Governo do Estado de Minas Gerais
Secretaria de Estado de Planejamento e Gestão
Subsecretaria de Gestão
Superintendência Central de Governança Eletrônica

Manual de Desenvolvimento e Aquisição de Sistemas Seguros

Volume 1
Requerimentos de segurança para desenvolvimento e validação de sistemas

Desenvolvidor por:

ERNST & YOUNG
Quality In Everything We Do

Renata Maria Paes de Vilhena
Secretária de Estado de Planejamento e Gestão

Eurico Bitencourt Neto
Secretário-Adjunto

Frederico César Silva Melo
Subsecretário de Gestão

Adriano Otávio Rocha Teixeira
Diretor Central de Gestão de Recursos de TIC

Carine Alves
Gerente Integrada de Riscos e Sistemas Empresariais - PRODEMGE

Leonardo Bruno Possa Andrade
Gerente Projeto

Sumário

Lista de Tabelas

Lista de Figuras

1. Introdução

1.1. Sobre o manual

Atendendo a requisição da Secretaria de Estado de Planejamento e Gestão de Minas Gerais, a Ernst & Young redigiu este manual com base em pesquisas e em sua experiência de mercado. Seu principal objetivo é compor uma referência única para os requerimentos de segurança e validação dos sistemas desenvolvidos e adquiridos para órgãos e entidades públicas de Minas Gerais. Ao longo do documento, são fornecidas diretrizes e exemplos para o estabelecimento de controles com base em padrões de mercado como:

- CobiT 4.1
- CMMi 1.2 para desenvolvimento
- ISO 27002, com ênfase nos itens de segurança de aplicações
- ISO 15408-2, para requerimentos funcionais de segurança
- ISO 15408-3, para as definições de avaliação de segurança e maturidade de sistemas
- Publicações especiais (SPs) 800 do NIST, que fornecem exemplos e práticas
- FIPS 199 e 200 do NIST, para a classificação de sistemas
- e-PING, que define os padrões de interoperabilidade do governo eletrônico
- Documentos de entidades não governamentais especializadas, como ISACA e OWASP

Há dois volumes do manual, sendo:

- Volume 1: Requerimentos de segurança para desenvolvimento e validação;
- Volume 2: Diretivas para codificação segura e interoperabilidade de sistemas.

1.2. Motivadores

Tradicionalmente, a segurança da informação tem como foco a implantação de sistemas de controle de acesso global, como firewalls e servidores de autenticação. A sofisticação dos sistemas sua utilização em modelo distribuído, tem direcionado ataques para as aplicações, que têm requerido maior segurança.

Dentre as ameaças às aplicações, destacam-se:

Código		Ameaça	Descrição	C	D	I
T	1	Roubo ou vazamento de informações	O atacante pode fazer uso de informações relevantes que não foram devidamente protegidas ou que, por algum erro, foram mostradas na tela do usuário	x		
T	2	Engenharia social	O atacante utiliza sua influência para descobrir informações relevantes que podem facilitar a descoberta de senhas ou configurações do sistema	x		
T	3	Ataques de injeção	Injeção de instruções ou códigos maliciosos em campos de entradas de dados, forçando o sistema a executar comandos do atacante	x		x
T	4	Cross-Site Scripting (XSS)	Uso de aplicações web para envio de códigos maliciosos a serem executados no browser de diferentes usuários finais, podendo acessar cookies e outras informações retidas pelo mesmo, além da possibilidade de alteração do conteúdo da página web	x		x
T	5	Roubo de sessões	Técnicas para utilização indevida de sessões de outros usuários autenticados no sistema. Quando este ataque é bem sucedido, o atacante adquire todos os privilégios do usuário que teve a sessão roubada	x		x
T	6	Cross-Site Request Forgery	Envio de links de páginas com códigos maliciosos que, se clicados pelo	x		

			destinatário, executam ações a favor do atacante			
T	7	Injeção de SQL	Injeção de códigos SQL maliciosos em campos de entradas de dados dos usuários. A execução destes códigos gera consultas no banco de dados, revelando informações ao atacante	x		x
T	8	Estouro de buffer	Ataques que reescrevem fragmentos de memória do sistema, podendo interromper a execução do sistema de forma inesperada. Tais ataques aproveitam-se de vulnerabilidades em entradas de dados de usuários		x	x
T	9	Execução remota	Tomada do controle direto do sistema, através da inserção de scripts maliciosos em interfaces de texto vulneráveis	x	x	x
T	10	Interrupção do sistema	Interrupção inesperada do sistema, causada por ataques ou pela sobrecarga de recursos não monitorados do sistema		x	
T	11	Spam	Utilização de e-mail para envio de mensagens com links maliciosos a diversos usuários	x	x	
T	12	Aferição de sucesso em ataques	Uso de informações geradas pelo próprio sistema para aferir se um ataque teve sucesso ou causou algum impacto. Mensagens de erro ou de tempo de execução são comumente usadas por atacantes para medir o sucesso de seus ataques	x	x	x
T	13	Ataques de robôs	Uso de robôs para envio de dados através de formulários ou para testes de senhas de acesso	x	x	x
T	14	Ataques de Força Bruta	Técnicas utilizadas para a descoberta de senhas e para burlar controles de acesso, com base na tentativa e erro	x		

Tabela 1 - Ameaças e seus impactos à Confidencialidade (C), Disponibilidade (D), Integridade (I).

Essas ameaças revelam um perfil emergente de atacantes, com foco na obtenção de informações confidenciais e na manipulação de sistemas, não somente na sua interrupção.

Um sistema governamental que apresenta falhas pode ter como consequências:

- Publicidade negativa;
- Investigações e aplicações legais;
- Perda de reputação;
- Perda da confiança do cidadão.

Naturalmente, há perdas financeiras decorrentes das falhas, com exemplos recentes:

- Dados de concursos, arrecadação, licitações e contratos, bem como informações de pessoas físicas fazem parte dos sistemas da informação de qualquer governo. Seu vazamento pode gerar custos por atrasar processos ou trazer ações judiciais.
    - Além dos sistemas, é importante ter atenção às operações e ao ambiente como um todo. O caso da fraude da prova do ENEM em 2009, por exemplo, obrigou o MEC a refazer contratos e imprimir novas provas gerando prejuízo de aproximadamente R$ 40 milhões (UOL, 2009);
- Para a segurança pública, há suspeitas de que informações privilegiadas reduziram o sucesso de operações policiais (IG, 2009);
- Os “mortos-vivos” da previdência social já consumiram cerca de R$ 1,67 bilhão em benefícios concedidos a segurados falecidos (Vaz, 2009), que

poderiam ser evitados caso houvesse integração entre os sistemas de cartórios e do INSS;

- Há polêmica quanto à compra de caças para o Brasil, em uma disputa entre fabricantes dos EUA, França e Suécia, que está sendo marcada pelo vazamento de informações;
- Sistemas inseguros podem ferir a legislação e atrapalhar investigações, como foi o caso de suspeita de “queima de arquivo público”, ao serem detectadas subtrações dos arquivos de segurança na Câmara dos Deputados (Agência Brasil, 2009).

1.3. O caminho

Ao contrário do pensamento comum, a segurança de aplicações não está embasada em controles técnicos, mas sim em metodologias e planejamento. A principal ferramenta para alcançar a segurança é a adoção de uma metodologia de Ciclo de Vida de Sistemas (SDLC, do inglês: Systems Development Life Cycle). Há diversos formatos de SDLCs, sendo um dos mais difundidos constituído pelas seguintes etapas:

Fases do SDLC	**Características**
Fase 1 - Concepção	É manifestada a necessidade de um sistema de TI e a proposta e o escopo são documentados.
Fase 2 - Elaboração	O sistema de TI é comprado ou desenvolvido.
Fase 3 - Transição (Implantação)	As características de segurança do sistema deverão estar configuradas, habilitadas, testadas e verificadas.
Fase 4 - Operação e Manutenção	O sistema está em funcionamento. Geralmente estão em constante mudança devido a alterações de hardware, software, políticas e processos.
Fase 5 – Desativação (término)	Essa fase pode envolver a disposição de informação, hardware e software. Atividades podem incluir transferência, arquivamento, descarte ou destruição da informação para eliminar evidências do hardware e do software.

Tabela 2 - Fases do SDLC e suas características.

Este manual foi desenvolvido com base no Processo de Desenvolvimento de Software Orientado a Objeto (PDSOO), a metodologia de desenvolvimento da Prodemge (Companhia de Tecnologia da Informação do Estado de Minas Gerais). Considerando outros fornecedores de sistemas, foi realizado o alinhamento das atividades de segurança com o padrão ANSI/IEEE 1074, o que confere portabilidade do manual para outras metodologias.

Embora os SDLCs tradicionais não abordem tópicos específicos de segurança, cabe ressaltar o ganho na maturidade do desenvolvimento e natural redução da superfície de ataque dos sistemas desenvolvidos.

Durante a fase de desenho, alguns princípios devem ser seguidos para o desenvolvimento ou aquisição de sistemas seguros:

Princípio	**O que é?**	**Exemplo**
Economia de mecanismos	Manutenção do desenho simples e menos complexo	Código modular, objetos compartilhados, e serviços centralizados
Padrões de falhas seguras	Acesso negado por padrão, e garantido explicitamente	Transação negada
Intervenção completa	Checagem de permissão a cada vez que for requisitado o acesso a objetos	Controles de autenticação e autorização
Desenho aberto	O desenho não é secreto, mas a implemetação da salvaguarda sim	Algoritmos criptográficos
Separação de privilégios	Mais de uma condição é requerida para completar uma tarefa	Chaves divididas, funções em compartimentos
Mínimo de privilégios	Direitos mínimos e acesso a usuários explicitamente concedidos	Restrição de contas administrativas
Mínimo de mecanismos comuns	Mecanismos comuns para mais de um usuário/processo/papel não são compartilhados	Funções e bibliotecas dinâmicas baseadas em papéis
Aceitabilidade psicológica	Mecanismo de proteção de segurança que facilite o uso e a aceitação pelo usuário final	Diálogos de ajuda, ícones visualmente atraentes

Tabela 3 - Princípios de segurança.

A segurança deve ser incorporada desde a fase inicial do SDLC, o que reduz consideravelmente o custo de desenvolvimento de sistemas seguros. Estudos conduzidos pela IBM (IBM, 2008) mostram que o custo para a correção de defeitos de sistemas em produção pode ser cerca de 30 vezes mais caro do que quando o defeito é identificado na concepção, conforme apresenta a tabela:

Tipo de Erro	Concepção	Elaboração	Transição	Operação	Manutenção
Desenho	1x	5x	10x	15x	30x
Codificação		1x	10x	20x	30x
Implantação			1x	10x	20x

Tabela 4 - Custo relativo da correção de erros, baseado no tempo de descoberta.

Para aumentar a segurança dos sistemas, podem ser definidos processos de segurança no SDLC e estabelecidas funções de segurança, implantadas tecnologicamente. De um modo geral, há três objetivos na adoção de segurança:

- Proteção do ambiente de desenvolvimento;
- Proteção da aplicação desenvolvida;
- Validação de segurança na aplicação desenvolvida.

Seguindo esses três objetivos, foi empregado um *framework* de segurança no desenvolvimento de sistemas (Mellado, Fernández-Medina, & Piattini, 2007):

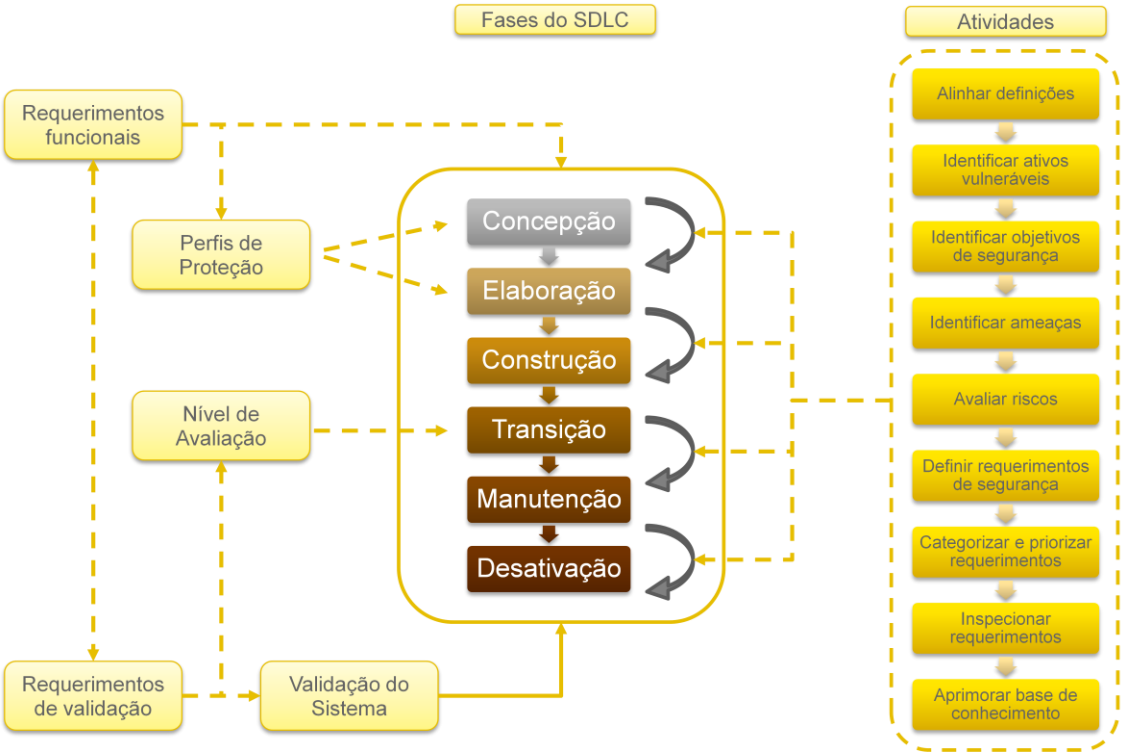


Figura 1 - *Framework* de segurança de sistemas.

