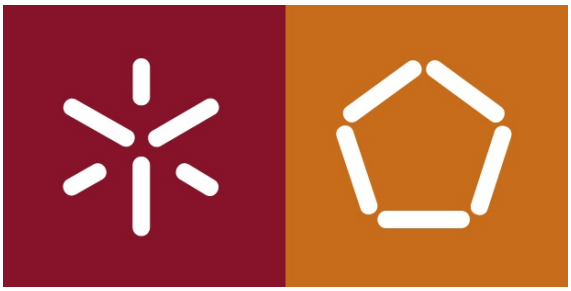

Universidade do Minho

Mestrado em Engenharia Informática

Criptografia e Segurança da Informação

Engenharia de Segurança

Desenvolvimento de Software Seguro - SAFECode

André Gonçalves - A80368
Nelson Sousa - A82053
Pedro Freitas - A80975

23 de Março de 2020

Resumo

No âmbito da Unidade Curricular de Engenharia de Segurança, enquadrada no perfil de especialização de Criptografia e Segurança da Informação, este projeto foi desenvolvido com o objetivo de abordar e explorar conceitos úteis e relevantes desta UC, sendo que se pretende adquirir conhecimentos e consciência da importância de Desenvolvimento de Software Seguro e a situação atual no que toca a esse aspeto. Durante este relatório vamos abordar um tópico em especial: SAFECode. SAFECode é uma organização cujo principal objetivo é garantir a segurança de uma aplicação de forma a satisfazer os requisitos do cliente como os direitos dos utilizadores finais.

Conteúdo

1 Introdução

A crescente dependência da nossa sociedade em software torna imperativa a necessidade de existir uma efetiva "software assurance". Contudo, para que esta questão de "software assurance"seja corretamente abordada necessitamos, no mínimo, de um entendimento do que fazer, como lidar com isso ou o porquê da sua necessidade.

Software assurance engloba o desenvolvimento e implementação de métodos e procedimentos de forma a assegurar que o software funciona como o esperado enquanto atenua os riscos de vulnerabilidades e da existência de código malicioso que poderia prejudicar o utilizador final.

Todos sabemos bastante bem que as organizações têm se preocupado cada vez mais com segurança e como minimizar os riscos de ataques, que de dia para dia, são cada vez mais sofisticados. Ao adquirir software, os clientes preocupam-se com a introdução de novas vulnerabilidades nos seus ambientes de TI que podem comprometer os dados do cliente, interromper os serviços e afetar a confiança. Há já algum tempo que ouvimos dizer que as empresas expressam frustração em relação à falta de um método amplamente aceitável, repetível e escalável, para avaliar a segurança do software adquirido. Embora muitas iniciativas tenham sido criadas para tentar resolver esse problema, sabemos que não existe uma solução única e universal que assegure uma melhor garantia de software. Aliás, para além de haver avaliações de segurança mal elaboradas existem também inúmeros clientes que baseiam as suas tomadas de decisão em torno de uma avaliação incompleta ou enganosa.

Ao longo do presente relatório iremos apresentar o SAFECode, que é uma framework que permite examinar a segurança do processo de desenvolvimento dos fornecedores de software.

2 Explicação do Problema

Todos os clientes têm preocupações acerca de software assurance, e todos querem ter a certeza de que o software que compram é seguro e confiável. Isso só é alcançado quando o software é criado e sustentado usando as práticas recomendadas de forma a ter um lifecycle seguro no desenvolvimento de software.

A tabela a seguir destaca os principais problemas tanto para o utilizador como para o fornecedor de não haver um método comum de avaliação:

Customer Concerns	Supplier Concerns
• No single, consistent way to achieve clear, testable, repeatable ways to build and maintain a fact-based trust between suppliers and customers • General lack of awareness within many enterprises of what to look for when evaluating software • Inadequate insight into what security due diligence has been performed on the components included in software • Need to understand whether a company has a secure development process and whether that process was applied to the specific product being purchased	• No scalable way to provide multiple customers with the information they require to make purchase decisions • Clearing multiple, often diverse, customer hurdles is costly and diverts resources from critical engineering tasks – a problem more acute for small and mid-sized vendors • No current agreement on what information customers should be requesting; some requests do not align well with real-world secure development practices

Figura 1: Principais preocupações dos clientes e dos fornecedores

É aqui que entra a SAFECode. SAFECode constatou que as empresas precisavam de um recurso de estrutura avaliativa para ajudá-las a selecionar e comprar produtos de tecnologia mais seguros e permitir que avaliassem melhor os riscos associados aos seus fornecedores de tecnologia. Após um ano de pesquisa e de análise da informação recolhida foi então criada uma framework de forma a ajudar os clientes a escolherem o método de avaliação que mais se lhes adequa.

