UNIVERSO – UNIVERSIDADE SALGADO DE OLIVEIRA
PRÓ-REITORIA ACADÊMICA
PÓS-GRADUAÇÃO EM SEGURANÇA DA INFORMAÇÃO

APLICAÇÃO DE UM MODELO DE MATURIDADE ABERTO NA IMPLANTAÇÃO DE PROGRAMAS DE MELHORIA DE SEGURANÇA DE SOFTWARE EM SISTEMAS CRÍTICOS DA ADMINISTRAÇÃO PÚBLICA FEDERAL BRASILEIRA

GOIÂNIA
2012
RAMON GOMES BRANDÃO

APLICAÇÃO DE UM MODELO DE MATURIDADE ABERTO NA IMPLANTAÇÃO DE PROGRAMAS DE MELHORIA DE SEGURANÇA DE SOFTWARE EM SISTEMAS CRÍTICOS DA ADMINISTRAÇÃO PÚBLICA FEDERAL BRASILEIRA

Artigo apresentado ao curso de Pós-Graduação em Segurança da Informação da Universidade Salgado de Oliveira – UNIVERSO, como parte dos requisitos para conclusão do curso.

Orientador: Prof. MSc. William Divino de Souza

GOIÂNIA
2012
RAMON GOMES BRANDÃO

APLICAÇÃO DE UM MODELO DE MATURIDADE ABERTO NA IMPLANTAÇÃO DE PROGRAMAS DE MELHORIA DE SEGURANÇA DE SOFTWARE EM SISTEMAS CRÍTICOS DA ADMINISTRAÇÃO PÚBLICA FEDERAL BRASILEIRA

Artigo apresentado ao curso de Pós-Graduação em Segurança da Informação da Universidade Salgado de Oliveira – UNIVERSO, como parte dos requisitos para conclusão do curso.

Nota:

PARECER

.....

.....

.....

.....

.....

.....

.....

.....

Goiânia, outubro de 2012

Prof. MSc. William Divino de Souza
Professor / UNIVERSO
Orientador / Examinador

APLICAÇÃO DE UM MODELO DE MATURIDADE ABERTO NA IMPLANTAÇÃO DE PROGRAMAS DE CODIFICAÇÃO SEGURA DE SOFTWARE EM SISTEMAS CRÍTICOS DA ADMINISTRAÇÃO PÚBLICA FEDERAL BRASILEIRA

Ramon Gomes Brandão[1]

Resumo

A Administração Pública Brasileira – APB, em suas esferas municipal, estadual e federal, vêm se apoiando, de forma consistente, no uso de sistemas de informática para sua atuação finalística. Com o advento da popularização da internet banda larga nos lares brasileiros e do acesso, cada vez maior, do cidadão à rede mundial de computadores, as chamadas *aplicações web* passaram a desempenhar importante papel, tanto no objetivo da prestação do serviço público de qualidade, quanto no funcionamento do próprio Estado. Tais sistemas são cada vez mais usados em boa parte da gestão da chamada infraestrutura crítica nacional. Entretanto, pela própria natureza dos sistemas *web* na Internet, a APB pode ficar exposta a grandes e inúmeros riscos (até mesmo riscos à soberania nacional), quando não se observa, dentre outros fatores, a qualidade do software no que tange às vulnerabilidades e aspectos de codificação segura de código. Sejam sistemas *web* desenvolvidos pelo próprio Estado, sejam aqueles obtidos através de contratações e aquisições, é necessário que a APB se atente constantemente à qualidade da segurança do *software* de seus sistemas críticos. Para apoiar essa necessidade e permitir ao Estado viabilizar, de forma evolutiva, ações de melhoria de segurança de *software* ao longo de todo o ciclo de vida de desenvolvimento, este trabalho estuda a viabilidade da utilização de um modelo de maturidade estruturado e aberto – o *Open Software Assurance Maturity Model* (OpenSAMM) – na realidade prática de um órgão público federal responsável pelo desenvolvimento e manutenção de um dos sistemas estruturantes de governo. Buscou-se a aplicação direta dos seus artefatos e a avaliação do esforço necessário para sua implementação.

Palavras chave: Codificação Segura. Modelo de Maturidade. OpenSAMM. Sistemas Estruturantes. Segurança de *Software*.

1 INTRODUÇÃO

[1] Graduado em Engenharia de Computação pela Universidade Estadual de Campinas – UNICAMP (2006). Atividade profissional em Segurança da Informação e Comunicações (SIC) desde 2007. Servidor Público Federal desde 2009, atuante na área de segurança de redes, *pentesting*, forense, inteligência e contra-inteligência, gestão de riscos e segurança de sistemas.

As aplicações desenvolvidas para a Internet[2] – doravante denominadas *aplicações web* – são uma realidade cada vez mais presente e indissociável da existência das corporações modernas. A portabilidade, facilidade e, sobretudo, escalabilidade que um sistema oferece ao ser acessível via navegador *web* (ou mesmo *web services*) revolucionou o mercado de tecnologia da informação (TI). Com o advento da computação em nuvem, as corporações encontram farto – e quase ilimitado – terreno para sua "expansão digital", onde essas aplicações desempenham papel fundamental. Quando se "aduba" esse terreno com a computação móvel, que é um fenômeno irrefutável dos últimos anos (e irrevogável nos vindouros), as aplicações *web* desempenham papel central ainda maior no que a grande maioria das pessoas entende por rede mundial de computadores, em todas as nações do mundo, em todas as culturas.

De uma maneira ou de outra, as sociedades modernas – incluindo a brasileira – são extremamente dependentes de tecnologia. Seja pela mais ampla gama de dispositivos, como computadores, *tablets*, *notebooks*, *hardwares* especializados, dispositivos de armazenamento e mídias, seja por toda a sorte de sistemas de informação, interconectados ou não por redes, não se pode dissociar a existência e o avanço humano daquilo que se conhece por tecnologia da informação e comunicação, tanto nos planos econômico e social, quanto cultural. Como nos traz MANDARINO (2010), a humanidade vive hoje o status de *sociedade da informação*.

Entretanto, ao mesmo tempo que a tecnologia é absorvida para o bem da sociedade da informação, ela também é um prato cheio para infindáveis tipos de agentes maliciosos. A velocidade com que esses atores, cujas intenções são permeadas pela ilicitude, se apropriam da evolução tecnológica, utilizando toda a sorte de ações criminosas, chega a ser assustadora. No mundo digital, sobretudo, valendo-se da ainda incipiente cultura de segurança da informação existente na sociedade, agentes criminosos desenvolvem inúmeros tipos de técnicas e capacidades de ataques à pessoas e aos sistemas – ou mesmo a Estados inteiros –, com as mais variadas motivações, como obtenção financeira, demonstração de posição ideológica ou política, ou ainda pior, a prática de atos terroristas ou extremistas facilitados por meio da interconexão promovida pelas redes de

[2] Nota do Autor: o termo "Internet", aqui empregado, designa toda a sorte de redes (intranets e extranets) nas quais a utilização de sistemas *web* é suportada.

computadores, que minimizam (ou mesmo extinguem) o risco sobre os próprios atacantes – conhecido como *cyberterrorismo*. (DCSINT, 2005).

Neste cenário, as aplicações *web*, ao mesmo tempo que provêm a interface tecnológica para a interação digital definitiva e positiva da sociedade, constituem-se como importantes (e muitas vezes os principais) vetores de ataque desses atores maliciosos[3]. Estes visam, através do comprometimento da segurança desses sistemas, atacar a própria sociedade da informação, muitas vezes desestruturando, no âmbito das nações, suas corporações, seus sistemas financeiros e, sobretudo, seus próprios Estados, quando é colocada em risco a continuidade da prestação do serviço público. Tal desestruturação ocorre caso o atacante logre sucesso no objetivo de comprometer as infraestruturas críticas da informação, definida pelo Gabinete de Segurança Institucional da Presidência da República – GSI/PR:

> As infraestruturas críticas da informação (ICI) são assim definidas como o subconjunto de Ativos de Informação – meios de armazenamento, transmissão e processamento, sistemas de informação, bem como os locais onde se encontram esses meios e as pessoas que a eles tem acesso – que afetam diretamente a consecução e a continuidade da missão do Estado e a segurança da sociedade. (GSI/PR, 2010)

Dado que a gestão governamental se apoia, majoritariamente, em infraestruturas críticas formadas por sistemas integrados de informação – sistemas estruturantes ou estruturadores (CINTRÃO, 2008) – em sua maioria acessíveis via internet e baseados em aplicações *web*, é necessário que se volte a atenção para a qualidade do *software* desses sistemas, no tocante aos seus aspectos de segurança da informação, segurança das comunicações e, sobretudo, resiliência a ataques, explorações e ações maliciosas que objetivam comprometer a sua disponibilidade, confidencialidade e autenticidade. Uma das formas eficientes e eficazes de materializar tal preocupação é introduzir, ao longo de todo o ciclo de vida desses sistemas estruturantes (das fases preliminares de projeto e desenvolvimento até a sua publicação e manutenção), requisitos de segurança de informação e comunicações, através de aspectos de codificação segura de *software* e mitigação de vulnerabilidades.

[3] Os estudos e estatísticas que fundamentam tal afirmativa serão abordados nas próximas seções deste trabalho.

Esta tarefa, no entanto, não é trivial. É necessária uma mudança de paradigma na concepção e desenvolvimento de sistemas de TI, de forma a inserir controles de qualidade de segurança de informação na codificação de *software*. Além disso, essa mudança tem que ser sustentável, viável e manutenível ao longo do tempo, pois deve existir em todas as versões dos sistemas, sobretudo as vindouras. Equipes e pessoas envolvidas devem ser capacitadas no tocante à codificação segura. A preocupação com a segurança dos sistemas que expõem as corporações estatais na *web* deve ser estratégica, e não apenas operacional, como o é na grande maioria das vezes. E, principalmente, as corporações devem ser capazes de melhorar, amadurecer e manter esses controles, de forma estruturada e rastreável.

O desafio é grande para os profissionais da área, para as corporações privadas e, sobretudo, para os Estados. É necessária a existência de meios estruturados, metodológicos e práticos, que sejam capazes de apoiar esses atores no processo de introdução e amadurecimento da preocupação com a segurança da informação das suas aplicações *web*. No âmbito dos Estados, a necessidade é ainda mais latente, dado o uso fundamental da Internet na gestão governamental e na melhora da prestação do serviço público de qualidade.