- A partir de um nível de segurança estipulado com a caracterização do sistema, são estabelecidos:
  - Perfis de proteção, que serão implementados no sistema, principalmente nas fases de concepção e elaboração;
  - Objetivos de validação, que verificarão se o sistema faz o que diz fazer, com ponto crucial na fase de transição (implantação).
- Os perfis de proteção e os níveis de validação são relacionados de forma direta, ou seja, um sistema terá os requerimentos funcionais correspondentes aos níveis de validação definidos

Durante todas as fases do SDLC há atividades de segurança, como a análise de riscos, a revisão dos requisitos, e o gerenciamento do conhecimento.

Para obter a maior proteção dos sistemas, deve ser considerada a proteção em camadas (Graff & Wyk, 2003):

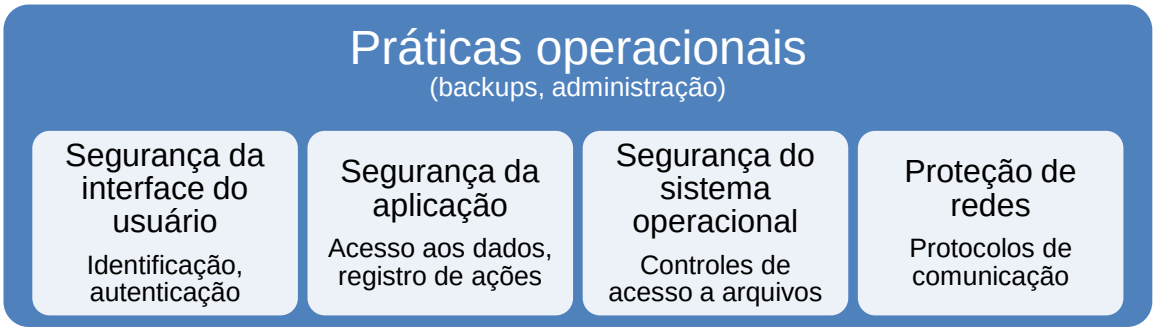


Figura 2 - Panorama de um sistema seguro.

Além disso, todo profissional de segurança da informação deve adotar as seguintes práticas:

#	**Melhores práticas para profissionais de segurança da informação**
1	Proteger a marca na qual seus clientes confiam
2	Conhecer seu negócio e suportá-lo com soluções seguras
3	Entender a tecnologia do software
4	Garantir conformidade com governança, regulações e privacidade
5	Conhecer os princípios básicos de segurança de software
6	Garantir a proteção de informações sensíveis
7	Desenhar software com características seguras
8	Desenvolver software com características seguras
9	Implantar software com características seguras
10	Educar a si mesmo e aos demais sobre como construir software seguro

Tabela 5 - Melhores práticas de um profissional de segurança.

1.4. Como utilizar este manual

1.4.1. Público alvo

Gestores, desenvolvedores e avaliadores de sistemas para órgãos e entidades governamentais de Minas Gerais.

1.4.2. Limitações

O manual não contempla todas as ferramentas ou emprega todos os controles de segurança disponíveis no mercado. Seu uso fundamental é para estabelecer um conjunto mínimo de padrões de segurança, que deverão ser estabelecidos com controles técnicos definidos pelos desenvolvedores ou compradores.

1.4.3. Procedimentos

A forma mais indicada de utilizar o manual é ter como referência o guia rápido de desenvolvimento seguro, em conjunto com o plano de segurança (a ser preenchido durante todo o desenvolvimento), disponíveis nos anexos deste volume.

1.4.4. Aquisição de sistemas seguros

Embora o manual seja voltado para o desenvolvimento de sistemas seguros, seu uso poderá ser estendido para o estabelecimento de requerimentos e validação de sistemas adquiridos. Nesses casos, os gestores terão uma ferramenta para garantir que as aplicações obtidas no mercado mantêm os mesmos níveis de segurança que as desenvolvidas com os requerimentos aqui contidos.

Um sistema adquirido deverá atender aos requerimentos de seu nível de segurança, estipulado pelo comprador, em acordo com o fornecedor. Portanto, deverão ser fornecidos os documentos necessários para a avaliação do sistema, conforme apresentado no capítulo 4.

2. Segurança no SDLC

O desenvolvimento de sistemas deve incorporar atividades de segurança para proteger informações e processos de negócio. A abordagem pela avaliação de riscos favorece o equilíbrio contínuo entre a segurança, os custos dos controles e as medidas de mitigação através do ciclo de vida do desenvolvimento de sistemas.

Os sistemas desenvolvidos deverão ser abordados sob a ótica da gestão de riscos. As principais atividades de segurança nas fases do SDLC são (NIST, 2008):

Fases	Ciclo de vida do desenvolvimento	Ciclo de vida da segurança
Concepção	Levantamento de requisitos; Levantamento de necessidades; Estabelecimento de relação com o negócio; Levantamento de custos; Revisão de investimentos e budget.	Categorização do sistema; Definição do nível de segurança; Avaliação preliminar de impactos; Definição de requerimentos de segurança;
Elaboração	Especificação de requisitos; Análise de requerimentos; Analise custo/ benefício; Plano de gerenciamento de riscos.	Definição de padrões; Análise de riscos; Planejamento da Segurança.
Construção	Implementação; Testes de casos de uso; Testes funcionais.	Desenvolvimento de controles; Testes de segurança; Documentação.
Transição	Implantação do sistema; Treinamento.	Avaliar a segurança.
Manutenção	Monitoramento do sistema; Gerenciamento de mudanças.	Gerenciamento de controles.
Desativação	Descarte de mídias; Fechamento do sistema.	Preservação das informações.

Tabela 6 - Fases do SDLC, atividades de desenvolvimento e segurança.

Além do ciclo de desenvolvimento e do ciclo de segurança, um sistema seguro contempla também o ciclo de validação. O gráfico apresenta a integração entre os três ciclos de vida do sistema:

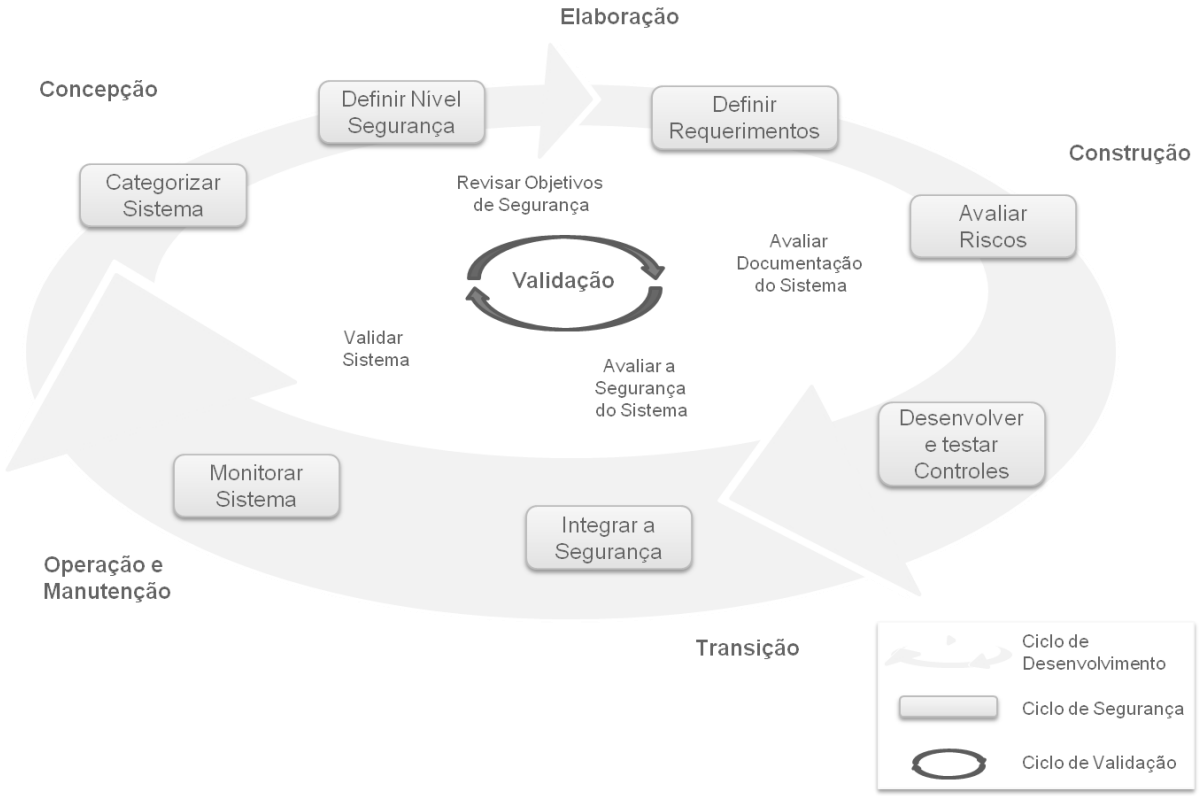


Figura 3 - Integração entre as atividades de desenvolvimento, segurança e validação de sistemas.

As atividades contidas no diagrama trazem duas linhas de proteção:

- Proteção do Ciclo de Vida do Desenvolvimento de Sistemas;
- Proteção do sistema desenvolvido.

Dentre as atividades de segurança do SDLC, há etapas fundamentais para estabelecer os requerimentos de segurança de sistemas:

1. Avaliação de impacto sobre a confidencialidade, integridade e disponibilidade;
2. Definição do nível de segurança;
3. Definição de requerimentos de segurança.

2.1. Avaliação de impactos e definição do nível de segurança

A avaliação de impacto deverá ser realizada pelo analista de privacidade e validada pelo proprietário do sistema, com base na tabela do FIPS 199 (NIST, 2004):

Objetivos de Segurança	Impacto Potencial		
	BAIXA	MÉDIA	ALTA
Confidencialidade Restrições quanto ao acesso e a divulgação das informações, incluindo meios de proteger informações de privacidade e direitos de propriedade pessoais.	A divulgação não autorizada da informação poderia causar efeitos prejudiciais limitados nas operações e nos ativos organizacionais ou individuais.	A divulgação não autorizada da informação poderia causar sérios efeitos prejudiciais nas operações e nos ativos organizacionais ou individuais.	A divulgação não autorizada da informação poderia causar efeitos prejudiciais severos ou catastróficos nas operações e nos ativos organizacionais ou individuais.
Integridade Proteção contra modificação ou destruição indevida das informações, e garantindo a autenticidade e o não-repúdio da informação.	A modificação ou destruição não autorizada da informação poderia causar efeitos prejudiciais limitados nas operações e nos ativos organizacionais ou individuais.	A modificação ou destruição não autorizada da informação poderia causar sérios efeitos prejudiciais nas operações e nos ativos organizacionais ou individuais.	A modificação ou destruição não autorizada da informação poderia causar efeitos prejudiciais severos ou catastróficos nas operações e nos ativos organizacionais ou individuais.
Disponibilidade Garantia de uso e de acesso confiável e em tempo à informação.	A interrupção do uso ou acesso à informação ou a um sistema poderia causar efeitos prejudiciais limitados nas operações e nos ativos organizacionais ou individuais.	A interrupção do uso ou acesso à informação ou a um sistema poderia causar sérios efeitos prejudiciais nas operações e nos ativos organizacionais ou individuais.	A interrupção do uso ou acesso à informação ou a um sistema poderia causar efeitos prejudiciais severos ou catastróficos nas operações e nos ativos organizacionais ou individuais.