3 A SAFECode

SAFECode tem uma abordagem diferente de outras iniciativas porque foi criado como uma framework (não uma checklist) para pensar em preocupações de garantia de segurança com uma avaliação baseada em processos para a adoção de estratégias de melhores práticas. Foi elaborada segundo a máxima que a segurança do software advém de um processo de desenvolvimento seguro.

Segundo a SAFECode uma avaliação de garantia de software deve-se concentrar principalmente no processo seguro de desenvolvimento de software e na sua aplicação ao produto que está a ser avaliado, levando em consideração o contexto do ambiente operacional pretendido do produto. Portanto, a framework apresentada pela SAFECode fornece uma estrutura para examinar o processo de desenvolvimento que beneficia tanto clientes como fornecedores. Esta identifica aquilo que os fornecedores podem ou não afirmar e explica o porquê de seguir um processo ser mais importante do que uma checklist, que às vezes pode ser prejudicial.

As organizações podem avaliar a maturidade do processo de segurança de software de um fornecedor e ajustar diferentes parâmetros na avaliação de um fornecedor através dos seguintes princípios para avaliação de segurança de software:

- *A garantia de software não é alcançada por uma única prática, ferramenta ou lista de verificação; antes pelo contrário, é o resultado de um processo abrangente de engenharia de software seguro.*
- *A diversidade de abordagens usadas pelas organizações que adquirem software e as desigualdades na adoção de práticas de garantia de software pelas organizações de desenvolvimento de TI tornou claro que precisamos de uma abordagem em camadas para avaliar a segurança dos softwares adquiridos com base na maturidade do fornecedor de tecnologia que desenvolve o software.*
- *Os problemas atuais enfrentados por muitos clientes e fornecedores exigem uma solução imediata a curto prazo e que sejam amplamente aceites a médio / longo prazo pelos padrões internacionais abrangentes.*
- *Os clientes podem exigir evidências para apoiar as reivindicações de um fornecedor*
- *Os clientes precisam de conhecer o processo de garantia, tanto na empresa quanto no produto para apoiar as suas necessidades de gerenciamento de riscos*

Como organização focada na segurança, a SAFECode trabalha não apenas para melhorar as práticas de segurança de software, mas também para comunicar e demonstrar aos clientes o que essas melhores práticas significam nos seus esforços de gerenciamento de riscos uma vez que mais transparência nas práticas de garantia de software é essencial para que as principais partes interessadas gerenciem os riscos com mais facilidade e eficácia.

4 Níveis de Abordagem

A segurança no desenvolvimento de software pode e deve variar segundo diversos factores. A noção de segurança pode ser extremamente sensível para certos clientes, e importante mas não fundamental para outros.

Do ponto de vista dos avaliadores de software isto tem de ser levado em conta, com o objetivo de poupar tempo na avaliação da segurança do software, e para evitar descontentamento da parte dos clientes.

Para isso a *SAFECode* decidiu dividir os níveis de segurança em 3, e elaborou o seguinte diagrama de decisão para esquematizar o processo de situar o cliente no respetivo nível de segurança desejado.

