus mecanismos.
	Atribuir responsabilidades de segurança no projeto.
	Implementar ambientes de processamento.
	Planejar o gerenciamento de incidentes de segurança.
Avaliar Vulnerabilidade de Segurança	Executar métodos de identificação de vulnerabilidade de segurança.
	Analisar as vulnerabilidades de segurança identificadas.
Modelar Ameaça de Segurança	Identificar as ameaças de segurança aos ativos críticos.
	Classificar as ameaças de segurança aos ativos.
	Desenvolver estratégias de redução das ameaças de segurança.
Avaliar Impacto de Segurança	Priorizar processos críticos influenciados pelo sistema.
	Revisar ativos do sistema que se referem à segurança.
	Identificar e descrever impactos de segurança.
Avaliar Risco de Segurança	Identificar exposição de segurança.
	Avaliar risco de exposição de segurança.
	Priorizar riscos de segurança.
Especificar Necessidades de Segurança	Compreender as necessidades de segurança do cliente.
	Capturar uma visão de alto nível orientada à segurança da operação do sistema.
	Definir requisitos de segurança.
	Obter acordo sobre requisitos de segurança.
Fornecer Informação de Segurança	Entender e revisar necessidades de informação de segurança.
	Determinar considerações e restrições de segurança.
	Identificar e analisar alternativas de segurança.
	Fornecer orientação de segurança.
	Identificar e revisar requisitos de garantia de segurança.
Verificar e Validar Segurança	Definir a abordagem de verificação e validação de segurança.
	Realizar verificação de segurança.
	Realizar validação de segurança.
	Revisar e comunicar resultados de verificação e validação de segurança.
Gerenciar Segurança	Gerenciar e controlar serviços e componentes operacionais de segurança.
	Gerenciar percepção, treinamento e programa de educação de segurança.
	Gerenciar a implementação de controles de segurança.
Monitorar Comportamento de Segurança	Analisar registro de evento com impacto na segurança.
	Preparar a resposta aos incidentes de segurança relevantes.
	Monitorar mudanças em ameaças, vulnerabilidades, impactos, riscos, no ambiente.
	Reavaliar mudanças em ameaças, vulnerabilidades, impactos, riscos e no ambiente.
	Revisar o comportamento de seg. do sistema para identificar mudanças
	Realizar auditorias de segurança.
Garantir Segurança	Definir estratégia de manutenção da garantia de segurança.
	Conduzir análise de impacto de segurança das mudanças.
	Controlar as evidências da manutenção da garantia de segurança.

A partir das atividades do processo seguro de desenvolvimento de software proposto, foi realizada uma pesquisa de campo aplicada a especialistas em segurança da informação e especialistas em processo de desenvolvimento de software, que avaliaram o grau de importância das atividades desse processo seguro, a partir da escala abaixo:

- **0:** a atividade não é importante para o processo seguro.
- **1:** a atividade raramente é importante para o processo seguro.
- **2:** a atividade é importante para o processo seguro.
- **3:** a atividade é muito importante para o processo seguro.
- **4:** a atividade é extremamente importante para o processo seguro.

Este questionário foi aplicado a quarenta e dois especialistas em vários estados brasileiros. Foram descartados apenas três desses questionários por estarem preenchidos de forma incompleta. A seguir, serão apresentados os principais resultados desta pesquisa de campo.

4. Avaliação dos Resultados

A Figura 1 mostra, em ordem decrescente do grau de importância, as sete atividades melhores avaliadas pelos especialistas desta pesquisa do processo seguro proposto.

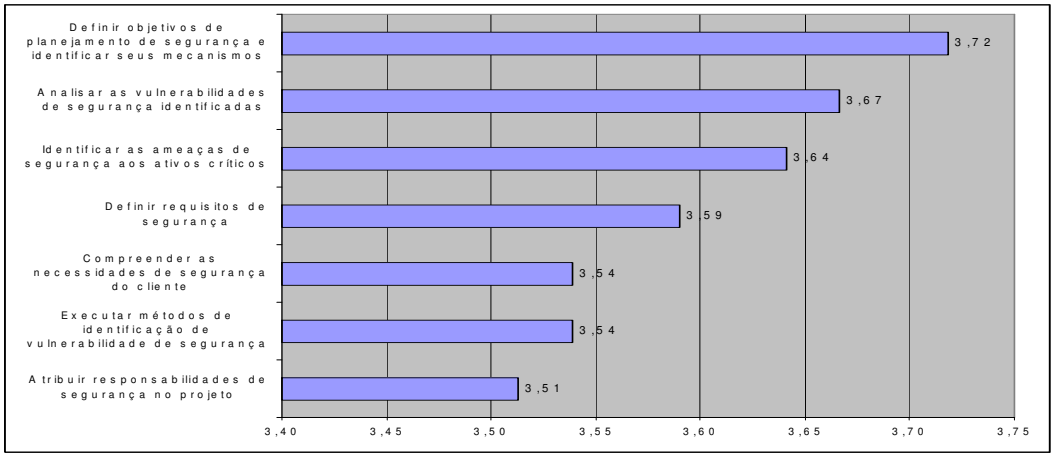

Figura 1. As 7 atividades com maiores graus de importância

A atividade "Definir objetivos de planejamento de segurança e identificar seus mecanismos" foi a melhor avaliada com o valor de 3,72. Isto evidencia a relevância do planejamento da segurança, através da definição de objetivos de segurança e da identificação de mecanismos necessários e suficientes para o gerenciamento do projeto.