Sob esta ótica, há alguns anos vem crescendo, no universo de TI mundial, além de novas tecnologias de segurança baseada em ativos de rede, iniciativas globais que visam a estruturação de modelos de maturidade e de boas práticas para apoiar corporações na tarefa de implementar aspectos de segurança no desenvolvimento de aplicações, sobretudo dos sistemas *web.* Dentre tais iniciativas, destaca-se o *Software Assurance Maturity Model* – "OpenSAMM" (CHANDRA, 2008), um projeto aberto sob licença Creative Commons 3.0, apoiado desde a sua concepção pelo *Open Web Application Security Project* – OWASP [6], uma comunidade aberta mundialmente, cujo foco é a melhora na segurança de *software*. O objeto de estudo desse artigo é, portanto, a aplicação direta do modelo proposto pelo OpenSAMM – incluindo-se seus artefatos, práticas e métricas – e, consequentemente, a análise da viabilidade de seu uso por uma instituição pública federal responsável pelo desenvolvimento, execução e manutenção de um dos sistemas *web* estruturantes do Estado brasileiro. A escolha desse modelo motivou-se tanto pela aparente simplicidade, quanto pelo fato de ser aberto, adaptável, livre e

pelo fato de se apoiar em uma comunidade com presença oficial no Brasil (o OWASP está presente em diversas cidades), inclusive com um capítulo ativo na capital federal[4].

2 DESENVOLVIMENTO

2.1 Dados e Estatísticas – Segurança das Aplicações e vazamento de dados – Motivação

Vem sendo observado, ao longo dos últimos anos, um crescimento substancial da severidade e da quantidade de *cyber* ataques às aplicações, tanto das pequenas quanto das grandes organizações. Tais ações levam consigo, em sua maioria, o que as corporações tem de mais valioso: a sua informação sensível, o seu conhecimento. Para os efeitos deste trabalho, considerando-se a natureza dos órgãos públicos, dar-se-á especial atenção aos números referentes à grandes organizações.

No cenário mundial, estatísticas recentes (VERIZON,2012) revelam que, em 2011, em se tratando de grandes corporações, do universo total das ações maliciosas amostradas (855 incidentes confirmados) e das informações comprometidas (174 milhões de registros), cerca de 98% dos ataques são originários de agentes externos, responsáveis pelo vazamento de 99% do total de dados. Tais agentes externos consistem, na sua maioria, de grupos criminosos organizados (33%) e de grupos ativistas (21%), estes últimos responsáveis por quase 60% dos dados vazados relativos a esses agentes. Esses grupos ativistas são responsáveis pelo que se conhece como *hacktivismo*[5] - desfiguração (*defacement*) de sites *web*, ataques coordenados de negação de serviço e outros atos para expressar revolta, ideologias, protesto, reivindicação de direitos ou demonstração de posições econômicas ou políticas contrárias, dentre outros. Além disso, o repertório de ações do *hacktivismo* passou a incluir ainda, geralmente com grande publicidade e intensidade, grandes vazamentos de dados.

A principal motivação dos agentes externos é, substancialmente, financeira ou ganho pessoal. O que se observou, entretanto, é a grande porcentagem de motivações ligadas à questões ideológicas ou de protesto – 25% dos casos (Gráfico

[4] https://www.owasp.org/index.php/Brasilia

[5] O termo *hacktivismo* foi definido pela primeira vez em 1996 por um membro do grupo *hacker* "Cult of the Dead Cow". Fonte: <http://en.wikipedia.org/wiki/Hacktivism>, acesso em 10 set. 2012.

1). Quando se considera que os Estados, por sua própria natureza, são alvos históricos do ativismo, este dado torna-se preocupante.

Em relação às categorias dos ataques à grandes corporações, a grande maioria dos agentes mal intencionados basearam suas ações em atividades de *hacking*[6] (58%), uso de programas maliciosos – *malwares* (28%), ou uma combinação dos dois, dentre outras técnicas (Gráfico 2).

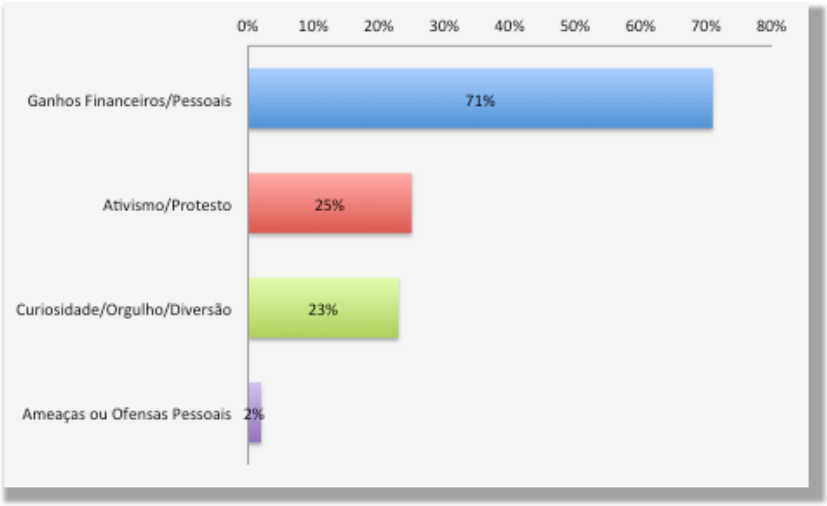

Gráfico 1: motivação dos agentes externos por porcentagem de ataques perpetrados por agentes externos. (Fonte: VERIZON, 2012 – adaptado)

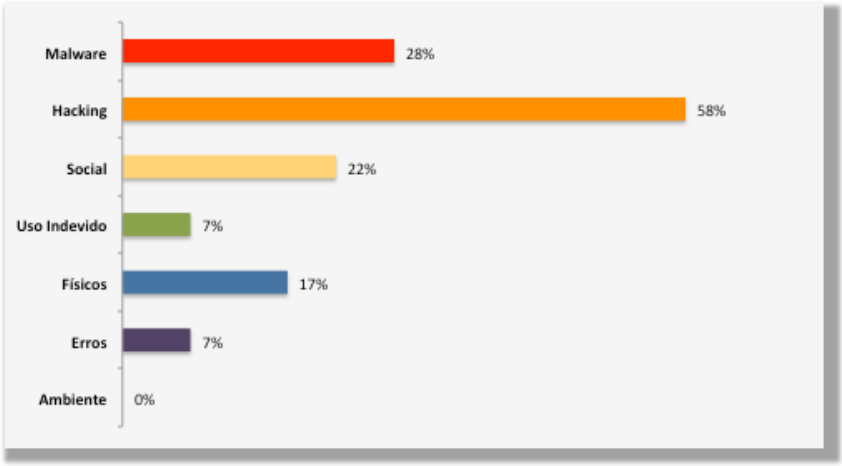

Gráfico 2: categorias de ameaças às grandes corporações por percentual de ataques. (Fonte: VERIZON, 2012 – adaptado)

Os dados que mais chamam a atenção são referentes aos vetores de ataques utilizados nas atividades de *hacking*, quando se considera as grandes corporações

[6] Segundo (VERIZON, 2011, 2012), o termo "*hacking*" é definido como a tentativa intencional de acesso ou comprometimento de ativos de informação, sem autorização ou excedendo a autorização, frustrando controles lógicos de segurança (tradução livre do autor).

(Gráfico 3). Apesar da presença de equipes de TI dedicadas e dos controles de segurança mais maduros observados nessas organizações quando comparadas àquelas de pequeno porte, as aplicações *web* foram utilizadas em mais da metade (54%) dos casos, e quase 40% dos dados comprometidos foram obtidos por este canal.

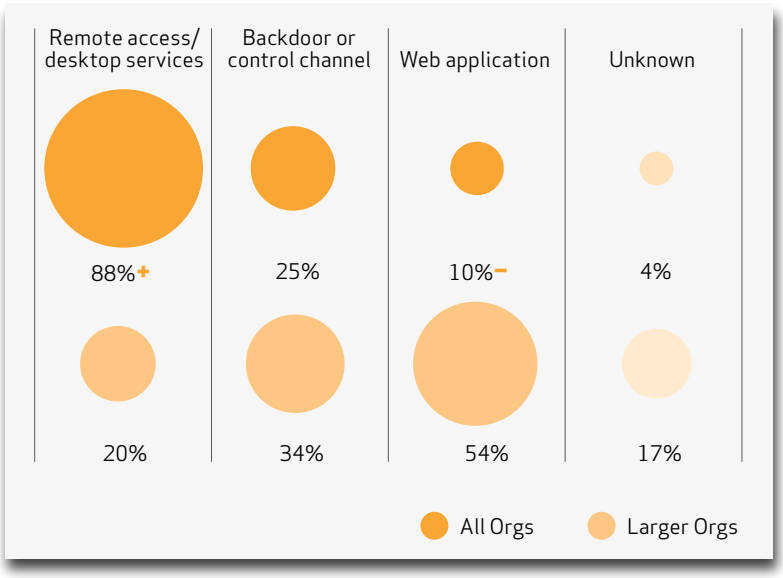

Gráfico 3: comparação entre principais vetores utilizados pelos ataques de *Hacking* nas organizações de todos os tamanhos (*all orgs*) e nas grandes corporações (*larger orgs*).
(Fonte: VERIZON,2012)

A necessidade inerente de visibilidade na Internet das aplicações *web* as tornam alvos naturais e importantes. Logo, a possibilidade de usá-las como vetor de entrada aos bancos de dados das corporações, e mesmo aos perímetros protegidos das organizações, atraem substancialmente o foco dos agentes maliciosos. Tal afirmação pode ser inferida do fato de que pelo menos metade (50%) dos ataques às grandes corporações foram direcionados – os alvos foram escolhidos e suas vulnerabilidades foram pesquisadas e exploradas – e não oportunistas (35%), quando os alvos não são escolhidos, mas apresentam deficiências que os atacantes sabem como explorar (Gráfico 4).

Para qualquer tamanho de organização, tão essencial quanto se proteger dos *cyber* ataques, é identificar tempestivamente a sua ocorrência e tomar, o mais rápido possível, as ações de mitigação dos possíveis prejuízos, bem como a correção das vulnerabilidades. Principalmente para as grandes organizações, isso é questão de sobrevivência de mercado e de imagem corporativa com seus consumidores, e, em

especial para os Estados, pode ser questão de soberania nacional. Entretanto, observa-se que as grandes corporações ainda engatinham na capacidade de, ativamente, detectar que foram vítimas. Em apenas 16% dos casos de ataques bem-sucedidos, tais organizações contaram, de forma ativa, com tecnologias como IPS/IDS/HIPS, monitoramento de *logs* e antivírus (dentre outras) para a prevenção, detecção e resposta às ações maliciosas. A "notícia" que a organização foi vítima de

Gráfico 4: direcionamento dos ataques em relação ao número de ocorrências relacionadas a grandes organizações. (Fonte: VERIZON, 2012 – adaptada)

um *cyber* ataque foi dada, em 28% dos casos, por algum funcionário ordinário da corporação, não relacionado à gestão da segurança das aplicações, após notar "algo estranho" no desempenho de suas atividades diárias. E finalmente, em 49% das ocorrências, a notícia foi dada por atores externos às organizações, terceiros, clientes, órgãos reguladores e, na maioria relativa dos casos (21%), pelo próprio agente malicioso que perpetrou o ataque, após divulgar publicamente sua ação – e, ainda pior, os dados roubados – em mídias ostensivas como blogs, redes sociais, etc, ou ao efetuar chantagens, provocações ou ameaças às corporações vitimadas.