Tabela 7 - Avaliação de impactos sobre os objetivos de segurança.

Para auxiliar na tarefa de classificação do impaco sobre as informações com as quais o sistema vai lidar, o seguinte fluxo de classificação pode ser utilizado:

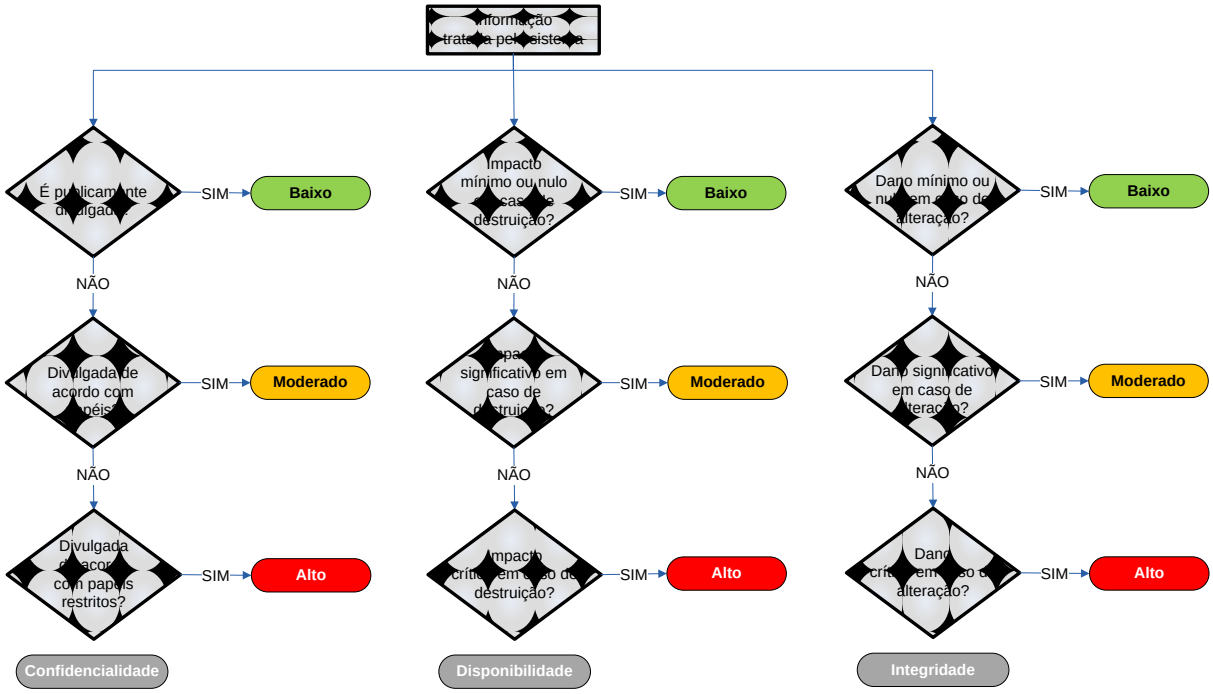


Figura 4- Fluxo de classificação de informações.

Para definir os níveis, deve ser utilizada a tabela de correspondência de impacto em confidencialidade, disponibilidade e integridade:

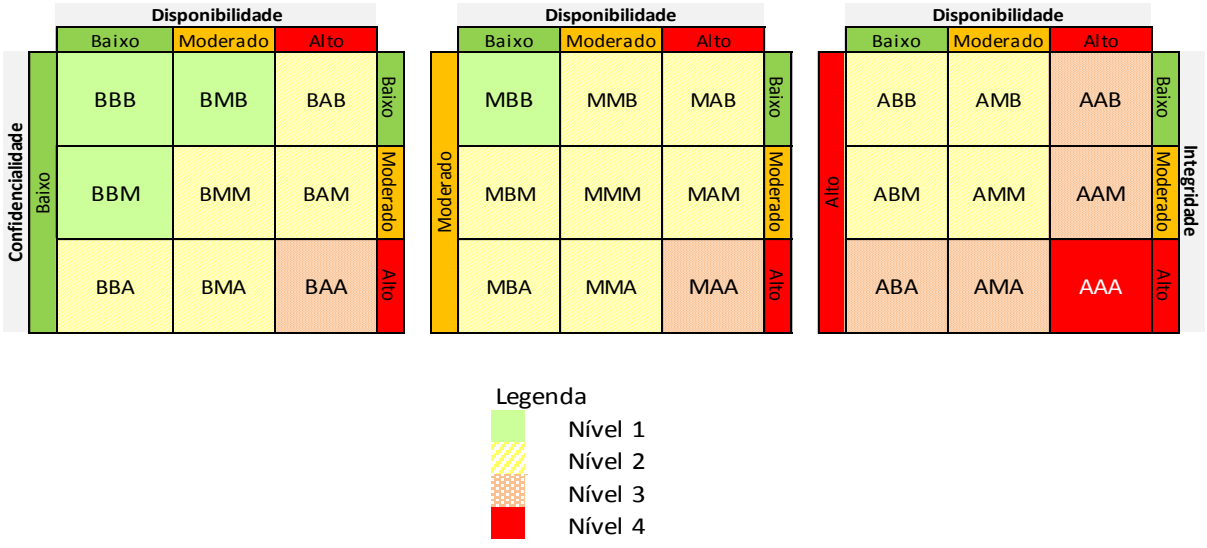
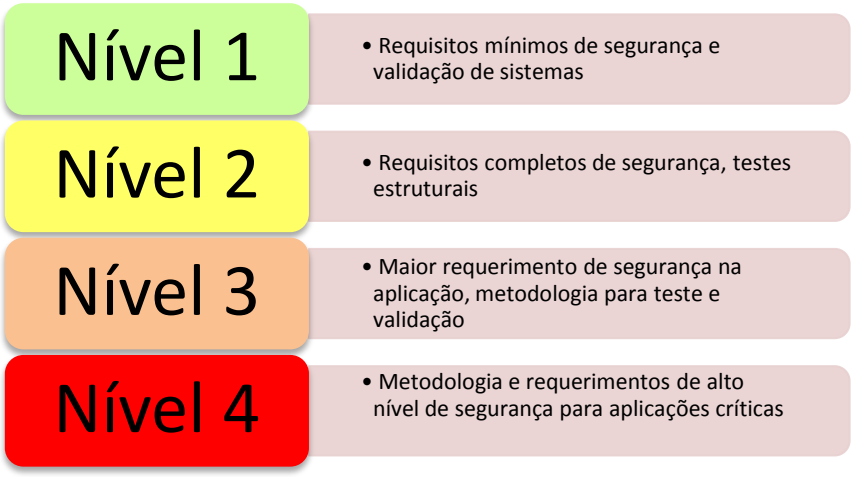


Tabela 8 - Identificação do nível de segurança a partir do impacto.

Para cada nível de segurança, há requerimentos específicos para a proteção dos sistemas desenvolvidos, descritos no capítulo 3.

Com a avaliação de impacto, é possível estabelecer o nível de segurança para a aplicação, dentre os quatro:



Instruções para utilização da tabela de correspondência:

- Primeiramente, deverá ser atribuído o grau de impacto sobre a confidencialidade. Caso seja baixo, deverá ser utilizado o primeiro bloco. Quando alto, o bloco à direita. Se moderado, deverá ser utilizado o bloco central;
- Em seguida, deverá ser encontrada a coluna com base no impacto sobre a disponibilidade;
- Por último, deverá ser localizada a célula que indicará o nível de segurança da aplicação, ao localizar o grau de impacto sobre a integridade

Exemplo de uso (sistema de fiscalização de renda):

1. Na concepção do sistema, deverão ser consideradas as necessidades funcionais desse e as principais ameaças;
2. Deverá ser realizada a avaliação preliminar de impacto sobre confidencialidade, disponibilidade e integridade:
    - As informações são sigilosas, e não devem ser divulgadas para terceiros sem autorização explícita. O vazamento de informações pode gerar custos com indenizações: **Impacto alto (A)**;

- Embora seja online, o sistema não demanda disponibilidade total, podendo haver curtas interrupções de serviço. A fase crítica de disponibilidade ocorre no final dos períodos fiscais: **Impacto moderado (M)**;
- As informações de arrecadação não podem, em hipótese alguma, sofrer alterações não autorizadas. Essas alterações podem gerar custos com fraudes pela falta de arrecadação e por processos administrativos: **Impacto Alto (A)**;

2. Ou seja, é uma aplicação de impacto "**AMA**": alto para confidencialidade, moderado para disponibilidade e alto para integridade;
3. Pela tabela de impacto, percebe-se que é um sistema que exige nível 3 de segurança e avaliação.

2.2. Papéis e responsabilidades

Há muitos participantes envolvidos no desenvolvimento de um sistema, e cada um tem seu papel a cumprir. A tabela exemplifica as principais responsabilidades no contexto do SDLC:

Papel	Responsabilidades	Atividades
Superintendência de Tecnologia da Informação	Gerenciamento global do processo.	
Chief Information Officer (CIO)	Executivo responsável pelo planejamento, orçamento, investimento e performance dos sistemas da informação.	Auxílio à equipe operacional na definição dos objetivos dos sistemas da informação.
Diretor de Qualidade e Testes	Responsável pelos testes e validação de sistemas, quanto às suas funções e o alinhamento com os requisitos.	Especificar testes de qualidade; Revisar testes do sistema com base nos critérios de segurança e especificações técnicas.
Proprietário do Sistema	Responsável pelo desenvolvimento, integração, modificação, operação e manutenção de um sistema da informação.	
Gerente de Contratos	Autorizar, administrar e encerrar contratos.	Validar a correspondência entre o solicitado, o proposto e o entregue.
Analista de Contratos	Avaliar e gerenciar aspectos técnicos dos contratos.	Avaliar a correspondência entre o solicitado, o proposto e o entregue.
Gerente de Autorização	Executivo que assume a responsabilidade pela operação, dentro dos níveis aceitáveis de riscos aos sistemas, ativos, indivíduos e outras organizações.	Revisão do plano de segurança; Validação do relatório de avaliação de segurança; Confirmação dos planos de

		ação e metas para reduzir ou eliminar vulnerabilidades em sistemas da informação.
Chief Information Security Officer	Promulgar políticas de incorporação da segurança ao SDLC e desenvolver padrões corporativos para segurança da informação.	
Gerente de Segurança	Responsável por garantir a segurança durante todo o ciclo de vida dos sistemas.	
Analista de Privacidade	Garantir que os sistemas da informação atendem aos requerimentos de privacidade de dados determinados pelas políticas locais.	
Gerente de Configurações	Gerenciar os efeitos das mudanças ou diferenças em configurações em um ambiente de sistemas da informação.	Auxílio à equipe operacional no alinhamento das mudanças, prevenindo alterações indesejadas.
Gerente de Desenvolvimento	Garantir a entrega do sistema de acordo com a proposta.	Entender os requerimentos de segurança para assegurar sua implementação
Arquiteto de Sistemas	Criar, conceitualmente, o panorama do sistema a ser desenvolvido.	Manter a integridade do sistema no decorrer das fases do desenvolvimento.
Desenvolvedor/ Programador	Escrever os códigos fonte dos sistemas desenvolvidos, incluindo a "codificação segura".	Coordenar e trabalhar com o Gerente de Configurações a implementação das mudanças.
Jurídico		Avaliar e auxiliar questões legais do processo.

Outros Participantes	De acordo com a complexidade do sistema, podem surgir outros participantes, trabalhando de forma associada com as funções listadas. Usuários, por exemplo, devem ser parte integrante do desenvolvimento dos sistemas, listando necessidades, refinando os requerimentos e inspecionando ou validando o sistema entregue. Também podem ser incluídos auditores, equipes de TI, pessoal de segurança patrimonial, entre outros.

Tabela 9 - Papéis e reponsabilidades.

Pela recorrente falta de pessoal, há possibilidade de acúmulo de papéis. Contudo, há papéis que não podem ser empenhados por uma mesma pessoa, conforme mostram as marcações (8) na tabela:

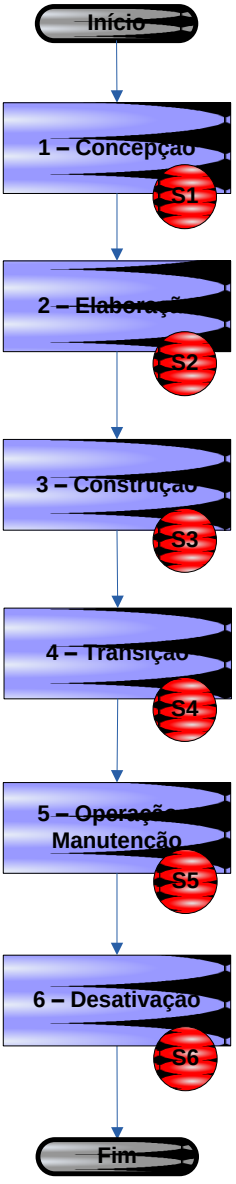
Papel	Gerente de Autorização	Gerente de Configurações	Gerente de Contratos	Gerente de Desenvolvimento	Diretor de Qualidade e Testes	Desenvolvedor/ Programador
Gerente de Autorização		8		8		8
Gerente de Configurações	8				8	
Gerente de Contratos				8		
Gerente de Desenvolvimento	8		8		8	
Diretor de Qualidade e Testes		8		8		8
Desenvolvedor/ Programador	8				8	

Tabela 10 - Incongruência de papéis.