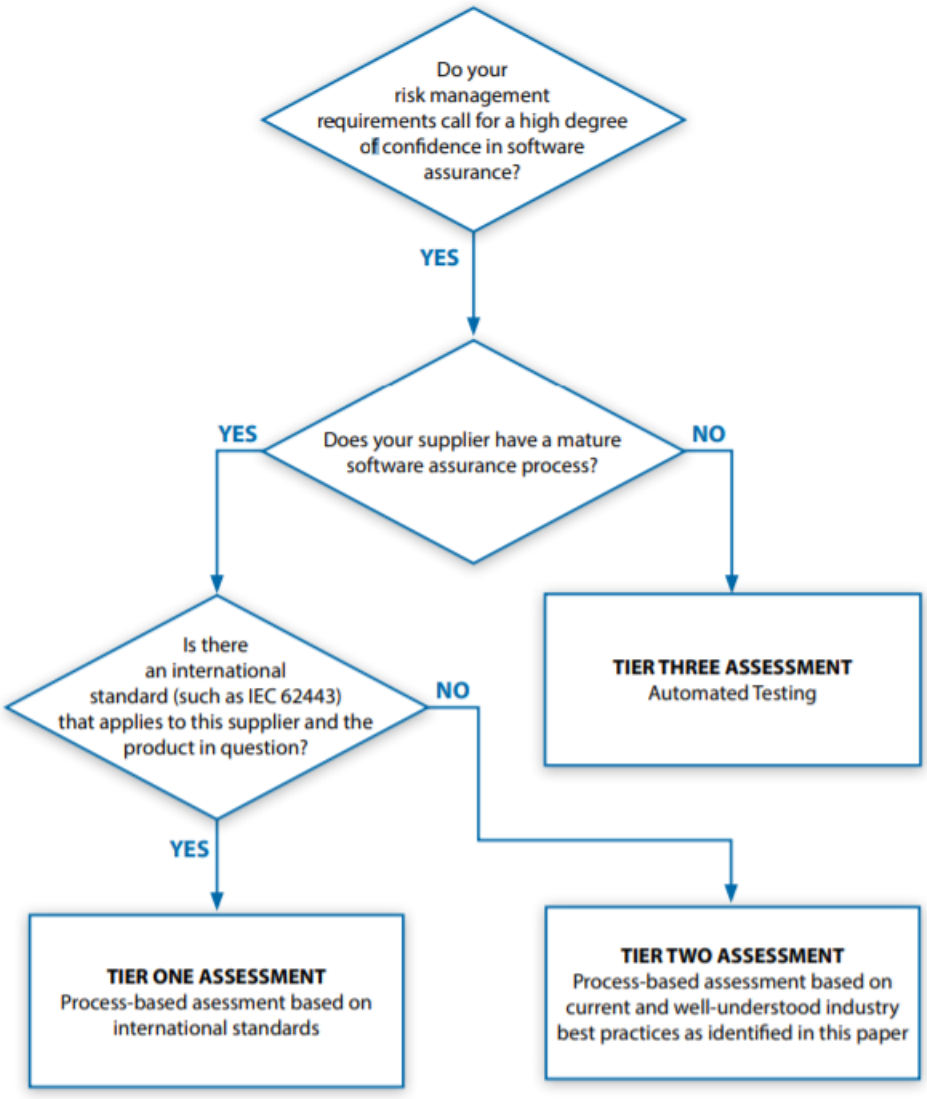

Figura 2: Diagrama dos Níveis de Avaliação

Tudo depende do software ou cliente a ser avaliado, se este tiver o processo de desenvolvimento bem enraizado e estiver disposto a mostra-lo então poderemos utilizar uma avaliação de nivel 1 ou 2.

Caso o processo de desenvolvimento não esteja bem especificado, ou não seja possível ao cliente mostrar-lo teremos que prosseguir para uma avaliação de nível 3.

4.1 Avaliação de Nivel 3

Neste nível o cliente não nos quer, ou não nos pode mostrar o seu processo desenvolvimento de software seguro. Teremos então de recorrer a testes automáticos.

Estes testes pode ser feitos utilizando ferramentas já conhecidas no mercado, e permitem encontrar falhas de segurança simples no software a ser desenvolvido.

Entre estas ferramentas estão, por exemplo, as de avaliação de código binário. Que possuem como principal vantagem a capacidade de avaliar a aplicação sem esta estar a correr, poupando tempo e recursos.

Apesar da sua utilidade, as ferramentas de testes automáticos têm dificuldade em analisar a arquitetura do software em si. O que faz deste nível uma opção de maior risco de segurança para o cliente.