Considerando-se especificamente o setor público, este se configura em um dos cinco primeiros grupos em termos do percentual de ataques a dados, com participação de 7% do total. À primeira vista, o valor pode parecer baixo, mas torna-se substancialmente expressivo ao se trazer à mente que os Estados executam e gerenciam sistemas relacionados à infraestrutura crítica da informação, cujo comprometimento traz enorme impacto para suas nações. De acordo com a Symantec (2012), o Setor Governamental é o segundo colocado em relação ao

número de vazamento de dados que puderam levar ao comprometimento de identidades e credenciais, com 14% em participação do total, atrás apenas dos setores de assistência médica e saúde.

Tratando-se especificamente do caso brasileiro, o país figurou como quarto colocado em atividades maliciosas no mundo digital, no período 2010-2011, detentor de uma média de 4,1% de todos os casos de códigos maliciosos, *spam zombies*, *phishing hosts*, computadores infectados com agentes de *botnets,* ataques contra redes e ataques contra *sites web*, atrás apenas de Estados Unidos, China e Índia (SYMANTEC,2011). No caso específico da Administração Pública Federal – APF, dados do Centro de Tratamento de Incidentes de Segurança de Redes de Computadores da APF – CTIR.Gov (órgão ligado à Presidência da República responsável pelo atendimento aos incidentes em redes de computadores da APF), referentes ao primeiro semestre de 2012 (CTIR.Gov, 2012), mostram que, dos 5683 eventos notificados ao órgão, mais de trinta por cento (30,84%) foram relacionados a ataques de "abuso" a *sites web* do governo, os quais, em 81,68% dos casos, sofreram desfiguração de sítio e em 11,34% das ocorrências, links de *spam* e de comerciais não legítimos foram injetados (Gráfico 5). Conforme elucida o órgão, "estes ataques são realizados por meio da exploração de vulnerabilidades em servidores *web* mal configurados e, **principalmente, aplicações *web* mal codificadas**" (grifo nosso).

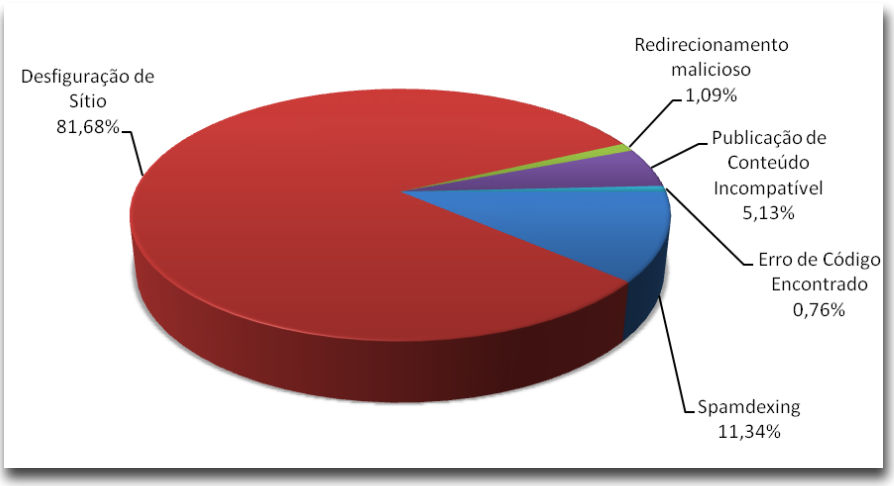

Gráfico 5: distribuição dos subtipos de abuso de sítios *web* das redes da APF. (Fonte, CTIR.Gov, 2012)

Tais dados se agravam quando são observados à luz do último Levantamento

de Governança de TI realizado pelo Tribunal de Contas da União (TCU, 2010), nos órgãos da APF (levantamento realizado desde 2007). De 265 órgãos avaliados nos três poderes da União, observa-se uma situação ainda precária no que diz respeito à segurança da informação. Notou-se a ausência de práticas recomendadas de análise de riscos em 84% das instituições, bem como a inexistência de gestão de incidentes de segurança da informação em 76% dos casos, dentre outras graves deficiências (Gráfico 6). Mais da metade dos órgãos não contam com processo de desenvolvimento de *software*, ao menos definido ou minimamente gerenciado. Além disso, 54% das entidades não realizaram auditorias de TI nos últimos 3 anos, e apenas 20% dos órgãos realizaram alguma auditoria nos seus sistemas de informação. Tais auditorias são atividades essenciais para o apoio à gestão de riscos relacionados aos sistemas e aplicações e peças essenciais à sua segurança.

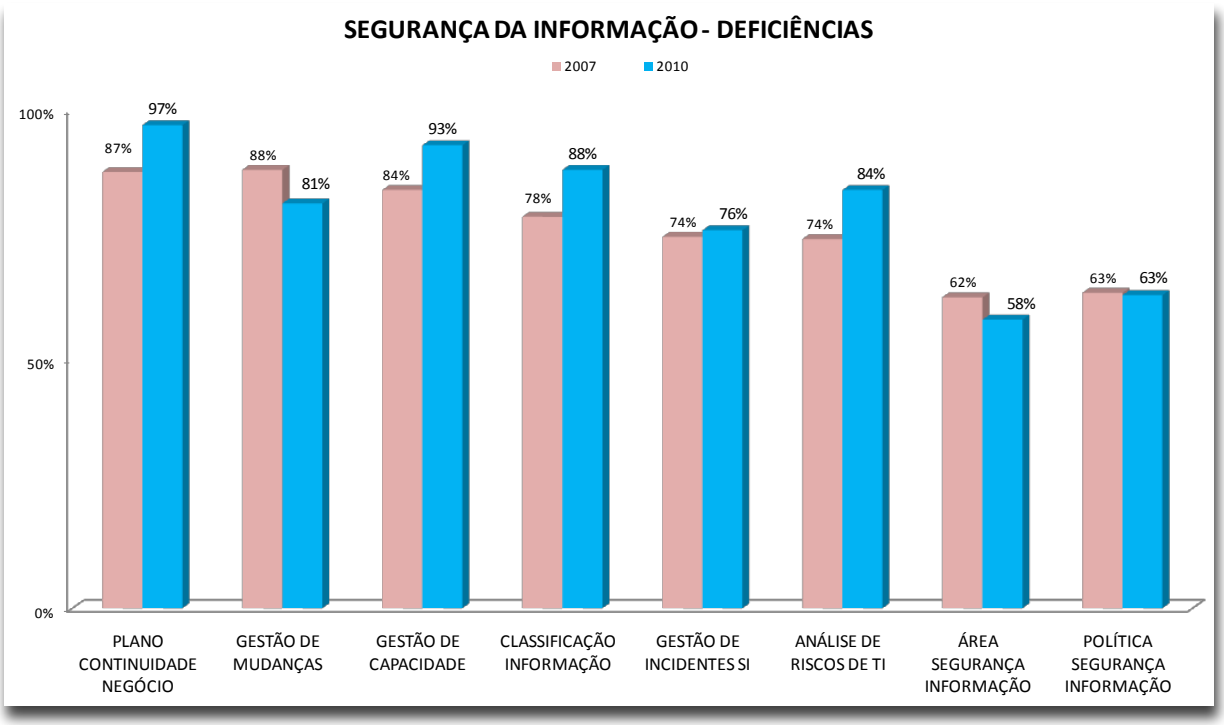

Gráfico 6: evolução dos indicadores de deficiência em seg. da Informação nos órgãos da APF. (Fonte: TCU, 2010)

Por fim, no tocante aos tipos de vulnerabilidades encontradas, em se tratando das aplicações *web* exclusivamente do setor governamental (sejam elas desenvolvidas *in house*, adquiridas ou *outsorced*), a empresa VERACODE (2011) mostra que deficiências de *Cross-Site Scripting* (XSS), vazamento de informações e Injeção SQL são protagonistas na maioria dos casos (Gráfico 7).

Frente a todas as estatísticas apresentadas nesta seção, é, de certa forma, indiscutível a necessidade de uma abordagem estruturada, compreensiva e mensurável tanto para a análise e tratamento dos riscos, quanto para a melhora da segurança das aplicações *web*. No caso da APF brasileira, tal preocupação torna-se fundamental e legítima, ao se considerar que o Brasil será palco de grandes eventos nos próximos anos, como a Jornada Mundial da Juventude e a Copa das Confederações (2013), a Copa do Mundo (2014) e os Jogos Olímpicos (2016). O mundo estará com os holofotes voltados para o país, assim como o crime organizado, os grupos *hacktivistas* e toda sorte de ameaças aos sistemas críticos nacionais de informação.

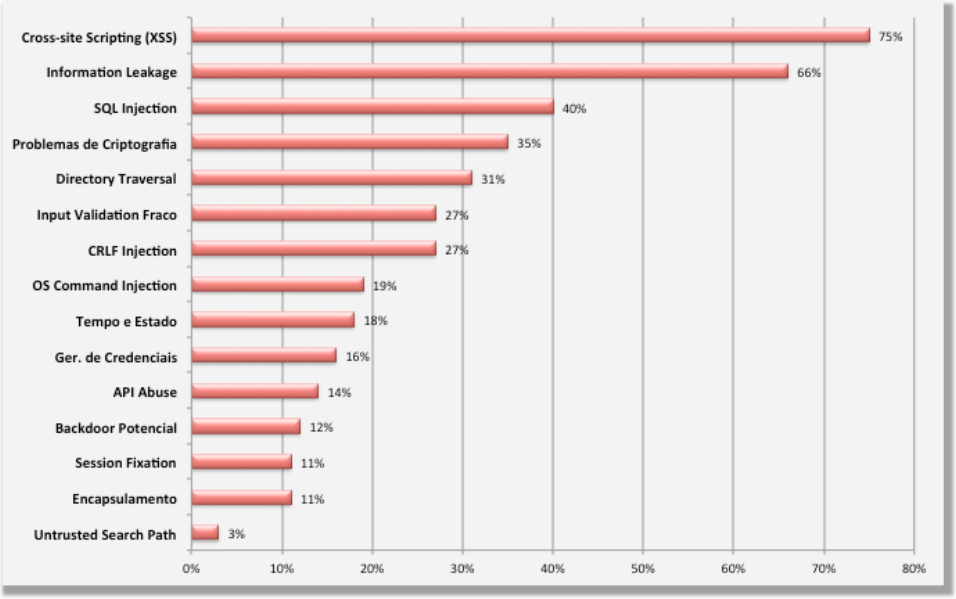

Gráfico 7: principais categorias de vulnerabilidades observadas nos sistemas do setor governamental, em percentual de aplicações *web* afetadas.
(Fonte: VERACODE, 2012 – adaptada)

2.2 Modelos, Iniciativas, Normativos e Legislação Correlata – Breve Resumo

Podem ser encontradas na literatura especializada diversas referências relacionadas ao tema de segurança da informação (SI) voltadas à segurança de aplicações, que também envolvem a abordagem sobre sistemas *web*.