Caso as restrições de acúmulo de funções não sejam respeitadas, deverá haver registro explícito que evidencie e justifique a opção adotada.

2.3. Ciclo de Vida da Segurança no SDLC

Como apresentado anteriormente, há atividades de segurança para cada fase do ciclo de vida do desenvolvimento de sistemas. Considerando que as atividades são realizadas em sequência e, quando implementadas pelo SDLC, iterativas, para cada grupo de atividades há pontos de controle para a verificação do sistema e sua documentação.



Legenda:

Fase do SDLC

Conjunto de atividades de segurança

Documento gerado para as etapas de validação

2.4. Concepção

Objetivos de Segurança da fase de Concepção:

Garantir que ameaças, requerimentos de segurança e potenciais restrições às funcionalidades e à integração do sistema serão considerados. Nesta fase do S-SDLC, a segurança é voltada aos riscos do negócio.

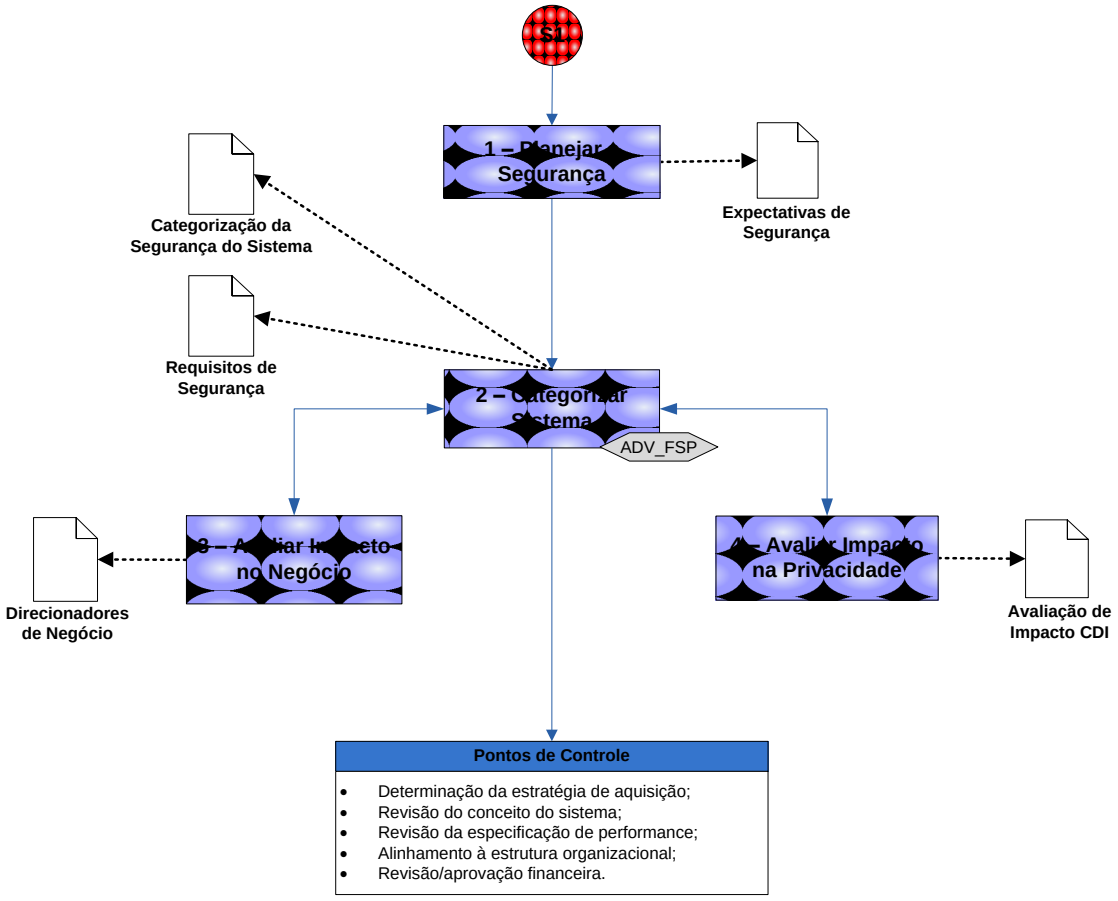


Figura 5 - Atividades de segurança e documentos da concepção.

Atividade			Output	Responsável
SD	1	**Planejamento de Segurança:** Identificar os papéis chaves para o desenvolvimento do sistema; Identificar as fontes de requerimentos de segurança, tais como leis, regulações e padrões; Garantir que todos os interessados (*stakeholders*) têm um comum entendimento, incluindo implicações, considerações e requerimentos de segurança; Gerar idéias iniciais de marcos do projeto relativos à segurança, incluindo prazos.	Documentos genéricos: slides, atas de reunião, etc.; Alinhamento das expectativas de segurança; Calendário inicial de atividades de segurança e decisões.	Órgão
SD	2	**Categorização do Sistema:** Identificação de quais informações suportam quais linhas de negócio; Avaliar a segurança quanto à confidencialidade, integridade e disponibilidade; Gerar links entre missão, informações e o sistema com o custo efetivo da segurança da informação.	Categorização da segurança: documentação das pesquisas, das decisões chave e da análise lógica de suporte à categorização da segurança do sistema, o que inclui o plano de segurança; Requerimentos de segurança de alto nível; Estimativas de nível de esforço.	Órgão

SD	3	**Avaliação de Impacto no Negócio:** Correlacionar componentes específicos do sistema com os serviços de negócio críticos que são fornecidos; Caracterizar as consequências de uma ruptura nestes componentes para a missão e o negócio; Avaliar o nível de impacto na disponibilidade;	Identificação das linhas de serviço suportadas pelo sistema e como as mesmas serão impactadas; Identificação dos principais componentes de sistema necessários para a manutenção das funcionalidades mínimas; Identificação do período de tempo que o sistema pode ter suas operações interrompidas, sem que o negócio seja impactado; Identificação do nível de tolerância do negócio à perda de dados.	Órgão
SD	4	**Avaliação de Impacto na Privacidade:** Avaliar detalhadamente as informações privadas de cada processo do sistema; Incorporar a avaliação de impacto na privacidade ao plano de segurança.	Avaliação de impacto na privacidade: detalhamento de onde e por qual motivo as informações privadas são coletadas, armazenadas e/ou criadas dentro do sistema.	Órgão

2.4.1. Pontos de Controle (Concepção)

Fase 1	Pontos de Controle (PC1)	
Concepção	1	Determinação da estratégia de aquisição
	2	Revisão do conceito do sistema
	3	Revisão da especificação de performance
	4	Alinhamento à estrutura organizacional
	5	Revisão / aprovação financeira

2.5. Elaboração

Objetivos de segurança da fase de Elaboração:

Conduzir a avaliação de riscos, usar os resultados para suprir a base de controles de segurança e analisar os requisitos de segurança.

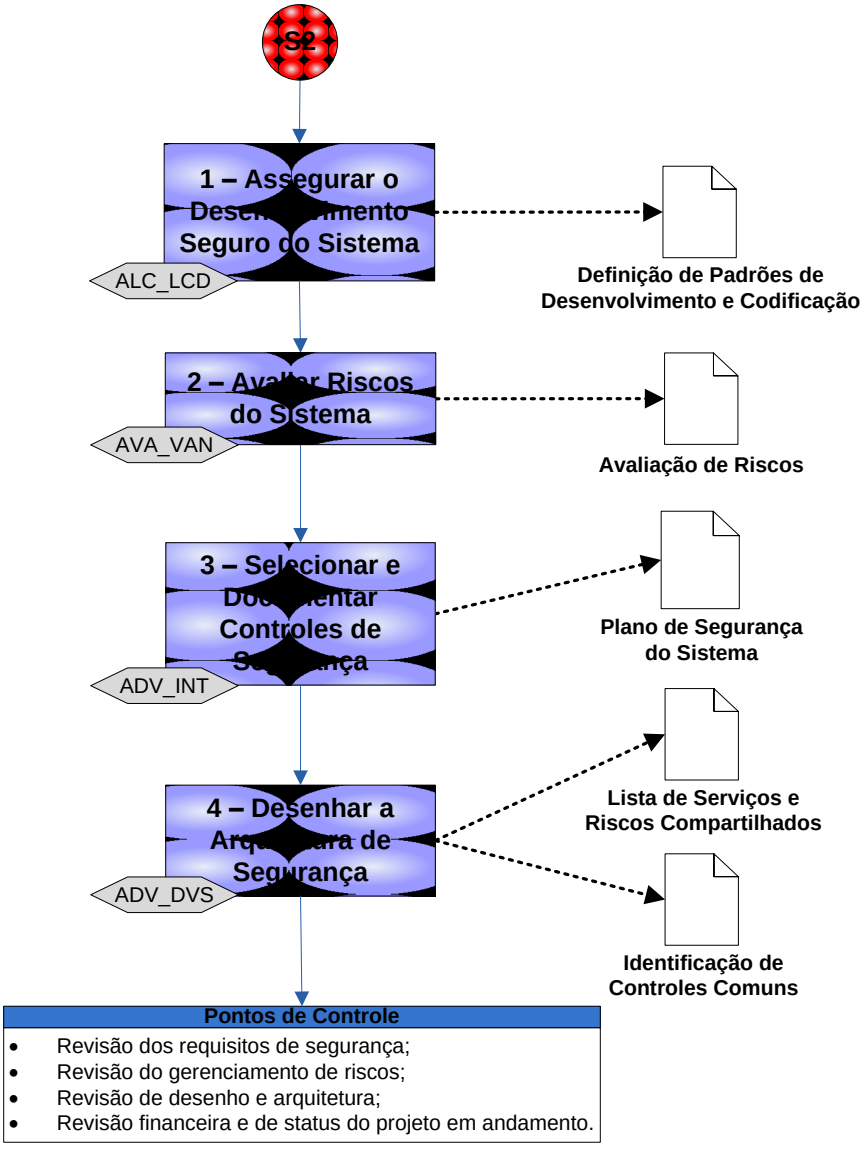


Figura 6 - Atividades de segurança e documentos da elaboração.

Atividade			Output	Responsável
SD	5	**Assegurar o Desenvolvimento Seguro do Sistema:** Definir atividades e plano de comunicação para gerenciar a qualidade do sistema;	Plano de treinamento de segurança para a fase de desenvolvimento; Plano de verificação de qualidade, entregas e marcos do projeto; Padrões de desenvolvimento e codificação;	Órgão
SD	6	**Avaliação de riscos do sistema:** Avaliar o conhecimento atual do desenho do sistema, requerimentos estabelecidos, e os requerimentos mínimos derivados do processo de categorização de segurança, para determinar sua efetividade na mitigação dos riscos antecipados; Avaliar se os controles de segurança fornecem a proteção apropriada; Avaliar como o sistema poderia afetar outros sistemas com os quais este estará conectado.	Avaliação refinada dos riscos, baseada em um desenho mais maduro do sistema, que reflete melhor os potenciais riscos do sistema, as vulnerabilidades conhecidas no na fase de desenho, as restrições identificadas, e as ameaças conhecidas. Transição entre requerimentos iniciais e controles específicos do sistema.	Fornecedor
SD	7	**Seleção e documentação dos controles de segurança:** Selecionar os controles de segurança; Complementar a lista de controles com controles adicionais baseados em condições locais.	Plano de segurança do sistema: especificação dos controles de segurança, identificando quais, onde e como os controles serão aplicados.	Fornecedor

SD	8	**Desenho da arquitetura de segurança:** Planejar a forma como os controles de segurança serão integrados ao sistema; Planejar a integração dos serviços obtidos externamente.	Esquema de integração de segurança fornecendo detalhes de onde, dentro do sistema, a segurança é implementada e compartilhada. Arquiteturas de segurança devem ser detalhadas e representadas graficamente; Lista de serviços compartilhados e riscos compartilhados resultantes; Identificação de controles comuns usados pelo sistema.	Fornecedor

2.5.1. Pontos de Controle (Elaboração)

Fase 2	Pontos de Controle (PC2)	
Elaboração	1	Revisão do gerenciamento de riscos
	2	Revisão de desenho e arquitetura
	3	Revisão financeira e de status do projeto em andamento

2.6. Construção

Objetivos de segurança da fase de Construção:

Assegurar a presença e a integração dos controles de segurança. Avaliar a efetividade dos controles.