4.2 Avaliação de Nivel 2

Neste nível admite-se que o processo de desenvolvimento de software seguro do cliente é conhecido e pode ser avaliado, mas não existe nenhum standard internacional que vá de encontro aquele cliente ou produto. Utiliza-se então um conjunto de técnicas bem conhecidas de avaliação da segurança do software. Estas técnicas irão ser especificadas no presente relatório nas secções Gestão da Segurança do Software, Práticas de Construção de Software Seguro, e Resposta a Vulnerabilidades

4.3 Avaliação de Nivel 1

Este é o mais critico, a avaliação da segurança deve ser minuciosa. Seguindo processos baseados em standard internacionais já definidos.

5 Gestão da Segurança do Software

Para garantir a segurança de um software, toda uma estrutura deve estar montada. Sendo assim possível dar respostas às diferentes etapas que constituem um desenvolvimento de software seguro.

A gestão desta estrutura é de grande importância para que o processo não falhe. É necessário saber se o cliente necessita que as suas equipas de desenvolvimento tenham treino, ou sejam ensinadas a desenvolver software seguro.

É preciso que os vários níveis de gestão da empresa estejam de acordo sobre a postura de segurança a adotar num determinado produto.

O cliente necessita também de ter especificadas estratégias para combater futuras falhas encontradas.

6 Práticas de Construção de Software Seguro

A SAFECode publicou *papers* com os fundamentos para a Prática de Construção de Software Seguro. Através destes *papers* eles tinham como objetivo auxiliar as indústrias com as melhores práticas de construção de Software para garantir que este seria seguro. Desta forma eles forneceram um conjunto de critérios comuns entre os clientes e os fornecedores do serviço que garantem um desenvolvimento cuidado e seguro do software.

Tendo estes critérios e práticas em mente, os clientes poderão tirar partido de algumas questões, que podem pôr aos seus fornecedores, para determinar como é que o seu software foi desenhado, construído e implementado, assim como perceber como foi a inclusão de componentes externas no seu produto:

- *O fornecedor define os requisitos de segurança específicos do produto como parte do seu ciclo de vida de desenvolvimento?*
- *O fornecedor realiza análises de risco da arquitetura ou modelo de ameaça como parte de seu ciclo de vida do produto e define mitigações apropriadas?*
- *O fornecedor realiza revisões de código estático automático para identificar defeitos de segurança introduzidos durante a codificação?*
- *O fornecedor realiza testes de segurança dinâmicos automatizados para identificar vulnerabilidades de segurança comuns?*
- *O fornecedor faz a triagem dos defeitos de segurança identificados a partir das atividades acima e os corrige como parte de seu ciclo de vida?*
- *O fornecedor tem um processo de gestão da cadeia de risco para gerir a segurança e integridade dos componentes de origem?*

Como vemos pelas perguntas em cima apresentadas, o cliente poderá perceber se a segurança foi realmente tida em conta. Podemos ver que deverá existir a preocupação de garantia da mesma desde a recolha de requisitos, passando para a fase do desenho de arquitetura e continuando na fase de implementação com testes a várias componentes diferentes do código.

6.1 Algumas práticas sugeridas...

Em cima foram descritas como o cliente poderá obter informação sobre o seu software através de algumas questões que poderão colocar ao seu fornecedor de serviço, ou seja, à entidade responsável pela construção do software. Porém, os *papers* publicados sugerem também as práticas a ter em conta pela parte dessas mesmas entidades que irão produzir um produto seguro para o cliente.

Assim na seguinte tabela vamos mostrar algumas práticas de construção de software seguro sugeridos e defendidos pela SAFECode.

Threat Modeling	Eliminar a Criptografia Fraca
Usar o Least Privilege	Usar técnicas de Logs e Tracking
Implementar Sandboxes	Determinar superfícies de ataque
Minimizar o uso de strings não seguras e funções de Buffer	Usar ferramentas de teste adequadas
Validar I/O para mitigar as Vulnerabilidades Comuns	Realizar testes de robustez
Utilizar operações robustas de inteiros para alocações dinâmicas de memória e offsets de arrays	Realizar testes de penetração
Usar bibliotecas XSS (Anti-Cross Site Scripting)	Use um conjunto de ferramentas do compilador recente
Usar dados em formatos canônicos	Usar ferramentas de análise estática
Evitar concatenações de Strings para comandos dinâmicos de SQL	

6.2 Entidades reguladores de segurança do produto

Os fornecedores de serviços que possuem um processo de garantia de segurança do software com um grau de maturidade considerável, possuem, normalmente, uma estrutura de regulação de segurança robusta que fornecem supervisão a todo o processo de garantia de segurança do software.