Dentre as mais conhecidas, podemos citar, a princípio, as normas da família ISO 27000, em foco a NBR ISO/IEC 27002:2007 e a NBR ISO/IEC 27005:2008, que estabelecem, respectivamente, “diretrizes e princípios gerais para iniciar, implementar, manter e melhorar a gestão de segurança da informação em uma

organização" e "diretrizes para o processo de gestão de riscos de segurança da informação". Perpassando todos os aspectos da SI, tais diretrizes englobam a segurança de aplicações de forma geral, como o faz a primeira norma, através de sua seção 12 – "Aquisição, Desenvolvimento e Manutenção de Sistemas de Informação". Apenas em novembro do último ano que foi editada a parte introdutória da primeira norma internacional que trata, especificamente, do guia para apoiar as organizações na inclusão de controles de segurança nas aplicações – a norma ISO/IEC 27034-1:2011. Essa parte 1, de um total de seis, trata apenas dos conceitos, definições e princípios básicos da segurança de *software*, mas já direciona os trabalhos para a definição normativa de um processo formal de gerenciamento de segurança de aplicações.

O *National Institute of Standards and Technology* – NIST – trouxe, em 2009 (NIST SP 800-53 rev 3), um conjunto de 145 controles de segurança para os sistemas de informação federais dos Estados Unidos, que buscam satisfazer "a amplitude e a profundidade dos requisitos de segurança" de tais sistemas. Motivado por tal publicação, o SANS Institute sumarizou esta centena de controles do NIST, e também de outras fontes, em 20 controles críticos essenciais para o tratamento da SI das organizações – *20 Critical Security Controls* (SANS-CSIS, 2011), trazendo no controle número 6 abordagem específica para segurança de aplicações *web* e não *web*.

Quanto ao desenvolvimento de aplicações e sistemas, a Microsoft disponibiliza gratuitamente, desde 2008, a sua proposta para ciclo de vida de desenvolvimento seguro de *software* (MS-SDL, 2012). De forma semelhante, o OWASP provê um conjunto gratuito e aberto de processos de segurança que podem ser integrados em qualquer ciclo de vida de desenvolvimento de *software* – o *Comprehensive, Lightweight Application Security Process* (CLASP-CHANDRA, 2007). Empresas de consultoria, como a Citigal Inc., comercializam diversos serviços de apoio ao desenvolvimento seguro para corporações (CIGITAL, 2012). Na avaliação de segurança de aplicações, organizações como HP, IBM, Acunetix e Veracode oferecem ferramentas tanto de análise estática de código (SAST[7]) quando de análise dinâmica de aplicações (DAST[8]). Muitas destas ferramentas se baseiam em *rankings* das mais críticas vulnerabilidades de sistemas *web*, como o OWASP

[7] Acrônimo para *Static Application Security Testing.*
[8] Acrônimo para *Dinamic Application Security Testing.*

Top 10 (OWASP-Top10, 2010) e o CWE/Sans *Top 25 Most Dangerous Software Errors* (CWE-SANS-Top25, 2011).

No tocante aos frameworks estruturados abertos de boas práticas, destaca-se o *Building Security In Maturity Model 4* (MCGRAW *et al*, 2012), resultante do estudo das iniciativas de segurança em aplicações utilizadas por cinquenta e uma empresas ao redor do mundo, e também o framework aberto que é o objeto de estudo deste trabalho: o OpenSAMM, abordado na próxima seção.

Por fim, no âmbito brasileiro, a segurança de aplicações *web* é levada muito a sério pelo setor financeiro nacional[9]. Entretanto, especificamente no setor governamental (objeto das considerações deste artigo), as iniciativas são incipientes e isoladas, mas o cenário começa a mudar. Atualmente[10], o GSI/PR, dentro de suas competências funcionais, está em fase final de elaboração de minuta de projeto de Norma Complementar à Instrução Normativa GSI N. 1, a qual versará sobre "diretrizes para desenvolvimento e obtenção de software seguro nos órgãos e entidades da APF". Espera-se que este instrumento jurídico traga o norte definitivo para o desenvolvimento da musculatura de segurança de aplicações *web* no Estado brasileiro.

2.3 O *Framework* OpenSAMM – Características

De acordo com sua definição[11], o *Software Assurance Maturity Model*, conhecido como OpenSAMM, é um conjunto livre, flexível e aberto de recursos e conceitos (doravante nomeado por *framework*), cujo objetivo é apoiar organizações, de todos os tamanhos e com quaisquer estilos de desenvolvimento de aplicações, na formulação e implementação de estratégias para a segurança de *software*, feitas sob-medida para os riscos que tais corporações enfrentam. Tais objetivos são alcançados pelo *framework* através da: avaliação do estado atual de práticas de segurança de *software* existentes na organização; construção de um programa de acreditação de segurança de *software* dividido em iterações bem definidas; demonstração de melhorias concretas através desses programas e definição e

[9] "Febraban investe R$ 7 bilhões por ano em segurança da informação". Disponível em <http://colunistas.ig.com.br/leisenegocios/2012/07/18/febraban-investe-r-7-bilhoes-por-ano-em-seguranca-da-informacao/>. Acesso em ago/2012.

[10] Informação dada através de contato telefônico direto com o Órgão.

[11] O OpenSAMM não conta ainda com tradução oficial para o português. Todos os conceitos são traduções livres ou adaptações deste autor, a partir do original em inglês.

medição de atividades ligadas à tais programas de melhoria de segurança de *software* em toda a organização.

O *framework* se apóia em três princípios básicos. O primeiro é o entendimento de que a mudança de comportamento das organizações é lenta. Em decorrência desse fato, um programa de melhoria em segurança de *software* deve ser especificado em pequenas iterações, as quais devem ser capazes de oferecer ganhos tangíveis e práticos no curto prazo, ao passo que fornecem o substrato para alcançar objetivos maiores de longo prazo. O segundo leva em consideração que não existe uma receita única para todo tipo de organização. As estratégias de melhoria devem ser flexíveis e devem permitir sua adaptação à realidade das corporações, quase sempre substancialmente diferentes tanto na tolerância a riscos quanto na forma de uso das aplicações. O terceiro princípio, por fim, considera que as práticas e guias relacionados às atividades de segurança em geral devem ser prescritivas, simples, bem definidas e, sobretudo, mensuráveis. Observa-se, nesses princípios, características típicas de modelos de maturidade, a exemplo do que é observado no CMMI e no modelo presente no COBIT.

O modelo proposto pelo OpenSAMM constrói-se a partir de 12 práticas de segurança (*security practices* – SP) distribuídas entre quatro funções essenciais de negócio (*business functions* – BF), onde cada BF contém três SP's (Figura 1). Tais BF's são abordadas, em algum grau, por qualquer organização que esteja envolvida com o desenvolvimento ou aquisição de *software*. O caráter iterativo do modelo advém de três níveis de maturidade definidos para cada uma das 12 SP's, que resultam em um amplo conjunto de atividades que as organizações podem se engajar e adotar para reduzir os riscos associados às aplicações, e promover a melhora e a acreditação de segurança de seus *softwares*. Os níveis de maturidade contém o detalhamento básico das atividades envolvidas e suas recomendações, resultados esperados, métricas específicas de sucesso e recursos necessários, bem como estimativas qualitativas dos custos envolvidos para o atingimento de tal nível.

Cada nível de maturidade é caracterizado por um objetivo, que é sucessivamente mais sofisticado (com métricas de sucesso mais refinadas) à medida que as atividades de determinado estágio são cumpridas em detrimento do nível anterior. Ademais, cada SP pode ser melhorada independente das outras, mas invariavelmente levará a otimizações nas demais, dado as correlações existentes

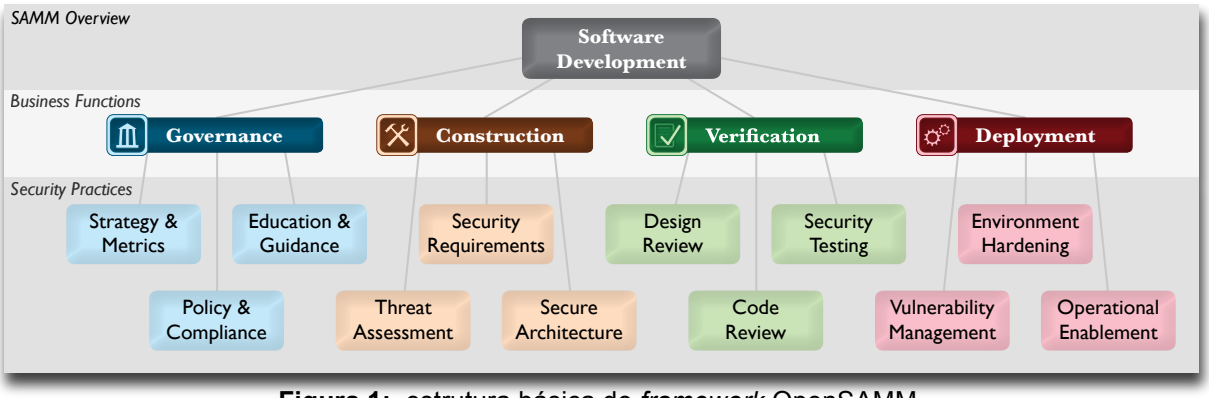

Figura 1: estrutura básica do *framework* OpenSAMM.
(Fonte: CHANDRA, 2008 [5])

entre as atividades dos diversos níveis.

As BF's consistem em:

a) **Governança (*Governance*)**: centrada nos processos e atividades relacionados à forma que a organização realiza a gestão de seus processos de desenvolvimento de *software*. Abrange os grupos envolvidos e os processos estratégicos de negócio referentes ao desenvolvimento;

b) **Construção (*Construction*)**: preocupa-se com os processos e atividades relacionados a como a organização define objetivos e cria *softwares* para atendê-los, dentro de projetos de desenvolvimento. Perpassam processos de gerenciamento do produto, engenharia de requisitos, especificação de arquiteturas, *design* de *software* e implementação;

c) **Verificação (*Verification*)**: focada nos processos e atividades relacionadas à forma que a corporação testa e verifica seus artefatos produzidos durante a fase de desenvolvimento de *software*. Inclui, tipicamente, garantia e controle de qualidade das aplicações; e

d) **Implantação (*Deployment*)**: aborda os processos e atividades relacionadas à forma que a organização gerencia a implantação do *software* que foi criado. Envolve, dentre outros fatores, a entrega da aplicação para os usuários, implantação em servidores externos ou internos e a execução em ambiente de produção.