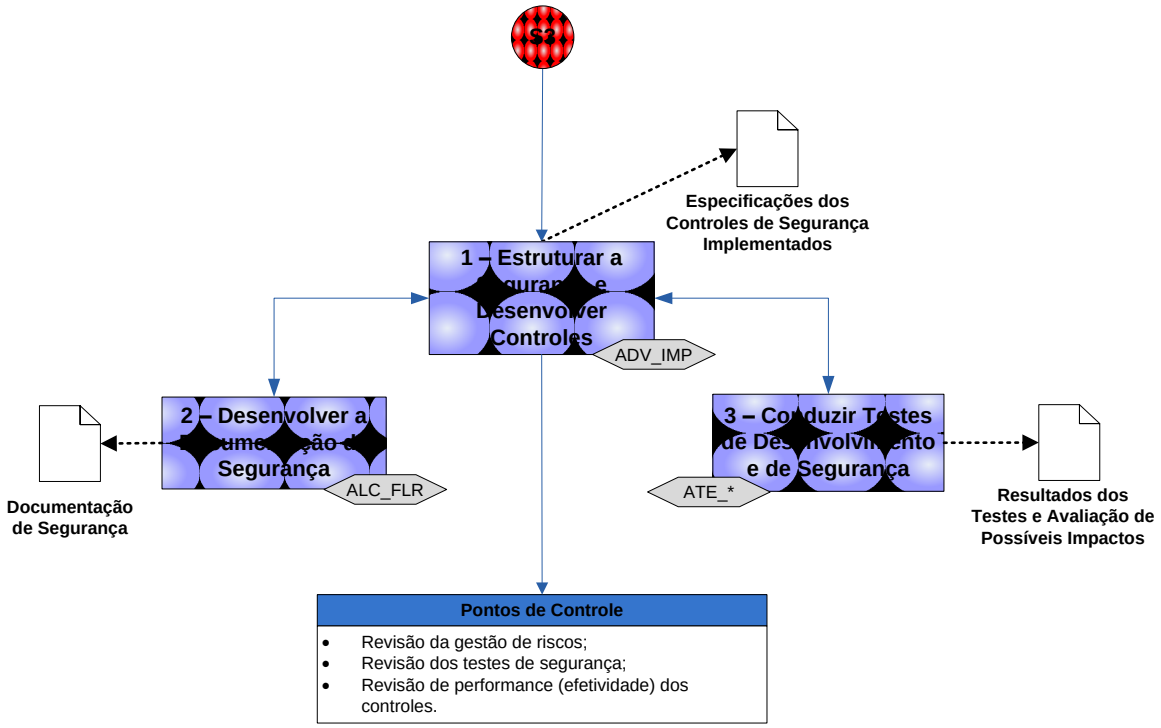


Figura 7 - Atividades de segurança e documentos da elaboração.

Atividade			Output	Responsável
SD	9	**Estruturação da segurança e desenvolvimento de controles:** Implementar controles de segurança; Documentar as principais decisões tomadas e seus direcionadores; Implementar e documentar controles compensatórios.	Controles implementados, com a documentação da especificação para inclusão no plano de segurança; Lista de variações de controles de segurança resultantes das decisões do desenvolvimento; Potenciais cenários de avaliação para testar vulnerabilidades e limitações conhecidas.	Fornecedor
SD	10	**Desenvolvimento da documentação de segurança:** Complementar a documentação do sistema com: plano de gerenciamento de configuração; plano de contingência; plano de monitoramento contínuo; plano de treinamento e conscientização; plano de resposta a incidentes.	Documentação adicional de segurança suportando o plano de segurança do sistema.	Fornecedor
SD	11	**Condução de testes de desenvolvimento, funcionais e de segurança:** Testar e avaliar as funcionalidades, a qualidade e a segurança do sistema; Verificar se o sistema foi desenvolvido de acordo com os requerimentos funcionais e de segurança;	Documentação dos resultados dos testes, incluindo qualquer variação inesperada.	Fornecedor / Órgão

2.6.1. Pontos de Controle (Construção)

Fase 3	Pontos de Controle (PC3)	
Construção	1	Revisão da gestão de riscos
	2	Revisão dos testes funcionais e de segurança
	3	Revisão de performance (efetividade) de controles

2.7. Transição (Implantação)

Objetivos de segurança da fase de Transição:

Avaliar a integração do sistema ao seu ambiente. Completar atividades de acreditação do sistema.

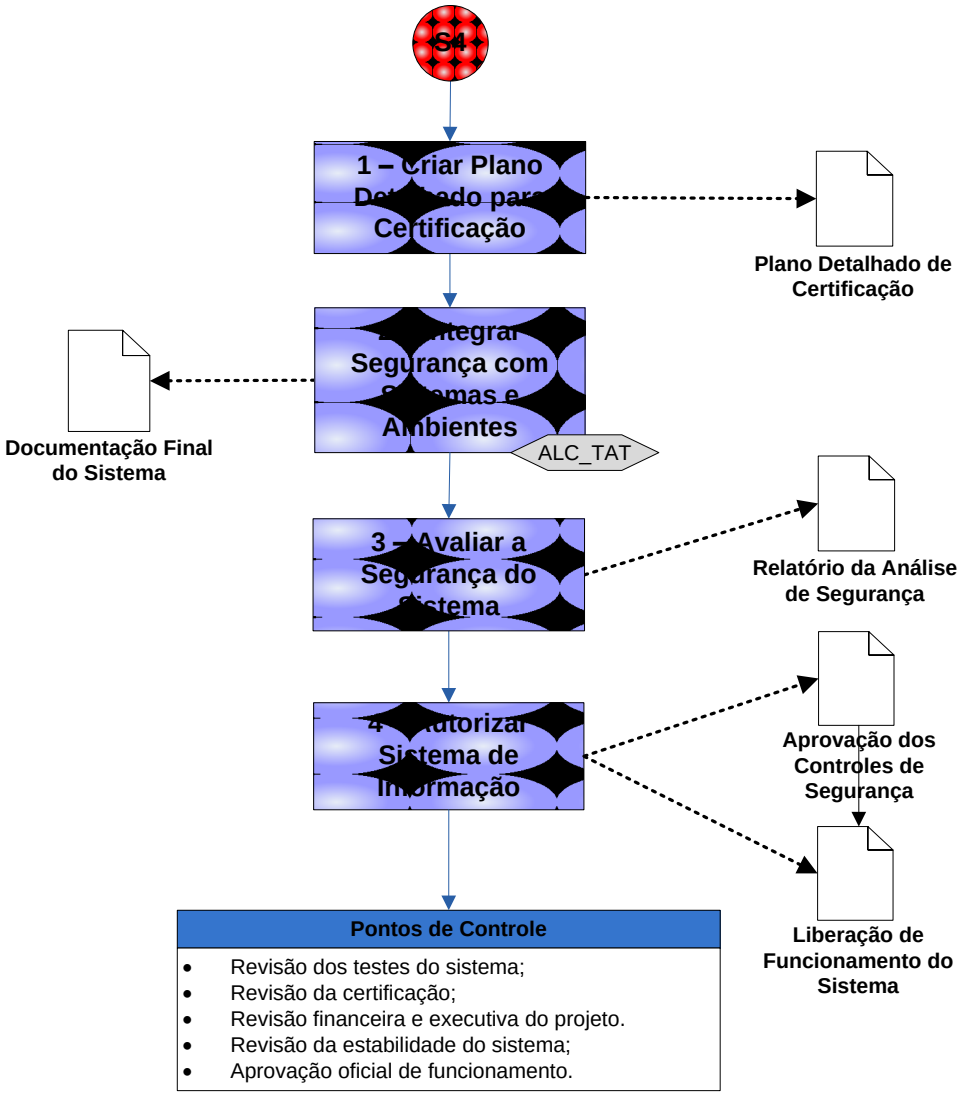


Figura 8 - Atividades de segurança e documentos da transição.

Atividade			Output	Responsável
SD	12	**Criação de plano detalhado para certificação e acreditação:** Identificar e avaliar a aceitação dos riscos residuais; Delimitar o escopo dos testes de implantação e o nível de rigor esperado.	Plano inicial de trabalho: documento de planejamento que identifica as atribuições dos colaboradores do projeto, componentes centrais, escopo dos testes, e o nível de rigor esperado. O pacote de avaliação deverá ser próximo do resultado final de avaliação.	Fornecedor / Órgão
SD	13	**Integração da segurança com sistemas e ambientes:** Verificar os controles de segurança operacionais e a integração dos mesmos ao ambiente de produção;	Lista de verificação dos controles de segurança; Documentação completa do sistema.	Fornecedor
SD	14	**Avaliação da segurança do sistema:** Avaliar a segurança do sistema; Validar se o sistema atende aos requerimentos funcionais e de segurança, e se será operado dentro de um nível aceitável de riscos residuais de segurança.	Relatório de avaliação de segurança do sistema, com base nas metodologias de testes estabelecidas no manual de segurança; Atualização do plano de segurança do sistema.	Órgão
SD	15	**Autorizar Sistema de Informação:** Autorizar formalmente o funcionamento do sistema;	Decisão de autorização de segurança endossada pelo Gerente de Autorização; Pacote final de segurança do sistema, contemplando: plano de segurança completo, resultados dos testes de segurança e os planos de ação e metas estabelecidos para reduzir e eliminar as vulnerabilidades do sistema	Órgão

2.7.1. Pontos de Controle (Transição)

Fase 4	Pontos de Controle (PC4)	
Transição	1	Revisão dos testes do sistema
	2	Revisão da certificação e acreditação
	3	Revisão financeira e executiva do projeto
	4	Revisão da estabilidade do sistema
	5	Aprovação oficial de funcionamento

2.8. Operação e Manutenção

Objetivos de segurança da fase de Operação e Manutenção:

Assegurar que, após iniciar o funcionamento, o sistema tenha a plenitude das funcionalidades e dos controles de segurança. Acompanhar mudanças de software e hardware para garantir que as demandas de segurança serão mantidas.

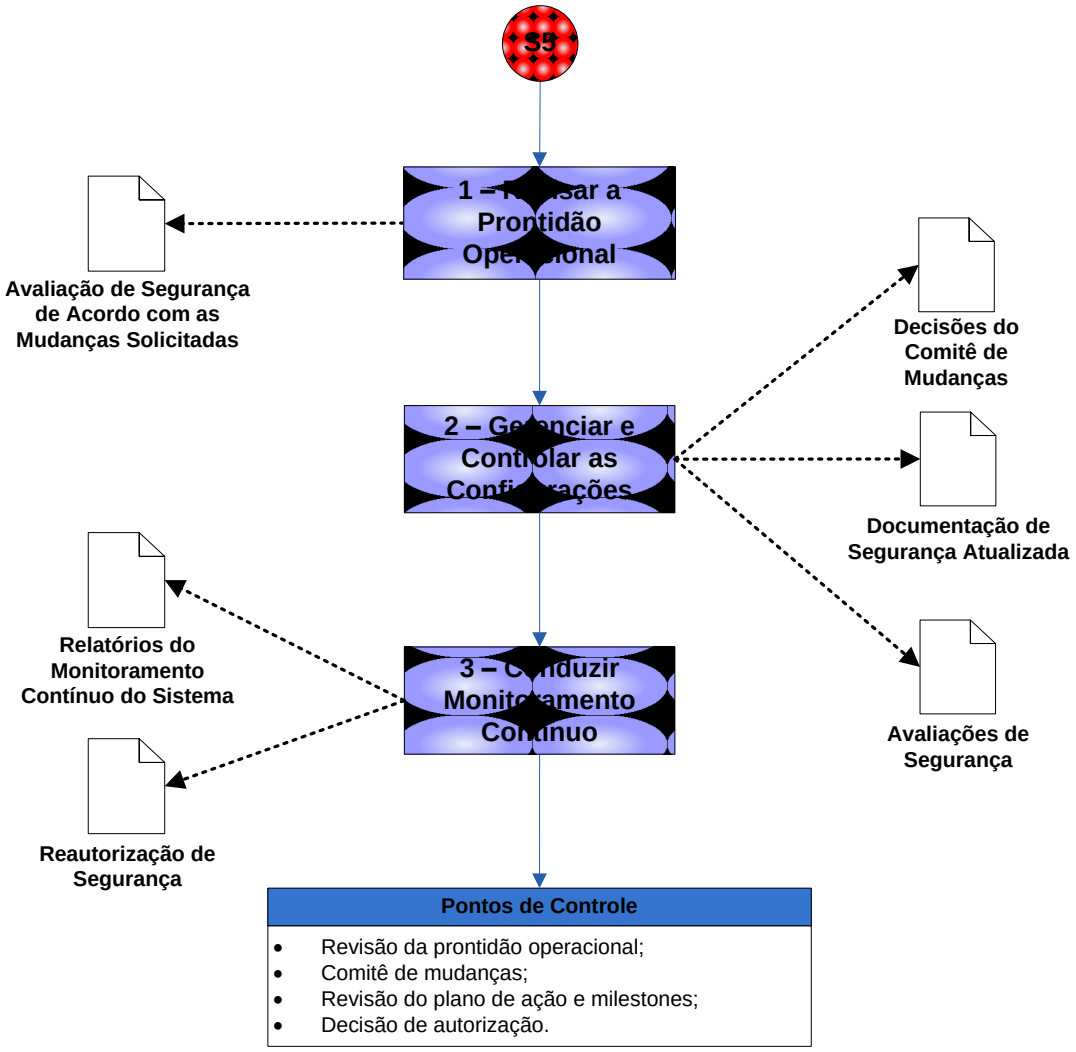


Figura 9 - Atividades de segurança e documentos da manutenção.