A atividade "Analisar as vulnerabilidades de segurança identificadas" foi a segunda melhor avaliada com o valor de 3,67. Neste contexto, é altamente recomendável que seja realizada uma avaliação, que determine se as vulnerabilidades de segurança identificadas possam levar à violação de funcionalidades de segurança por seus usuários.

A atividade "Identificar as ameaças de segurança aos ativos críticos" foi a terceira melhor avaliada com o valor de 3,64. Isto reforça a importância de que as ameaças de segurança de cada ativo crítico do sistema devam ser identificadas, através de um mapeamento das áreas de perigo para cada ativo crítico segundo o perfil de ameaça daquele ativo.

A atividade "Definir requisitos de segurança" foi a quarta melhor avaliada com o valor de 3,59. Isto ressalta a importância de se definir um conjunto consistente de requisitos, que estabeleçam o nível de segurança a ser implementado em um sistema.

As atividades "Compreender as necessidades de segurança do cliente" e "Executar métodos de identificação de vulnerabilidade de segurança" foram bem

pontuadas e obtiveram um valor de 3,54. Isto mostra que todas as informações necessárias para um entendimento das necessidades de segurança do cliente devem ser coletadas. Igualmente, métodos, técnicas, e critérios devem ser executados para identificar e caracterizar as vulnerabilidades de segurança do sistema. A Figura 2 apresenta as três atividades que obtiveram os menores graus de importância, segundo os especialistas desta pesquisa.

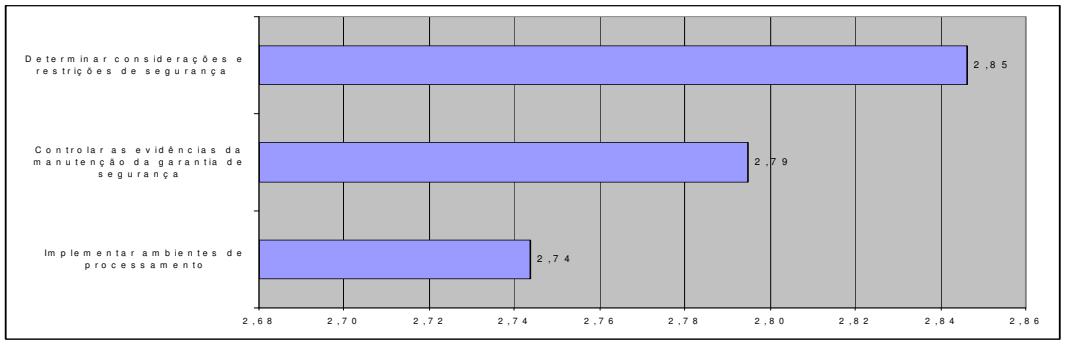

Figura 2. As 3 atividades com menores graus de importância

A atividade "Implementar ambientes de processamento" foi a menos pontuada no processo seguro de software, com o valor de 2,74. Isto significa os especialistas deram pouca relevância para a implementação de ambientes de ensaio e ambientes de produção de segurança e a avaliação do nível de separação necessário entre eles para prevenir problemas operacionais.

As atividades "Controlar as evidências da manutenção da garantia de segurança" e "Determinar considerações e restrições de segurança" também foram consideradas de menor relevância. Assim, é considerada de média relevância manter a garantia de segurança através de evidências que ratifiquem a segurança do sistema. Isto pode auxiliar na tomada de decisões do processo e na realização de escolhas conscientes de itens de segurança da informação.

5. Conclusão

Este trabalho apresentou um conjunto de atividades que compõem o processo seguro de desenvolvimento de software, proposto a partir de padrões e normas modelos de grande relevância. Uma pesquisa de campo foi realizada com um número significativo de especialistas em vários estados brasileiros para validar esse processo. Os resultados da pesquisa demonstram que mesmo havendo atividades com menor pontuação, estas ainda têm importância significativa para o processo seguro.

Referências

ALBERTS, C. *et al*. (2001) "*OCTAVE - The Operationally Critical Threat, Asset, and Vulnerability Evaluation*", Carnegie Mellon – Software Engineering Institute, www.cert.org/octave.

CLASP (2006) *Comprehensive, Lightweight Application Security Process*, Version 1.2. www.securesoftware.com/process/clasp

DIAS, C. (2001), Segurança e Auditoria da Tecnologia da Informação, AXCEL BOOKS.

ISO/IEC 15408-1. (2005a) *Information Technology – Security Techniques – Evaluation Criteria for IT Security – Part 1: Introduction and General Model*.

ISO/IEC 15408-2. (2005b) *Information technology – Security techniques – Evaluation criteria for IT security – Part 2: Security functional requirements.*

ISO/IEC 15408-3. (2005c) *Information technology – Security techniques – Evaluation criteria for IT security – Part 3: Security assurance requirements.*

ISO/IEC 17799. (2005) *Tecnologia da informação – Técnicas de segurança - Código de prática para a gestão da segurança da informação, ABNT, Rio de Janeiro.*

SSE-CMM. (2003) *System Security Engineering – Capability Maturity Model*, Version 3, www.sse-cmm.org.