As práticas de segurança, por sua vez, são definidas no modelo como[12]:

[12] O detalhamento de cada SP e suas atividades envolvidas, dada a sua extensão, não será abordado neste trabalho. Está disponível na fonte [5], pág 32.

Referentes à Governança:

i) **Estratégia e Métrica (*Strategy & Metrics* – SM)**: engloba o direcionamento estratégico geral do programa de acreditação de segurança de *software*, bem como a instrumentação de processos e atividades para coletar as métricas relacionadas à postura de segurança da instituição;

ii) **Política e Conformidade (*Policy & Compliance* – PC)**: engloba a implementação de um ferramental de gestão de segurança, conformidade e auditoria, com o intuito de viabilizar, de forma crescente e contínua, a acreditação de segurança de software em construção e em operação;

iii) **Educação e Instrução (*Education & Guidance* – EG)**: engloba o treinamento e capacitação em desenvolvimento seguro de *software* e tópicos envolvidos, para todos os profissionais envolvidos no processo, de acordo com seus papéis;

Referentes à Construção:

iv) **Avaliação de Ameaças (*Threat Assessment* – TA)**: reúne as atividades de identificação e caracterização precisa dos potenciais ataques direcionados aos *softwares* da organização, de forma a auxiliar os processos de análise e gestão de risco;

v) **Requisitos de Segurança (*Security Requirements* – SR)**: engloba as atividades que promovem a inclusão de requisitos de segurança ao longo do ciclo de desenvolvimento de *software*, desde a sua concepção;

vi) **Arquitetura Segura (*Secure Architecture* – SA)**: engloba o reforço do processo de design da arquitetura de software com atividades que promovam a inserção de controles, tecnologias e recursos de segurança, por padrão, na fase de design;

Referentes à Verificação:

vii) **Revisão de Design (*Design Review* – DR)**: engloba a inspeção dos artefatos gerados no processo de design, de forma a assegurar, no *software*, mecanismos adequados de segurança, aderentes aos padrões da organização;

viii) **Revisão de Código (*Code Review* – CR)**: engloba a avaliação e revisão do código fonte das aplicações da corporação, para promover a

descoberta de vulnerabilidades e as atividades de mitigação correspondentes, além de estabelecer uma linha base para padrão de código seguro;

ix) **Testes de Segurança (*Security Testing* – ST)**: engloba os testes de *software* nos seus ambientes de produção, tanto para descobrir vulnerabilidades quanto para definir padrões mínimos de segurança para liberação de código;

Referentes à Implantação:

x) **Gestão de Vulnerabilidades (*Vulnerability Management* – VM)**: atividades que englobam o estabelecimento de processos consistentes de gerência de vulnerabilidades, reportadas tanto internamente quanto de fontes externas, de forma a reduzir a exposição a riscos e a otimizar o próprio processo de acreditação de software.

xi) **Segurança do Ambiente (*Environment Hardening* – EH)**: engloba a implementação de controles de segurança no ambiente operacional que suporta o *software* da organização, de forma a reforçar o caráter seguro do *software* que foi implantado; e

xii) **Habilitação Operacional (*Operational Enablement* – OE)**: engloba identificar e capturar informações relevantes para o operador do sistema, permitindo que este configure, implante e execute, nos ambientes adequados, as aplicações da organização.

E para cada uma das SP's de i a xii, três níveis de maturidade, designados no intervalo de 1 a 3 (e mais um nível zero, implícito), se fazem presentes. Representam, por sua vez:

0) **Nível Zero**: nenhuma atividade da SP é cumprida pela organização;
1) **Nível 1**: entendimento inicial do que são as SP's, mas iniciativas ainda *ad hoc* no cumprimento de uma ou outra atividade;
2) **Nivel 2**: crescimento da eficácia e da efetividade no cumprimento das atividades relacionadas à SP; e
3) **Nível 3**: cumprimento e otimização compreensiva, em escala, da SP como um todo.

A aplicação do OpenSAMM é, de certa forma, intuitiva. O modelo propõe uma planilha de avaliação (anexa a este artigo)[13], que consiste em uma série de perguntas relacionadas às iniciativas de segurança de *software* realizadas pela organização. Tais perguntas são mapeadas às SP's e aos seus níveis de maturidade, de forma que a resposta positiva ou negativa permite medir o estágio atual de maturidade da instituição, para cada prática de segurança. Se, para determinada SP e determinado nível, todas as respostas às perguntas forem afirmativas, a organização atingiu completamente aquele estágio, e assim por diante. Como as atividades das SP's são independentes entre si, por vezes a organização atingiu determinado nível, mas desempenha algumas atividades relacionadas a outros níveis superiores, sem, no entanto, atingí-los completamente. O modelo então prevê que, em tais casos, a atribuição de um nível intermediário seja representado pelo símbolo "+", como "0+", "1+" e assim por diante, sendo possível, inclusive, a avaliação de um nível "3+", quando a corporação desempenha práticas além do que é proposto pelo OpenSAMM.

Esta primeira fase (*lightweight*), que constitui-se de um panorama geral baseado em entrevista ou avaliação direta, é seguida por um trabalho adicional de verificação e auditoria (*detailed*), de forma a confirmar a execução das atividades de determinado nível de maturidade. Tal confirmação é realizada através da verificação do atingimento das métricas de sucesso sugeridas para cada estágio. Um ajuste na pontuação (ou nas respostas do questionário) é, então, se necessário, efetuado (Figura 2)

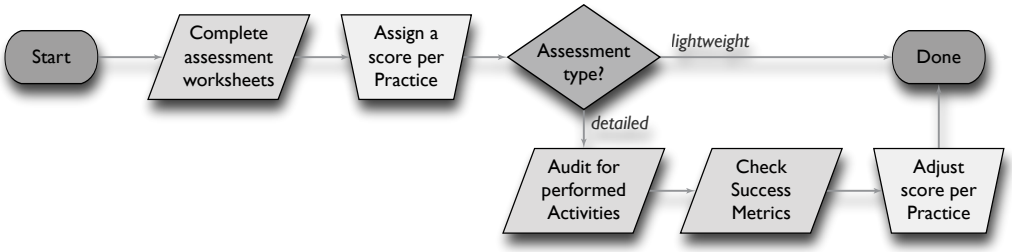

Figura 2: processo de avaliação (*assessment*) do OpenSAMM
(Fonte: CHANDRA, 2008 [5])

[13] O sítio do projeto disponibiliza uma planilha em formato XLS já configurada para a fase de avaliação (http://www.opensamm.org/downloads/resources/20090925-SAMM-Assessment-v0.4.xls).

De posse dessas avaliações, a organização pode apoiar-se, então, para implementar programas estruturados de melhoria de segurança de *software*. Tais programas consistem em linhas de atuação, divididas em fases (quantas forem necessárias) no tempo, de modo que a organização trace um objetivo planejado e desempenhe ações práticas que visem o aprimoramento da segurança de seus *softwares*, com o objetivo de atingir um nível maior de maturidade nas SP's ao final de cada fase (Figura 3). Configura-se, assim, o programa de melhoria e acreditação

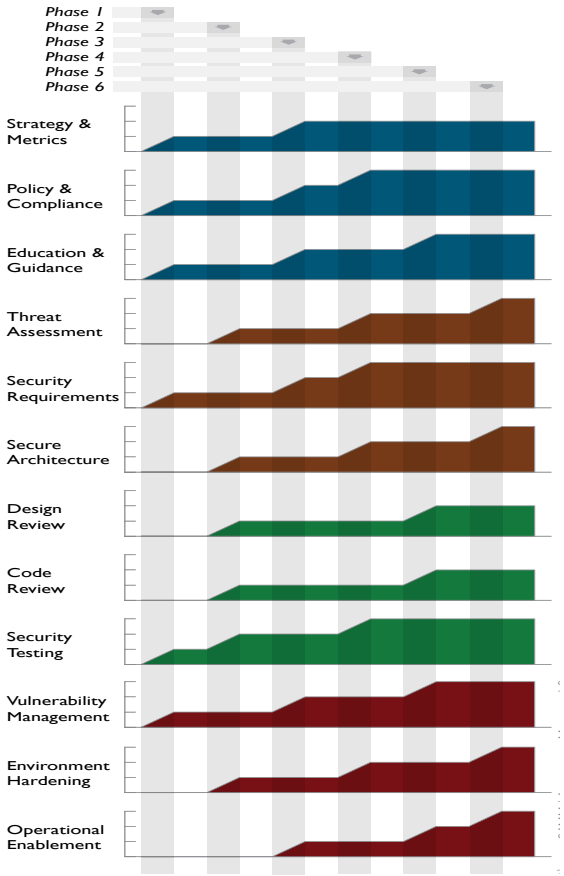

Figura 3: exemplo de um programa de melhoria e acreditação de *software* para órgãos governamentais, dividido em seis fases (linhas verticais). Cada mudança de patamar (ao final da fase) representa o aumento do nível de maturidade da prática de segurança correspondente. (Fonte: CHANDRA, 2008 [5], pág. 31)

de *software*: a organização inicia as fases de melhorias e trabalha para atingir os níveis de maturidade desejados, cumprindo as atividades e ações prescritas no modelo. No fim da fase, a linha de atuação e as pontuações são ajustadas, e a próxima fase começa.

O OpenSAMM sugere alguns modelos prontos de programas para implantação, de acordo com o tipo de organização. O modelo correspondente às

organizações governamentais (Figura 3) propõe dar ênfase inicial (ainda nas primeiras fases) ao conjunto de práticas das BF's de Governança e Verificação, sobretudo à prática de Testes de Segurança (ST). Os grandes riscos oferecidos pela exposição pública daquelas instituições, aliado ao fato da grande quantidade de *softwares* legados geralmente existentes, traz à luz a necessidade iminente de que, ao se implantar o programa, tais aplicações passem imediatamente por testes de segurança, de forma a salientar as vulnerabilidades existentes e permitir correções imediatas críticas. Nos casos em que o órgão governamental contrate o desenvolvimento de *software*, o modelo também sugere que a SP de Requisitos de Segurança (SR) seja envolvida nas primeiras fases do programa, de modo a inserir tais controles de segurança no *software* desenvolvido pela empresa contratada.

2.4 Aplicação do OpenSAMM na Administração Pública Federal - Resultados

Com o intuito de verificar a viabilidade de aplicação direta do modelo OpenSAMM em órgãos governamentais brasileiros, em aspectos gerais, buscou-se a criação de uma proposta de programa de melhoria e acreditação de *software* para uma Secretaria da administração pública federal direta. Esta Secretaria (doravante denominada por "Secretaria X") faz parte de um dos Ministérios federais e é responsável pelo desenvolvimento, manutenção e execução de um dos sistemas *web* estruturantes do Estado brasileiro, acessível via internet (inclusive via *web services*) e utilizado por todos os órgãos da APF e dos poderes da União, no apoio às atividades de gestão orçamentária. Tal sistema vem passando por uma reestruturação e reconsolidação geral desde 2009, incluindo seu desenvolvimento completo na linguagem J2EE e reforma de seu ambiente operacional, baseado em virtualização. Antes dessa data, o sistema era mantido e hospedado por outro órgão federal, e era baseado em tecnologias e linguagens antigas de terminal *mainframe*.