Atividade			Output	Responsável
SD	16	**Revisar a prontidão operacional:** Analisar as mudanças ocorridas e verificar a integridade dos controles.	Avaliação das implicações de segurança após alterações no sistema.	Órgão
SD	17	**Realizar o gerenciamento de configurações:** Analisar potenciais impactos em relação a mudanças específicas no sistema; Estabelecer patamar dos componentes de hardware e software do sistema; Controlar e manter o inventário de mudanças do sistema.	Decisões do comitê de controle de mudanças; Atualização da documentação de segurança; Avaliações de segurança das mudanças do sistema.	Órgão
SD	18	**Conduzir monitoramento contínuo:** Determinar se os controles de segurança do sistema continuam sendo efetivos.	Consolidação de dados do monitoramento contínuo; Revisão dos planos de ação e metas (POA&M); Revisões, métricas e medidas de segurança, com análise de tendências; Atualização da documentação de segurança e revalidação de decisões, quando necessário.	Órgão

2.8.1. Pontos de Controle (Operação e Manutenção)

Fase 5	Pontos de Controle (PC5)	
Manutenção	1	Revisão da plenitude operacional
	2	Revisão das mudanças propostas pelo comitê de mudanças
	3	Revisão de planos e milestones definidos na implantação
	4	Decisões de acreditação (a cada três anos ou após mudanças significativas no sistema)

2.9. Desativação

Objetivos de segurança da fase de Desativação:

Identificação explícita dos problemas de segurança da informação relacionados à desativação do sistema. Proteção de informações críticas, recursos e ativos.

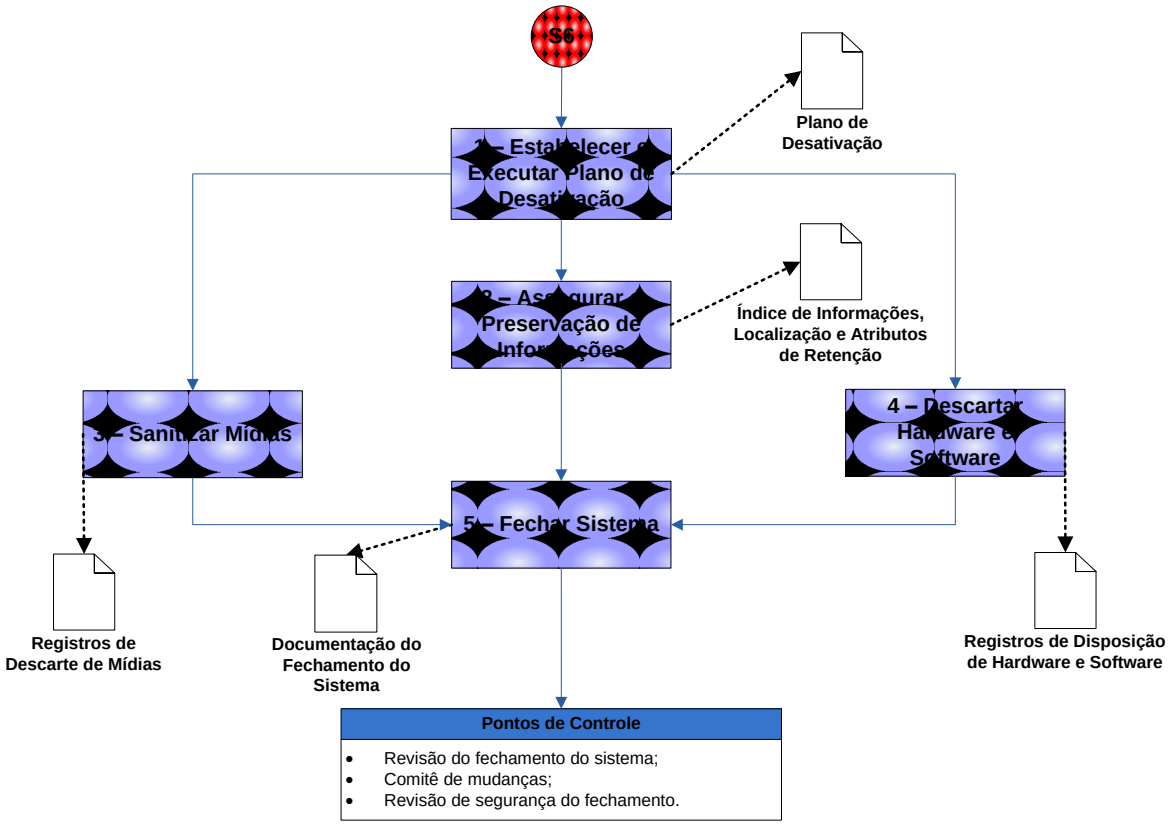


Figura 10 - Atividades de segurança e documentos da desativação.

Atividade			Output	Responsável
SD	19	**Estabelecer e executar planos de descarte ou transição:** Criar plano para garantir que todos os interessados estão conscientes do futuro do sistema e de suas informações; Manter status sobre a transição ou descarte dos serviços, informações e componentes críticos; Identificar passos, decisões e marcos de projeto necessários para o fechamento, transição ou migração do sistema e suas informações;	Documentação do plano de descarte/transição para o sistema e as informações contidas nesse.	Órgão
SD	20	**Assegurar a preservação de informações:** Identificar ações que serão necessárias para a recuperação das informações no futuro; Identificar requisitos legais para retenção de registros.	Índice das informações preservadas, localização e prazos de retenção.	Órgão
SD	21	**Sanitizar mídias:** Sanitizar e destruir mídias digitais antes da desativação do sistema; Trilhar, documentar e verificar as ações de sanitização e destruição de mídias; Prevenir a utilização indevida das informações contidas nas mídias.	Registros de sanitização de mídias.	Fornecedor / Órgão

SD	22	**Descartar hardware e software:** Verificar implicações quanto ao descarte de hardware e software; Descartar hardware e software.	Registros de descarte de software e hardware. Esses registros deverão conter listas de dispositivos e sistemas liberados (vendidos, descartados ou doados), e listas de reaproveitamento desses recursos em outros locais da organização.	Fornecedor / Órgão
SD	23	**Fechar sistema:** Desativar formalmente o sistema.	Documentação de verificação do encerramento do sistema, incluindo a notificação de fechamento encaminhada às pessoas: gerente de autorização, proprietário, gerente de desenvolvimento, e o gerente de segurança.	Órgão

2.9.1. Pontos de Controle (Desativação)

Fase 6	Pontos de Controle (PC6)	
Desativação	1	Revisão do fechamento do sistema
	2	Quadro de controle de mudanças
	3	Revisão de segurança do fechamento

3. Requerimentos de Segurança para as Aplicações

A partir da avaliação de impacto sobre os pilares confidencialidade, disponibilidade e integridade, é estabelecido o nível de segurança requerido para a aplicação.

3.1. Objetivos gerais de segurança para as aplicações

3.1.1. Objetivos Gerenciais

Área	Código		Objetivo
Planejamento	O	1	Desenvolver, documentar, atualizar e implantar políticas de segurança, para evitar o vazamento de informações, paradas não programadas ou alterações indevidas em dados e processos.
Avaliação de riscos	O	2	Avaliar riscos às operações, processos, informações e usuários, associados ao uso dos sistemas da informação
Conscientização e treinamento	O	3	Manter todos os envolvidos com o sistema cientes das políticas de segurança, das ameaças existentes e do papel de cada um para a manutenção da confidencialidade, integridade e disponibilidade das informações e sistemas.
Certificação, validação e avaliação de segurança	O	4	Manter avaliações periódicas para certificar a eficiência dos controles de segurança, bem como autorizar a execução de serviços e sistemas.
Auditoria	O	5	Criar, proteger e reter os registros dos eventos de segurança ou de uso indevido. Garantir que indivíduos sejam responsabilizados por suas ações.

Tabela 11 - Objetivos gerenciais de proteção dos sistemas.

3.1.2. Objetivos Operacionais

Área	Código		Objetivo
Gerenciamento de configurações	O	6	Estabelecer e manter linhas-guia de configurações para cada fase do desenvolvimento dos sistemas. Formalizar a documentação.
Continuidade dos serviços	O	7	Estabelecer, manter e implementar controles para assegurar a perenidade dos serviços, ou atender a critérios mínimos de disponibilidade.
Resposta a incidentes	O	8	Estabelecer controles para garantir a recuperação dos sistemas após eventos que alterem seu funcionamento normal.
Manutenção	O	9	Realizar ajustes e melhorias periódicas nos sistemas, corrigindo falhas e programando atualizações.
Proteção de mídias	O	10	Proteger mídias (em papel ou digitais) referentes aos sistemas da informação ou dados sensíveis, fornecendo o apropriado controle de acesso. Garantir o descarte apropriado dessas.

Tabela 12 - Objetivos operacionais de proteção dos sistemas.

3.1.3. Objetivos Técnicos

Área	Código		Objetivo
Controle de acesso	O	11	Limitar o acesso às informações e serviços somente aos usuários, processos e dispositivos autorizados.
Identificação e autenticação	O	12	Identificar usuários, processos ou dispositivos e verificar (autenticar) suas identidades como pré-requisito para permitir seus acessos nos sistemas.
Proteção de sistemas e comunicações	O	13	Monitorar, controlar e proteger comunicações internas e entre sistemas da informação. Implantar técnicas de software seguro e design de arquitetura para efetivar a segurança.
Integridade de sistemas e informações	O	14	Identificar, registrar e corrigir informações e sistemas alterados de forma indevida ou por falha. Proteger sistemas contra código maliciosos.
Segurança em sistemas	O	15	Adotar práticas de desenvolvimento seguro, implantar restrições no uso dos sistemas e estabelecer confiança com sistemas externos com base nos controles adequados.

Tabela 13 - Objetivos técnicos de proteção dos sistemas.

3.1.4. Objetivos Ambientais

Área	Código		Objetivo
Segurança física e ambiental	O	16	Limitar o acesso físico aos sistemas da informação, equipamentos e ambientes operacionais somente ao pessoal autorizado. Proteger os acessos às unidades em que residem os sistemas da informação, fornecendo os equipamentos necessários para inibir e monitorar o acesso indevido. Proteger os sistemas contra fatores ambientais (enchentes, incêndios, furacões, etc), bem como controlar, apropriadamente, as variáveis ambientais (umidade, temperatura).
Segurança pessoal	O	17	Garantir que as pessoas que ocupam cargos de responsabilidade (mesmo terceiros) são idôneas, confiáveis, e competentes para suas atribuições. Preparar os sistemas de modo a serem independentes às mudanças de cargo, gestão e pessoal. Ter medidas e sanções em casos de uso indevido ou falha no cumprimento das políticas de segurança.

Tabela 14 - Objetivos ambientais de proteção dos sistemas.

3.2. Definição de Controles

O nível de segurança estabelece os requerimentos de segurança de uma aplicação. Ao receber a demanda, um desenvolvedor precisa indicar controles técnicos para atender a esses e, com base na avaliação de riscos e custos, desenvolvedor e cliente de alinham as expectativas de controles, conforme mostra a figura adaptada do SP 800-34 (NIST, 2002).