Estas entidades têm também a garantia que tanto o processo de construção de software como o processo de supervisão são bem explícitos e compreendidos por toda a sua organização.

Esta estrutura de regulação de segurança possuem características chave usadas para a sua revisão e garantia de segurança:

- *O fornecedor do serviço exige treino e aptidões de segurança na sua equipa de desenvolvimento de software e um método para garantir que os requisitos da sua estrutura e do seu processo de garantia de segurança de software sejam amplamente compreendidos?*
- *Os responsáveis apropriados de gestão da organização analisam e assinam a qualidade de segurança do produto?*
- *O fornecedor realiza um planeamento adequado para identificar e registar passos para, no futuro, solucionar quaisquer descobertas não mitigadas?*

A entidade reguladora deve adotar uma avaliação apropriada a quaisquer riscos identificados como parte do processo de desenvolvimento assim como parte das atividades realizadas durante o ciclo de vida da produção.

Porém não é esperado que a entidade reguladora corrija todos os defeitos de segurança do seu produto, mas ter uma abordagem baseada no risco desses mesmo defeitos. Assim eles demonstram a maturidade que o seu processo de construção de software seguro possui.

6.3 Resposta a Vulnerabilidades

Apesar de todo o cuidado que um fornecedor de serviço possua no seu processo de construção de software, com a continuação da evolução tecnológica ou até por má análise da equipa de construção do produto, é possível identificar futuramente algumas vulnerabilidades, quer pela parte do cliente, quer pelo utilizador final. Assim é necessário um cuidado extra pela parte do fornecedor de serviço em criar uma forma de manter contacto com o cliente.

Assim é necessário que estes garantam:

- *A existência de uma forma do cliente ou qualquer utilizador reportar vulnerabilidades que o produto possua.*
- *A existência de uma forma da entidade fornecedora do serviço comunicar aos clientes de vulnerabilidades através de alertas ou qualquer outro método que chegue ao cliente.*
- *O uso de ID's no formato da CVE para listar vulnerabilidades na NVD.*

7 Métodos de avaliação

De acordo com todos os aspetos referidos em cima neste relatório e usando a framework da SAFECode como base para a avaliação de segurança os clientes terão de optar por um método para aplicar ao seu fornecedor de serviço com base nos requisitos derivados do seu processo interno de análise e gestão de risco.

Na análise de um desenvolvimento seguro temos três abordagens diferentes que podemos tomar:

- **Transparência da documentação do processo:** Documentar o método de desenvolvimento de software seguro durante seu ciclo de vida. Muitas entidades que desenvolvem software documentam abertamente o seu processo em páginas web focadas em segurança, por meio de white papers ou posts em blogs públicos. Esta documentação pública inclui uma visão detalhada das práticas que a própria entidade considera importantes para o seu método de desenvolvimento de produto.

- **Partilha sob NDA(Non Disclosure Agreement):** Se a documentação pública não estiver disponível ou não for apropriada (muitas vezes para não revelar detalhes privados do método usado pela entidade fornecedora do serviço), o fornecedor pode ser capaz de compartilhar os detalhes sob NDA com o cliente quer para demonstrar confiança na relação fornecedor-cliente, quer para fornecer informações sobre o processo e regulação das medidas de segurança aplicadas.

- **Validação por terceiros:** No caso da validação por terceiros ser um requisito, a avaliação deve ser baseada nos elementos do processo e de regulação de métodos de segurança referidos na secção anterior.

8 Conclusão e Análise Crítica

Após a realização deste trabalho de pesquisa e leitura do *paper* sugerido pudemos tomar consciência do papel que a SAFECode adquiriu para todas as empresas fornecedoras de software, para que estas adoptem métodos e procedimentos que garantam uma construção de produtos seguros.

Desta forma pudemos tanto adquirir novos conhecimentos dos métodos que são usados no mundo empresarial, assim como perceber que alguns métodos abordados noutras UC's são de factos úteis e usados (como por exemplo *Threat Modeling*, Implementação de *SandBoxes*, Testes de Penetração).

Damos assim concluído o nosso trabalho prático, saindo do mesmo felizes e confiantes que os objetivos foram cumpridos com sucesso.