O órgão conta com equipe própria de TI, em sua maioria também reestruturada a partir de 2009, na forma de uma coordenação geral, composta por 90 pessoas, entre servidores públicos concursados e colaboradores contratados. A equipe é subdividida entre as áreas de desenvolvimento de aplicações (20 pessoas), infraestrutura (27 pessoas), escritório de projetos e governança (7 pessoas), apoio aos sistemas e informações gerenciais (14 pessoas), gestão do conhecimento e documentação (19) e administração e apoio (3).

O estudo foi realizado entre a última semana de agosto/2012 e a primeira semana de setembro/2012. O método utilizado foi a avaliação direta, por este autor (membro da equipe de infraestrutura de TI), da conjuntura de segurança de *software* existente no órgão, através da aplicação da planilha de avaliação do OpenSAMM, na versão anexa a este trabalho. Posteriormente, buscou-se a construção de uma sugestão de um plano simples de melhoria e acreditação de *software*, dividido em 4 fases, construído a partir do modelo fornecido pelo OpenSAMM[14], para ser implementado ao longo de 16 meses.

O quadro 1 mostra os níveis de maturidade em segurança de *software* atualmente verificados para o órgão em questão, divididos para cada SP. Não foram encontradas dificuldades substanciais na aplicação da planilha de avaliação, e esta se mostrou satisfatoriamente abrangente em relação ao que se espera das atividades envolvidas na segurança de *software*. As assertivas que geraram alguma dúvida, como as associadas à prática de Habilitação Operacional (OE) puderam ser rapidamente sanadas por meio de consulta direta ao detalhamento da prática no *framework* (vide [5], pág.32).

Dada a reestruturação recente da TI da Secretaria X, era esperado que diversas práticas relacionadas ao desenvolvimento seguro não encontrassem grandes pontuações de maturidade. Pelo menos o primeiro nível foi atingido (e suas respectivas métricas de sucesso) para as práticas relacionadas a Educação e Instrução (EG), Avaliação de Ameaças (TA), Testes de Segurança (ST) e Gestão de Vulnerabilidades (VM). Nos três últimos casos, o nível de maturidade verificado é corroborado, majoritariamente, pela presença de equipe dedicada e residente de segurança e resposta a incidentes, ainda que tal equipe seja restrita à equipe de infraestrutura.

O nível 2 de maturidade foi evidenciado na prática de Requisitos de Segurança (SR). Nesta, foram verificados requisitos de segurança na Secretaria X que, de acordo com o modelo, corroboram esse estágio de maturidade, tais como a preocupação com a segurança de dados (os bancos de dados possuem níveis específicos e controlados de acesso), controle de acesso à aplicação (existência, desde a concepção da aplicação, de perfis de acesso bem definidos, de acordo com o papel do usuário), integridade de transações, níveis adequados de disponibilidade

[14] O template para formatação visual do programa de melhoria é uma planilha XLS fornecida pelo sítio do projeto, disponível em <http://www.opensamm.org/downloads/resources/20090610-Samm-roadmap-chart-template.xls>.

Quadro 1: Planilha resultante da avaliação da Secretaria X

OpenSAMM — Planilha de Avaliação - Secretaria X

Funções de Negócio	Práticas de Segurança	Atividades	Resposta	Nível de Maturidade
Governança	Estratégia e Métrica	Já existe algum programa de melhoria e acreditação de segurança de *software*?	No	0+
		A maioria dos *stakeholders* de negócio já conhecem o perfil de risco de sua organização?	Yes	
		A maioria das equipes de desenvolvimento está ciente dos planos futuros do programa de melhoria e acreditação?	No	
		A maioria de suas aplicações e recursos são categorizados de acordo com o seu risco associado?	Yes	
		As avaliações de risco são utilizadas para moldar as ações de garantia de segurança?	Yes	
		A grande parte da organização sabe o que é necessário baseada em análise de risco?	No	
		São coletadas, por projeto, informações relacionadas aos custos das ações de garantia de segurança?	No	
		A organização compara regularmente os seus gastos/investimentos em segurança com outras organizações?	No	
	Política e Conformidade	A maioria dos stakeholders dos projetos de *software* estão cientes do nível de conformidade destes projetos?	No	0+
		Requisitos de conformidade são especificamente considerados pelas equipes de projetos?	Yes	
		A organização utiliza um conjunto de políticas e padrões para controlar o desenvolvimento de *software*?	No	
		As equipes de projetos são capazes de solicitar uma auditoria de conformidade com políticas e padrões?	No	
		Os projetos são periodicamente auditados, de forma a garantir uma linha base de conformidade com pol. e padrões?	No	
		A organização usa, sistematicamente, auditorias para coletar evidências de controles de conformidade?	No	
	Educação e Instrução	A maioria dos desenvolvedores já recebeu algum treinamento focado em segurança no desenvolvimento de *software*?	Yes	1+
		As equipes de projeto tem acesso a guias e boas práticas de desenvolvimento seguro?	Yes	
		A maioria dos papéis no ciclo de desenvolvimento recebeu treinamento e guia específico para aquele papel?	No	
		Os stakeholders estão habilitatos em solicitar consultores em segurança de software para uso em seus projetos?	No	
		A orientação em segurança é controlada de forma centralizada, mas consistentemente distribuída na organização?	Yes	
		Existe avaliação de pessoal que assegura um nível mínimo de competência para atuação em desenv. seguro?	No	
Construção	Avaliação de Ameaças	A maioria dos projetos em sua organização consideram e documentam possíveis ameaças?	Yes	1+
		A organização entende, reconhece e documenta os tipos possíveis de atacantes que pode enfrentar?	Yes	
		As equipes de projeto analisam os requisitos funcionais dos projetos em relação a possíveis casos de abuso?	No	
		As equipes de projeto utilizam métodos para avaliar e classificar ameaças, visando uma comparação relativa?	No	
		Os stakeholders estão cientes das ameaças relevantes e de sua classificação associada?	No	
		As equipes de projeto consideram os riscos específicos de *softwares* externos e de terceiros?	Yes	
		Todos os mecanismos de proteção e controle são mapeáveis às ameaças correspondentes?	No	
	Requisitos de Segurança	A maioria das equipes de projeto especificam requisitos de seguranca durante o desenvolvimento?	Yes	2
		As equipes de projeto extraem os requisitos de segurança de boas práticas e guias de conformidade?	Yes	
		A maioria dos stakeholders revisam matrizes de controle de acesso para projetos relevantes?	Yes	
		As equipes de projeto especificam requisitos baseadas no feedback de outras atividades de segurança?	Yes	
		A maioria dos stakeholders revisam acordos e contratos de *vendors* no tocante a requisitos de segurança?	No	
		Os requisitos de segurança especificados pelas equipes de projeto estão sendo auditados?	No	
	Arquitetura Segura	É fornecida uma lista de componentes recomendados de terceiros para as equipes de projeto?	No	0+
		A maioria das equipes de projeto estão cientes de princípios de design seguro e estão aplicando esses princípios?	Yes	
		Recursos compartilhados existentes de segurança, com guias de utilização, são promovidos para as equipes de proj.?	No	
		Design Patterns prescritivos de seg. baseados na arquitetura das aplicações dos projetos são providos para as equipes?	No	
		As equipes de projetos constróem aplicações baseadas em frameworks e plataformas controladas centralizadamente?	Yes	
		As equipes de projeto estão sendo auditadas no tocante ao uso de componentes arquiteturais de segurança?	No	
Verificação	Revisão de Design	As equipes de projeto documentam a superfície de ataque relacionado ao design de software?	No	0+
		As equipes de projeto verificam o design de software contra os riscos de segurança conhecidos?	Yes	
		A maioria das equipes de projeto analisam elementos do design especificamente em relação a mecanismos de seg.?	Yes	
		A maioria dos stakeholders sabe como obter ou está habilitada a solicitar uma revisão formal de design do software?	No	
		O processo de revisão formal de design incorpora análise de dados detalhada?	Yes	
		As rotinas de auditoria de projeto requerem a existência de linha de base de resultados de revisão de design?	No	
	Revisão de Código	A maioria das equipes de projeto possuem checklists de revisão de código baseadas em problemas conhecidos?	No	0+
		A maioria das equipes executam revisão de códigos selecionados como de alto risco, de forma geral?	Yes	
		A maioria das equipes de projeto tem acesso a ferramentas automatizadas de revisão e análise de código?	No	
		A maioria dos stakeholders requerem e revisam, de forma consistente, os resultados das revisões e análise de código?	No	
		A maioria das equipes de projeto utiliza a automação para checar o código de acordo com padrões requeridos de aplic.?	No	
		A auditoria rotineira de projetos requerem uma linha de base mínima de revisão antes de liberar o código para public.?	No	
	Testes de Segurança	Os projetos especificam alguns testes de segurança baseados nos requisitos existentes?	Yes	1+
		São executados testes de penetração na maioria dos projetos, antes da publicação?	Yes	
		A maioria dos stakeholders estão cientes dos resultados dos testes de segurança, antes da publicação?	Yes	
		Os projetos se utilizam de automação para avaliar os casos de teste de segurança?	No	
		São utilizados, na maioria dos projetos, processos consistentes de avaliação e comunicação dos testes de segurança?	Yes	
		Casos de testes de segurança são derivados da lógica específica da aplicação, de forma compreensiva?	No	
		As rotinas de auditoria estipulam resultados padrões mínimos para os testes de segurança?	No	
Implantação	Gestão de Vulnerabilidades	A maioria dos projetos tem um ponto de contato para questões e problemas de segurança?	Yes	1+
		A organização possui equipe própria designada para o tratamento e resposta às questões de segurança?	Yes	
		A maioria das equipes de projeto estão cientes do(s) ponto(s) de contato de segurança e equipe(s) de resposta?	Yes	
		A organização utiliza processo consistente para o tratamento e comunicação de incidentes?	No	
		A maioria dos stakeholders são comunicados a respeito das questões de segurança de seus projetos de software?	Yes	
		A maioria dos incidentes são inspecionados até a sua causa raiz, de forma a gerar recomendações subsequentes?	No	
		A maioria dos projetos coletam e comunicam dados e métricas relacioanadas a incidentes, de forma consistente?	No	
	Segurança do Ambiente	A maioria dos projetos documentam algum requisito de segurança para o ambiente operacional?	No	0+
		A maioria dos projetos verificam a atualização de segurança dos componentes de software de terceiros?	No	
		É utilizado um processo consistente na aplicação de patches e atualizações para os componentes críticos?	Yes	
		A maioria dos projetos usam a automação para verificar a saúde da aplicação e do ambiente operacional?	Yes	
		Os stakeholders estão cientes a respeito de ferramentas adicionais para proteção de software no ambiente operacional?	No	
		Existe rotina de auditoria para checagem das linhas de base de saúde operacional da maioria dos projetos ?	No	
	Habilitação Operacional	São divulgadas notas de segurança juntamente à maioria das publicações de software?	No	0+
		Alertas e condições de erros relacionados à segurança são documentados para a maioria dos projetos?	Yes	
		A maioria dos projetos utiliza um processo de gerenciamento de mudanças conhecido e entendido pelos envolvidos?	Yes	
		As equipes de projeto entregam algum guia de segurança operacional juntamente a cada publicação de aplicação?	No	
		A maioria dos projetos estão sendo auditados, no tocante à existência de informações apropriadas de seg. Operacional?	No	
		É executada rotineiramente a assinatura do código dos componentes, segundo um processo consistente?	No	