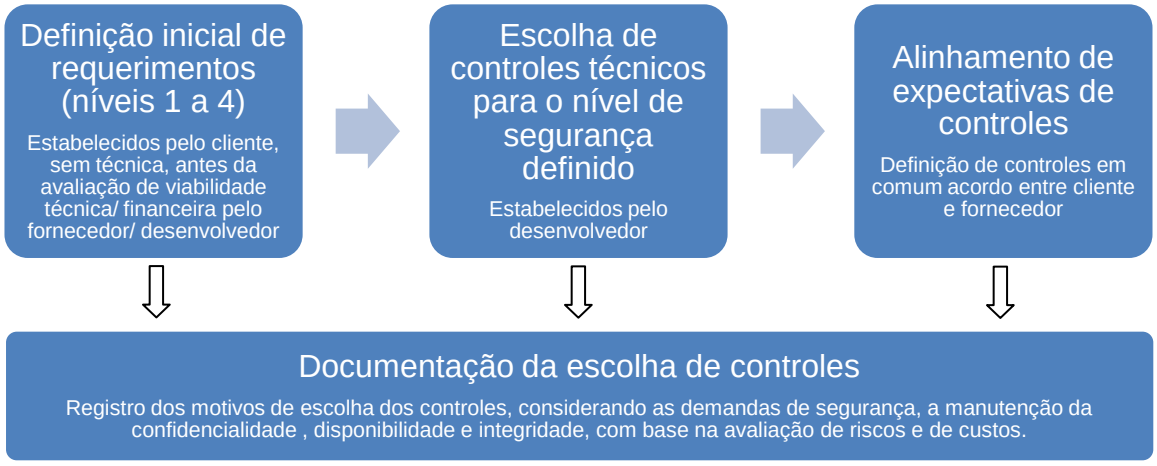


Figura 11 - Processo de estabelecimento de controles.

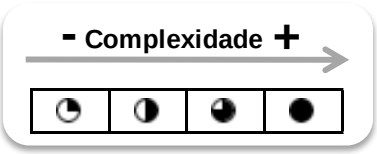
3.3. Classes Funcionais e suas Famílias

Este capítulo enumera requerimentos de segurança com base nas Classes e Famílias funcionais da ISO 15408-2 (International Organization for Standardization, 2008). Ao todo, são 11 classes:

Classe Funcional	Sigla	Descrição	Objetivos atendidos
Auditoria de Segurança	FAU	Requerimentos para seleção, análise e proteção dos eventos de auditoria e segurança	O-4, O-5, O-9
Comunicação	FCO	Estabelecimento de não repúdio, identificando origem e destino na troca de dados	O-13, O-15
Criptografia	FCS	Uso e gerenciamento de chaves criptográficas	O-13, O-15
Proteção de Dados do Usuário	FDP	Políticas e requerimentos para a proteção de dados de usuários	O-10, O-11, O-14, O-17
Identificação e Autenticação	FIA	Determinar e verificar a identidade dos usuários, e estabelecer limites de uso no sistema.	O-11, O-12, O-17
Gerenciamento de Segurança	FMT	Gerenciamento de atributos, dados e funções de segurança. Gerenciamento de regras e interações.	O-1, O-2, O-3, O-6, O-9
Privacidade	FPR	Funções de anonimato de usuários e proteção contra interceptação de dados.	O-12
Proteção das Funcionalidades de Segurança	FPT	Integridade e gerenciamento dos mecanismos e dados das funções de segurança.	O-7, O-8, O-10, O-13, O-14, O-16, O-17
Utilização de Recursos	FRU	Controles de tolerância a falhas, alocação de recursos e disponibilidade de serviços.	O-7, O-14
Acesso ao Sistema	FTA	Limitação do escopo de acesso e sessões de usuários.	O-12, O-13
Canais de Confiança	FTP	Confiança na comunicação entre usuários e sistemas.	O-13, O-15

Tabela 15 - Classes Funcionais da ISO 15408-2.

Cada classe funcional contém suas famílias de requerimentos e, quanto maior o nível de segurança do sistema, maior pode ser a complexidade desses requerimentos. A tabela 16 apresenta, para cada nível de segurança, as variações nos graus de complexidade dos requerimentos.



Como exemplo de leitura dessa tabela, pode ser verificado que a família FRU_FLT contém três graus de complexidade, mas não é obrigatória sua aplicação em sistemas de nível 1. Por outro lado, a família FPT_PHP contém quatro graus de complexidade, coincidindo com os níveis de segurança.

Classe Funcional	Família	Sigla	Nível			
			1	2	3	4
			Complexidade			
Auditoria de Segurança	Resposta Automática de Auditoria	FAU_ARP	◔	◑	◑	◑
	Geração de Dados de Auditoria	FAU_GEN	◔	◑	◕	●
	Análise da Auditoria de Segurança	FAU_SAA		◔	◑	◕
	Revisão da Auditoria de Segurança	FAU_SAR	◔	◑	◑	◕
	Seleção de Eventos da Auditoria de Segurança	FAU_SEL		◔	◔	◔
	Armazenamento dos Eventos de Auditoria de Segurança	FAU_STG		◔	◔	◑

Classe Funcional	Família	Sigla	Nível			
			1	2	3	4
			Complexidade			
Comunicação	Irretratabilidade da Origem	FCO_NRO		◔	◔	◔
	Irretratabilidade do Destinatário	FCO_NRR	◔	◑	◑	◕

Classe Funcional	Família	Sigla	Nível			
			1	2	3	4
			Complexidade			
Criptografia	Gerenciamento de Chaves Criptográficas	FCS_CKM	◔	◔	◔	◑
	Operação Criptográfica	FCS_COP	◔	◑	◕	◕

Classe Funcional	Família	Sigla	Nível 1	2	3	4
			Complexidade			
Proteção de Dados do Usuário	Política de Controle de Acesso	FDP_ACC	◔	◑	◕	◕
	Funções de Controle de Acesso	FDP_ACF	◔	◑	◕	●
	Autenticação de Dados	FDP_DAU	◔	◑	◕	●
	Exportação de Dados	FDP_ETC		◔	◑	◕
	Política do Fluxo de Informação	FDP_IFC	◔	◑	◕	●
	Funções do Fluxo de Informação	FDP_IFF	◔	◑	◕	●
	Importação de Dados	FDP_ITC	◔	◑	◑	◑
	Transferência Interna de Dados	FDP_ITT	◔	◑	◕	●
	Proteção da Informação Residual	FDP_RIP	◔	◑	◕	●
	Reversão	FDP_ROL	◔	◑	◕	●
	Integridade de Dados Armazenados	FDP_SDI		◔	◑	◑
	Confidencialidade dos Dados em Trânsito dos Usuários	FDP_UCT		◔	◑	◕
	Integridade de Dados em Trânsito dos Usuários	FDP_UIT		◔	◑	◑

Classe Funcional	Família	Sigla	Nível 1	2	3	4
			Complexidade			
Identificação e Autenticação	Falhas de Autenticação	FIA_AFL	◔	◑	◕	●
	Definição de Atributos de Usuários	FIA_ATD		◔	◔	◔
	Especificação de Segredos	FIA_SOS	◔	◑	◑	◕
	Autenticação de Usuários	FIA_UAU	◔	◑	◕	●
	Identificação de Usuários	FIA_UID	◔	◔	◔	◔
	Ligação usuário-assunto	FIA_USB	◔	◑	◑	◑

Classe Funcional	Família	Sigla	Nível 1	2	3	4
			Complexidade			
Gerenciamento de Segurança	Gerenciamento de funções do sistema	FMT_MOF	◔	◔	◑	◑
	Gerenciamento de atributos de segurança	FMT_MSA		◔	◔	◔
	Gerenciamento de dados das funções de	FMT_MTD	◔	◑	◑	◑
	Revogação	FMT_REV	◔	◑	◑	◕
	Validade dos atributos de segurança	FMT_SAE	◔	◑	◑	◑
	Especificação das funções de gerenciamento	FMT_SMF	◔	◔	◔	◔
	Regras de gerenciamento de segurança	FMT_SMR	◔	◔	◔	◔

Classe Funcional	Família	Sigla	Nível 1	2	3	4
			Complexidade			
Privacidade	Anonimato	FPR_ANO	◔	◔	◑	◑
	Pseudoanonimato	FPR_PSE		◔	◔	◔
	Desvinculação	FPR_UNL		◔	◔	◔
	Restrição à interceptação	FPR_UNO	◔	◑	◕	●

Classe Funcional	Família	Sigla	Nível 1	2	3	4
			Complexidade			
Proteção das Funcionalidades de Segurança	Falha segura	FPT_FLS		◔	◑	◑
	Disponibilidade de dados exportados da	FPT_ITA		◔	◔	◔
	Confidencialidade de dados exportados da	FPT_ITC		◔	◑	◕
	Integridade dos dados exportados da aplicação	FPT_ITI		◔	◔	◔
	Segurança nas transferências internas de dados	FPT_ITT	◔	◔	◔	◔
	Segurança física para a aplicação	FPT_PHP	◔	◑	◕	●
	Recuperação confiável	FPT_RCV	◔	◑	◕	●
	Detecção de repetição (replay)	FPT_RPL		◔	◔	◔
	Protocolo de sincronia de estado	FPT_SSP		◔	◑	◑
	Carimbos de tempo	FPT_STM		◔	◔	◑
	Consistência de dados	FPT_TDC	◔	◑	◑	◑
	Teste de sistemas externos	FPT_TEE	◔	◑	◑	◑
	Consistência na replicação de dados	FPT_TRC	◔	◑	◑	◑
	Auto-teste da aplicação	FPT_TST	◔	◑	◑	◑

Classe Funcional	Família	Sigla	Nível 1	2	3	4
			Complexidade			
Utilização de Recursos	Tolerância a falhas	FRU_FLT		◔	◑	◕
	Prioridade do serviço	FRU_PRS	◔	◔	◔	◔
	Alocação de recursos	FRU_RSA	◔	◔	◔	◔

Classe Funcional	Família	Sigla	Nível 1	2	3	4
			Complexidade			
Acesso ao Sistema	Limitação da sessão	FTA_LSA	◔	◔	◔	◔
	Limitações em sessões concorrentes	FTA_MCS	◔	◑	◑	◑
	Bloqueio e encerramento de sessão	FTA_SSL	◔	◑	◕	◕
	Avisos de acesso	FTA_TAB		◔	◔	◔
	Histórico de acesso	FTA_TAH			◔	◔
	Estabelecimento de sessão	FTA_TSE		◔	◔	◔

Classe Funcional	Família	Sigla	Nível			
			1	2	3	4
			Complexidade			
Canais de Confiança	Canais de confiança entre sistemas	FTP_ITC	◔	◔	◔	◔
	Canais seguros	FTP_TRP	◔	◔	◔	◔

Tabela 16 - Grau de complexidade dos requerimentos para cada nível de segurança.

3.3.1. Instruções de leitura dos requerimentos de segurança

- Cada nível de segurança acumula os requerimentos dos níveis abaixo. Por exemplo, o nível 3 tem como pré-requisitos os requerimentos dos níveis 1 e 2.
- A falta de requisitos listados em um nível não o isenta das obrigações de segurança. Ou seja, mesmo que o nível 4 não apresente diretivas explícitas, obrigatoriamente esse terá que cumprir com os requerimentos de todos os níveis abaixo.
- Há casos em que não há requisitos para os níveis mais baixos. Por exemplo, quando não há diretrizes para o nível 1, o mesmo estará isento com aquela família funcional.

3.3.2. Auditoria de Segurança (FAU)

A classe de auditoria de segurança conta com as seguintes famílias:

- Resposta Automática de Auditoria (ARP)
- Geração de Dados de Auditoria (GEN)
- Análise da Auditoria de Segurança (SAA)
- Revisão da Auditoria de Segurança (SAR)
- Seleção de Eventos da Auditoria de Segurança (SEL)
- Armazenamento dos Eventos de Auditoria de Segurança (STG)

Família	**Resposta Automática de Auditoria**
Sigla	FAU_ARP
Nível 1	Todos os eventos que possam indicar violação de segurança deverão ser registrados.
Nível 2	Falhas em eventos de auditoria deverão ser registradas.

Nota: Todos os eventos anormais ao sistema deverão ser registrados. A partir do nível 2, caso haja falha em registros de auditoria, o sistema deverá sinalizar que houve falha no registro de eventos.

Família	Geração de Dados de Auditoria
Sigla	FAU_GEN
Nível 1	A utilização bem sucedida das funções administrativas deverá ser registrada.
Nível 2	Todas as tentativas de acesso às funções administrativas deverão ser registradas. A utilização bem sucedida de funções privilegiadas deverá ser registrada.
Nível 3	Cada ação registrada deverá ter a devida identificação do autor.
Nível 4	Todas as atividades deverão ser registradas, excetuando-se os valores de senhas. As marcas de tempo devem ter uma referência oficial bem definida e confiável.
Observações	Para todos os níveis de segurança, deverá ser criada uma lista de todos os itens a serem auditados. Esta lista deverá ser criada pelo desenvolvedor, e aprovada pelo requerente. As ações de segurança contempladas em cada nível deste manual devem ser, por padrão, registradas para que sejam passíveis de auditoria.