e identificação de processos críticos de negócio. Verificou-se também a existência de testes efetuados no sistema pela equipe dedicada à testes de aplicação, de forma a verificar e controlar as matrizes de controles de acesso (verificar as

permissões de perfis de cada usuário, de acordo com seu papel no sistema). Para as outras práticas, foram verificadas iniciativas pontuais e *ad hoc* que, apesar de efetivas na maioria dos casos, não constituem, compreensivamente, níveis completos de maturidade (e as métricas de sucesso não foram atingidas). Dentre estas iniciativas, destacam-se: minuta de normativo, em fase final de aprovação, que regulamenta a gestão e controle de acessos dos usuários ao sistema estruturante; curso prático ministrado em codificação segura para a maioria dos programadores; política de segurança da informação e comunicações aprovada e publicada e normativo interno regulamentador do processo formal de publicação de novas versões do sistema, norma esta que disciplina a necessidade obrigatória de testes de performance e segurança.

De posse desta avaliação inicial, foi possível desenhar um plano geral de melhoria e acreditação de segurança de *software* para a Secretaria X (Figura 4), que contempla ações a serem realizadas nos próximos 16 meses. O plano leva em consideração a sugestão do modelo proposto pelo OpenSAMM para as organizações governamentais. Dado o fraco desempenho observado na prática de Política e Conformidade (PC), sugere-se a priorização das atividades relacionadas já na primeira fase, uma vez que tal prática é crítica para órgãos de Estado, haja vista, sobretudo, a necessidade de cumprimento da legislação vigente e a atuação dos órgãos de controle, como o TCU. Além disso, o órgão conta com política de segurança própria (que demanda a conformidade com suas diretrizes) e, em breve, legislação específica de Segurança de Software será publicada pelo GSI/PR, jurisdicionada a todos os órgãos da APF.

Destaca-se também a priorização das prática de Revisão de Código (CR) e Avaliação de Ameaças (TM) já na primeira fase. A reestruturação do sistema se deu de forma muito rápida, dada as necessidades urgentes de negócio, e milhares de linhas de código foram produzidas sem que houvesse um processo formal de verificação, revisão e documentação. Em meio a esse cenário, torna-se essencial adotar medidas que permitam a revisão desse código, de forma a identificar e gerenciar possíveis ameaças existentes no *software* em si (ferramentas automatizadas de análise estática de código e dinâmica já estão sendo prospectadas para aquisição pela Secretaria X). Por fim, buscou-se criar um programa com prazo suficiente para assimilação eficiente das atividades de melhoria

de segurança – a duração de quatro meses de cada fase busca cobrir eventuais prazos licitatórios, férias de equipes, desenhos de processos de negócio e duração de atividades de ensino e capacitação de pessoal.

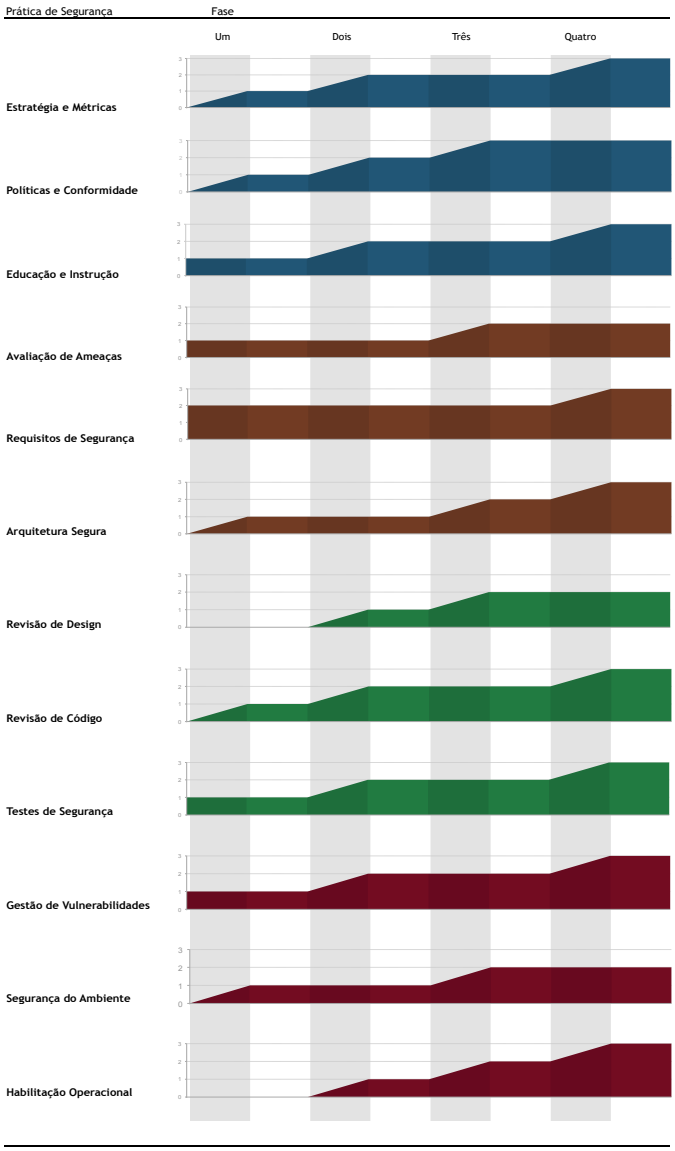

Figura 4: Plano de melhoria e acreditação de segurança de *software* da Secretaria X. Cada fase dura 4 meses. (Fonte: vide nota 14 – adaptado)

Documentação adequada e detalhamento de cada fase do programa devem, logicamente, acompanhar esta sugestão de plano de melhoria, e devem passar pelo crivo e aprovação das chefias envolvidas, fatos que extrapolam o escopo deste artigo. Como escopo de trabalho futuro, sugere-se o estudo comparativo entre a avaliação observada neste artigo e a avaliação futura após a implementação do

plano de melhorias, de forma a analisar os ganhos reais e oportunidades de melhorias.

3 CONCLUSÃO

Conclui-se, a partir dos estudos deste trabalho, que a aplicação direta do modelo proposto pelo OpenSAMM é factível e viável para o estabelecimento de um programa de melhoria e acreditação de segurança de *software* em um órgão da APF. Consistindo-se o *framework* de um modelo aberto, adaptável, livre e prescritivo (e não um modelo fechado de certificação), foi possível criar uma proposta básica de programa de melhoria onde, mesmo levando-se em conta as sugestões do OpenSAMM, suas fases foram adaptadas e priorizadas de acordo com as necessidades da Secretaria X, para que as práticas de segurança mais deficitárias sejam “atacadas” primeiro do que as outras. O *framework* também mostrou-se suficiente nos artefatos básicos de apoio à construção do programa, como a planilha de avaliação, considerada satisfatoriamente abrangente em relação às atividades envolvidas no desenvolvimento de *software*, e os detalhamentos prescritivos dos níveis de maturidade, com sugestões bem definidas de métricas de sucesso.

A realidade de TI da Secretaria X pode ser observada em algumas dezenas de órgãos federais, como Banco Central e Controladoria Geral da União, e em centenas de órgãos das esferas estadual e municipal. Com a exposição cada vez maior de tais entidades na Internet, através de suas aplicações *web*, portais e afins, os riscos associados aos seus *softwares* aumentam substancialmente. Carecendo esses órgãos de formas viáveis e estruturadas de melhoria de segurança de *software*, além de ações (muitas vezes urgentes) de melhoria da segurança da informação como um todo, o Software Security Maturity Model – OpenSAMM – mostra-se uma ferramenta valiosa e uma alternativa viável para o atingimento desses objetivos.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] MANDARINO JUNIOR, Raphael. **Segurança e Defesa do Espaço Cibernético Brasileiro**. Recife: Cubzac, 2010. 182p.

[2] DCSINT – US Army Deputy Chief of Staff for Intelligence. **Cyber Operations and Cyber Terrorism, Handbook Number 1.02**. Ago. 2005. Disponível em <http://www.au.af.mil/au/awc/awcgate/army/guidterr/sup2.pdf>. Acesso em Ago. 2012.

[3] GSI/PR – Gabinete de Segurança Institucional da Presidência da República. **Guia de Referência para a Segurança das Infraestruturas Críticas da Informação.** Nov, 2010. Disponível em <http://dsic.planalto.gov.br/documentos/publicacoes/2_Guia_SICI.pdf>. Acesso em ago/2012.

[4] CINTRÃO, Luciano Pezza – **Apostila do Curso: "Sistemas Integrados de Informação para a Gestão Governamental"**. Escola Nacional de Administração Pública – ENAP, 2008, p.19. Curso atendido na ENAP em 17,18 e 19 out. 2011.

[5] CHANDRA, Pravir – **Software Assurance Maturity Model (SAMM).** Ago, 2008, mantido pelo projeto *The Open Software Assurance Maturity Model Project* – OpenSAMM. Disponível em <http://www.opensamm.org>. Acesso em jun/2012.

[6] OWASP – **The Open Web Application Security Project**. Disponível em <http://www.owasp.org>. Acesso em jun/2012.

[7] VERIZON – **2012 Data Breach Investigation Report**. Mar, 2012. Disponível em < http://www.verizonbusiness.com/resources/reports/rp_data-breach-investigations-report-2012_en_xg.pdf>. Acesso em jul/2012.

[8] SYMANTEC – **Internet Security Threat Report, Volume 17**. Abr, 2012. Disponível em <http://www.symantec.com/threatreport/topic.jsp?id=threat_activity_ trends&aid=malicious_activity_by_source>. Acesso em ago/2012.

[9] CTIR.Gov – **Estatísticas de Tratamento de Incidentes de Rede na APF**. Jul, 2012. Disponível em <http://www.ctir.gov.br/arquivos/estatisticas/2012/Estatisticas_ CTIR_Gov_1o_Semestre_2012.pdf>. Acesso em set/2012.

[10] VERACODE – **State of Software Security Report, Volume 4**. Dez, 2011. Disponível em <http://info.veracode.com/rs/veracode/images/VERACODE-SOSS-V4.PDF>. Acesso em jul/2012.

[11] NIST – **NIST Special Publication 800-53 Rev. 3**. Jul, 2009. Disponível em <http://www.nist.gov/manuscript-publication-search.cfm?pub_id=903280>. Acesso em jul/2012.

[11] SANS-CSIS – **20 Critical Security Controls v3.1**. Out, 2011. Disponível em <http://www.sans.org/critical-security-controls/>. Acesso em jul/2012.

[12] MS-SDL – **Microsoft Security Development Lifecycle**. Disponível em < http://www.microsoft.com/security/sdl/default.aspx>. Acesso em jun/2012.

[13] CLASP-CHANDRA – **Comprehensive, Lightweight Application Security Process**. Jun, 2007. Disponível em < https://www.owasp.org/index.php/Category: OWASP_CLASP_Project>. Acesso em jun/2012.

[14] CIGITAL – **Cigital Inc. Products**. 2012. Disponível em < http://www.cigital.com/products>. Acesso em jun/2012.

[15] OWASP-Top10 – **The OWASP Top 10 Project**. Abr, 2010. Disponível em < https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project>. Acesso em jun/2012.

[16] CWE-SANS-Top25 – **2011 CWE/SANS Top 25 Most Dangerous Software Errors**. Abr, 2010. Disponível em <http://www.sans.org/top25-software-errors/>. Acesso em jun/2012.

[17] MCGRAW, Gary, Ph.D., MIGUES, Sammy, WEST,Jacob – **Building Security In Maturity Model 4**. Set, 2012. Disponível em <http://bsimm.com>. Acesso em set/2012.

[18] SIMÃO, Márcia, BARCELOS, Roberta – **Manual para Elaboração de Trabalhos Acadêmicos**. Universidade Salgado de Oliveira – Pró-reitoria de Pós-Graduação e Pesquisa, Niterói, 2005.

[19] TCU (Tribunal de Contas da União) – **Relatório do Levantamento de Governança de TI 2010**. Ago, 2010. Disponível em <http://portal2.tcu.gov.br/portal/page/portal/TCU/comunidades/tecnologia_informacao/pesquisas_governanca/Relatório%20do%20Levantamento%20Governança%20de%20TI%202010.pdf>. Acesso em out/2012.

ANEXO

Planilha de Avaliação proposta pelo OpenSAMM* (adaptada**)

OpenSAMM — Planilha de Avaliação

Funções de Negócio	Práticas de Segurança	Atividades	Resposta	Nível de Maturidade
Governança	Estratégia e Métrica	Já existe algum programa de melhoria e acreditação de software?	Yes	0+
		A maioria dos *stakeholders* de negócio já conhecem o perfil de risco de sua organização?	Yes	
		A maioria da equipe de desenvolvimento está ciente dos planos futuros do programa de melhoria e acreditação?	No	
		A maioria de suas aplicações e recursos são categorizados de acordo com o seu risco associado?	Yes	
		As avaliações de risco são utilizadas para moldar as ações de garantia de segurança?	No	
		A grande parte da organização sabe o que é necessário baseada em análise de risco?	No	
		São coletadas, por projeto, informações relacionadas aos custos das ações de garantia de segurança?	No	
		A organização compara regularmente os seus gastos/investimentos em segurança com outras organizações?	Yes	
	Política e Conformidade	A maioria dos stakeholders dos projetos sabem o nível de conformidade destes projetos?		0
		Requisitos de conformidade são especificamente considerados pelas equipes de projetos?		
		A organização utiliza um conjunto de políticas e padrões para controlar o desenvolvimento de *software*?		
		As equipes de projetos são capazes de solicitar uma auditoria de conformidade com políticas e padrões?		
		Os projetos são periodicamente auditados, de forma a garantir uma linha base de conformidade com pol. e padrões?		
		A organização usa, sistematicamente, auditorias para coletar evidências de controles de conformidade?		
	Educação e Instrução	Os desenvolvedores já receberam algum treinamento focado em segurança no desenvolvimento de *software*?		0
		As equipes de projeto tem acesso a guias e boas práticas de desenvolvimento seguro?		
		A maioria dos papéis no ciclo de desenvolvimento recebeu treinamento e guia específico para aquele papel?		
		Os stakeholders são capazes de solicitar consultores em segurança de software para uso em seus projetos?		
		A orientação em segurança é controlada de forma centralizada, mas consistentemente distribuída na organização?		
		Existe avaliação de pessoal que assegura um nível mínimo de competência para atuação em desenv. seguro?		
Construção	Avaliação de Ameaças	A maioria dos projetos em sua organização consideram e documentam possíveis ameaças?		0
		A organização entende, reconhece e documenta os tipos possíveis de atacantes que pode enfrentar?		
		As equipes de projeto analisam os requisitos funcionais dos projetos em relação a possíveis casos de abuso?		
		As equipes de projeto utilizam métodos para avaliar e classificar ameaças, visando uma comparação relativa?		
		Os stakeholders estão cientes das ameaças relevantes e de sua classificação associada?		
		As equipes de projeto consideram os riscos específicos de *softwares* externos e de terceiros?		
		Todos os mecanismos de proteção e controle são mapeáveis às ameaças correspondentes?		
	Requisitos de Segurança	A maioria das equipes de projeto especificam requisitos de seguranca durante o desenvolvimento?	Yes	2
		As equipes de projeto extraem os requisitos de segurança de boas práticas e guias de conformidade?	Yes	
		A maioria dos stakeholders revisam matrizes de controle de acesso para projetos relevantes?	Yes	
		As equipes de projeto especificam requisitos baseadas no feedback de outras atividades de segurança?	Yes	
		A maioria dos stakeholders revisam acordos e contratos de *vendors* no tocante a requisitos de segurança?	No	
		Os requisitos de segurança especificados pelas equipes de projeto estão sendo auditados?	No	
	Arquitetura Segura	É fornecida uma lista de componentes recomendados de terceiros para as equipes de projeto?	No	0+
		A maioria das equipes de projeto estão cientes de princípios de design seguro e estão aplicando esses princípios?	Yes	
		Recursos compartilhados existentes de segurança, com guias de utilização, são promovidos para as equipes de proj.?		
		Design Patterns prescritivos baseados na arquitetura das aplicações dos projetos são providos para as equipes?		
		As equipes de projetos constróem aplicações baseadas em frameworks e plataformas controladas centralizadamente?		
		As equipes de projeto estão sendo auditadas no tocante ao uso de componentes arquiteturais de segurança?		
Verificação	Revisão de Design	As equipes de projeto documentam o perímetro de ataque ao design de software?		0
		As equipes de projeto verificam o design de software contra os riscos de segurança conhecidos?		
		A maioria das equipes de projeto analisam elementos do design especificamente em relação a mecanismos de seg.?		
		A maioria dos stakeholders sabe como obter ou solicitar uma revisão formal de design do software?		
		O processo de revisão formal de design incorpora análise de dados detalhada?		
		As rotinas de auditoria de projeto requerem a existência de linha de base de resultados de revisão de design?		
	Revisão de Código	A maioria das equipes de projeto possuem checklists de revisão de código baseadas em problemas conhecidos?		0
		A maioria das equipes executam revisão de códigos selecionados como de alto risco, de forma geral?		
		A maioria das equipes de projeto tem acesso a ferramentas automatizadas de revisão e análise de código?	No	
		A maioria dos stakeholders requerem e revisam, de forma consistente, os resultados das revisões e análise de código?	No	
		A maioria das equipes de projeto utiliza a automação para checar o código de acordo com padrões requeridos de aplic.?		
		A auditoria rotineira de projetos requerem uma linha de base mínima de revisão antes de liberar o código para public.?		
	Testes de Segurança	Os projetos especificam alguns testes de segurança baseada nos requisitos existentes?		0+
		São executados testes de penetração na maioria dos projetos, antes da publicação?	Yes	
		A maioria dos stakeholders estão cientes dos resultados dos testes de segurança, antes da publicação?		
		Os projetos se utilizam de automação para avaliar os casos de teste de segurança?		
		São utilizados, na maioria dos projetos, processos consistentes de avaliação e comunicação dos testes de segurança?		
		Casos de testes de segurança são derivados da lógica específica da aplicação, de forma compreensiva?		
		As rotinas de auditoria determinam resultados padrões mínimos para os testes de segurança?		
Implantação	Gestão de Vulnerabilidades	A maioria dos projetos tem um ponto de contato para questões e problemas de segurança?	Yes	1
		A organização possui equipe própria designada para o tratamento e resposta às questões de segurança?	Yes	
		A maioria das equipes de projeto estão cientes do(s) ponto(s) de contato de segurança e equipe(s) de resposta?	Yes	
		A organização utiliza processo consistente para o tratamento e comunicação de incidentes?	No	
		A maioria dos stakeholders são comunicados a respeito das questões de segurança de seus projetos de software?		
		A maioria dos incidentes são inspecionados até a sua causa raiz, de forma a gerar recomendações subsequentes?		
		A maioria dos projetos coletam e comunicam dados e métricas relacioanadas a incidentes, de forma consistente?		
	Segurança do Ambiente	A maioria dos projetos documentam algum requisito de segurança para o ambiente operacional?		0+
		A maioria dos projetos verificam a atualização de segurança dos componentes de software de terceiros?		
		É utilizado um processo consistente na aplicação de patches e atualizações para os componentes críticos?	Yes	
		A maioria dos projetos usam a automação para verificar a saúde da aplicação e do ambiente operacional?	No	
		Os stakeholders estão cientes a respeito de ferramentas adicionais para proteção de software no ambiente operacional?		
		Existe rotina de auditoria para checagem das linhas de base de saúde operacional da maioria dos projetos ?		
	Habilitação Operacional	São divulgadas notas de segurança juntamente à maioria das publicações de software?		0
		Alertas e condições de erros relacionados à segurança são documentados para a maioria dos projetos?		
		A maioria dos projetos utiliza um processo de gerenciamento de mudanças conhecido e entendido pelos envolvidos?		
		As equipes de projeto entregam algum guia de segurança operacional juntamente a cada publicação de aplicação?		
		A maioria dos projetos estão sendo auditados, no tocante à existência de informações apropriadas de seg. Operacional?		
		É executada rotineiramente a assinatura do código dos componentes, segundo um processo consistente?		

** a coluna "Nível de Maturidade" apresenta os valores 0, 0+, 1, 1+, 2, 2+ e 3 baseada nas respostas afirmativas da coluna "Resposta".*

*** a planilha foi traduzida por este autor.*