Nota: A referência de tempo pode vir de um relógio interno, declarado como referência da empresa (como um relógio de ponto) ou a partir da hora oficial brasileira, fornecida pelo Observatório Nacional.

Família	Análise da Auditoria de Segurança
Sigla	FAU_SAA
Nível 2	Com base no horário, duração, frequência ou conjunto de ações realizadas, deverá ser realizada uma análise potencial de violação de sistema.
Nível 3	Deverão ser estabelecidos padrões de uso do sistema para cada grupo de usuários e, caso haja incoerência ou desvio, deverá ser realizado registro.
Nível 4	O sistema deverá ter um detector de ataques, com base em possíveis sequências de invasão.

Nota: Os controles podem ser da própria aplicação ou do ambiente, como um sistema de detecção e prevenção de intrusos. O nível 2 tem avaliação simplificada e, como resposta, pode simplesmente acrescentar um registro de suspeita de ataque. O nível 3 conta com padrões e seção específica de registro.

Família	Revisão da Auditoria de Segurança
Sigla	FAU_SAR
Nível 1	Somente usuários explicitamente autorizados poderão acessar os registros de auditoria.
Nível 2	As funções de leitura e remoção de registros de auditoria deverão ser segregadas para usuários distintos.
Nível 4	Deverão ser categorizados os tipos de registros de auditoria para que sejam designados tipos de usuários com permissão específica para seu acesso.

Família	Seleção de Eventos da Auditoria de Segurança
Sigla	FAU_SEL
Nível 2	Deverão ser incorporadas ferramentas de seleção para facilitar a visualização dos registros de auditoria, como por exemplo a classificação dos eventos por ID, usuário, data/hora, etc.

Nota: Deverá haver uma forma de classificar e organizar os registros de auditoria. Poderão ser utilizados sistemas de apoio.

Família	Armazenamento dos Eventos de Auditoria de Segurança
Sigla	FAU_STG
Nível 2	Os registros de auditoria deverão ser contemplados com mecanismos de redundância e auto-verificação para evitar, detectar e corrigir alterações indevidas.
Nível 4	Os registros de auditoria não poderão ser removidos por usuários, mesmo os com direitos administrativos.
Observações	Para evitar problemas de espaço para armazenamento dos registros, podem ser consideradas possibilidades de compressão ou rotação automática, desde que obedecidos os prazos mínimos de armazenamento de documentos.

3.3.3. Comunicação (FCO)

A classe de comunicação conta com as famílias:

- Irretratabilidade da Origem (NRO)
- Irretratabilidade do Destinatário (NRR)

Família	Irretratabilidade da Origem
Sigla	FCO_NRO
Nível 2	O sistema deverá apresentar ao usuário garantias de que é, efetivamente, o sistema acessado.
Nível 3	Cada resposta enviada aos usuários ou armazenada pelo sistema deverá conter registro de tempo.
Tecnologia	Podem ser implementadas soluções com certificação digital para garantir a identidade dos sistemas.

Nota: As garantias podem ser realizadas, por exemplo, com segredos compartilhados ou certificados digitais.

Família	Irretratabilidade do Destinatário
Sigla	FCO_NRR
Nível 1	Cada usuário deverá ter identificação individual
Nível 2	As credenciais de acesso ao sistema não podem ser compartilhadas, por meios técnicos ou jurídicos.
Nível 4	Deverá ser realizado acesso utilizando, no mínimo, duplo fator de autenticação.
Tecnologia	Podem ser implementadas soluções com certificação digital para garantir a identificação dos usuários. Para o uso de múltiplos fatores de autenticação, deve ser considerado o armazenamento de certificados em hardware criptográfico, como tokens e smartcards. Dentre outros dispositivos de autenticação, podem ser utilizados recursos biométricos (impressão digital, vasculação da mão, retina, etc.) ou de uso único (OTP).

Nota: A maior dificuldade está no compartilhamento indevido de credenciais, que podem comprometer a irretratabilidade dos usuários/ sistemas remotos. Dentre as formas técnicas de evitar esse risco está na restrição de acessos simultâneos ou no uso de múltiplos fatores de autenticação.

3.3.4. Criptografia (FCS)

A classe de criptografia contém as famílias:

- Gerenciamento de Chaves Criptográficas (CKM)
- Operação Criptográfica (COP)

Família	Gerenciamento de Chaves Criptográficas
Sigla	FCS_CKM
Nível 1	Deverão ser estabelecidos mecanismos seguros de mercado para criação, distribuição, acesso e destruição de chaves criptográficas.
Nível 4	Deverá ser utilizado hardware criptográfico para funções de criação, armazenamento, uso e destruição de chaves criptográficas.
Tecnologia	Recomenda-se o uso de hardware criptográfico com certificação mínima NIST FIPS 140-2 nível 2 ou NSH-2, ou superior.

Família	Operação Criptográfica
Sigla	FCS_COP
Nível 1	Operações criptográficas devem ser transparentes em tráfego por firewalls e NAT.
Nível 2	Deverão ser empregados algoritmos criptográficos públicos e tamanhos de chaves reconhecidamente seguros.
Nível 3	As operações que envolvam o uso de chaves criptográficas deverão ser documentadas de forma detalhada.

Nota: Algoritmos privados podem oferecer risco por não terem sido publicamente testados.

3.3.5. Proteção de Dados do Usuário (FDP)

Esta classe contém as famílias:

- Política de Controle de Acesso (ACC)
- Funções de Controle de Acesso (ACF)
- Autenticação de Dados (DAU)
- Exportação de Dados (ETC)
- Política do Fluxo de Informação (IFC)
- Funções do Fluxo de Informação (IFF)
- Importação de Dados (ITC)
- Transferência Interna de Dados (ITT)
- Proteção da Informação Residual (RIP)
- Reversão (ROL)
- Integridade de Dados Armazenados (SDI)
- Confidencialidade dos Dados em Trânsito dos Usuários (UCT)
- Integridade de Dados em Trânsito dos Usuários (UIT)

Família	Política de Controle de Acesso
Sigla	FDP_ACC
Nível 1	Deverá ser estabelecida uma matriz de permissionamento entre os tipos de usuários e os tipos de informação disponibilizada pelo sistema.
Nível 2	Deverá ser estabelecida uma política de controle de acesso que defina os critérios para o gerenciamento de identidades, bem como as limitações e utilizações de canais de comunicação empregados pelo sistema.
Nível 3	Devem ser claros os objetivos de controle de acesso, relacionando objetos, atores e processos.

Nota: Essa família requer a documentação das permissões, que serão aplicadas de forma técnica.

Família	Funções de Controle de Acesso
Sigla	FDP_ACF
Nível 1	Com base na matriz de permissionamento, devem ser estabelecidos acessos sob o princípio do menor privilégio. Também deverão ser estabelecidos critérios de separação de funções, com regras baseadas na necessidade de acesso dos grupos de usuários e objetivos no sistema. Deverão ser agrupados os usuários com base nas funções de suporte ao sistema: gerenciamento do sistema, gerenciamento de configurações, programação, testes/validação de qualidade, segurança. Os grupos de administração, auditoria e segurança não podem ser acumulados.
Nível 2	Deverão ser definidas funcionalidades de controle de acesso, como: - Listas de Controle de Acesso (ACLs); - Especificações de controle de acesso com base de tempo; - Especificações de controle de acesso com base na origem; - Atributos de controle de acesso do proprietário.
Nível 3	Atributos de identificação deverão ser estabelecidos para usuários, objetos, e demais elementos de sistema que terão algum tipo de acesso, assim como seus grupos. Também deverão ser especificados os atributos de segurança requeridos para cada tipo de acesso.
Nível 4	Complementarmente, devem ser estabelecidas listas que explicitem os acessos que serão negados, permitidos e os casos de exceção.

Nota: Essa família deverá ter como base o documento gerado em FDP_ACC.

Família	Autenticação de Dados
Sigla	FDP_DAU
Nível 1	Os dados fornecidos pela aplicação aos usuários deverão ser passíveis de validação.
Nível 2	Deverão ser estabelecidos controles para a autenticação de dados, sendo criadas trilhas de auditoria.
Nível 3	Deverá haver controles que evidenciem a autenticidade dos dados críticos, de forma técnica ou operacional.
Nível 4	Dados críticos devem ser assinados com certificados digitais que garantam integridade, autenticidade e irretratabilidade.
Tecnologia	Podem ser utilizados algoritmos de resumo (hash) para que os dados sejam verificados. Considerando a atuação governamental, recomenda-se o uso de certificados digitais da ICP-BR.

Nota: O nível 1 não requer, de forma obrigatória, a autenticação dos dados para o usuário, a não ser que o sistema tenha essa demanda. Para a autenticação, podem ser utilizados verificadores de integridade, associados, por exemplo a marcas de tempo. Nos níveis superiores, a certificação digital pode ser uma forma de viabilizar essa autenticação.

Família	Exportação de Dados
Sigla	FDP_ETC
Nível 2	Por padrão, dados utilizados pelo sistema que sejam classificados como sensíveis ou confidenciais deverão ser explicitamente preservados ou simplesmente ignorados em processos de exportação.
Nível 3	Deverá ser definido um fluxo de exportação de dados. Caso sejam sensíveis ou confidenciais, deverão ser estabelecidos controles lógicos (como criptografia) ou físicos para restringir o acesso aos dados. Obrigatoriamente, os dados sensíveis ou confidenciais deverão estar acompanhados dos dados de usuário/ sistema, de forma integral, ou por referência direta. Os meios de exportação deverão utilizar controles que impeçam o vazamento de informações.
Nível 4	Deverão ser implementados mecanismos de verificação dos dados após a exportação, a fim de verificar sua integridade e autenticidade.
Observações	Caso seja necessário exportar dados sensíveis ou confidenciais, devem ser considerados os níveis de segurança 3 ou 4 deste item.

Nota: É importante ressaltar que os dados deverão ser exportados sob as mesmas condições de proteção em que se encontram no sistema.

Família	Política do Fluxo de Informação
Sigla	FDP_IFC
Nível 1	Deverá ser criada uma política que descreva os fluxos da informação, declarando suas origens, caminhos e destinos.
Nível 2	Deverão ser especificadas listas de sujeitos (usuários, computadores, processos), informações (e-mail, documentos) e operações que serão incluídos no fluxo de informação. Esse fluxo deverá fazer parte de uma política, onde são definidos os requerimentos de segurança para cada componente listado.
Nível 4	Deverá haver cláusulas específicas de tratamento e eliminação de fluxos ilícitos de informação.

Nota: O documento gerado nessa família deverá fornecer procedimentos para a proteção do fluxo de informação, incluindo a verificação de canais dissimulados. Podem ser definidos sistemas externos, como de prevenção de intrusos.

Família	Funções do Fluxo de Informação
Sigla	FDP_IFF
Nível 1	Com base na política de fluxos da informação, devem ser estabelecidos controles que garantam a integridade da informação em trânsito.
Nível 2	Deverão ser empregados controles para assegurar que as informações em trânsito sejam mantidas de forma privada.
Nível 3	Deverá haver monitoramento e, quando possível, bloqueio, de canais dissimulados ou de comunicações ilícitas a partir e para o sistema.
Nível 4	Toda informação transmitida deverá ter garantias quanto à autenticidade, confidencialidade, integridade e disponibilidade do emissor, receptor e dos canais de transmissão. De acordo com a política estabelecida, os canais ilícitos de comunicação deverão ser bloqueados.
Observações	Devem ser observados padrões de mercado para o estabelecimento das funções do fluxo de informação.

Família	Importação de Dados
Sigla	FDP_ITC
Nível 1	A importação de dados sem atributos de segurança deverá ocorrer de forma estruturada, observando seu formato.
Nível 2	Dados com atributos de segurança deverão ser importados para o sistema deverão ser verificados quanto ao seu formato e consistência.

Nota: A partir do nível 2, recomenda-se que os dados sejam validados antes da importação.

Família	Transferência Interna de Dados
Sigla	FDP_ITT
Nível 1	As informações transferidas dentro do próprio sistema devem ser verificadas quanto a sua integridade após a operação.
Nível 2	As informações transferidas internamente deverão ser classificadas, para que a transferência ocorra de acordo com seus atributos. Atributos de segurança sempre deverão acompanhar os dados dos usuários.
Nível 3	Deverão ser estabelecidos controles de fluxo para assegurar a segurança dos dados, seja de forma lógica ou física.
Nível 4	Deverão ser estabelecidos mecanismos de controle para prevenir erros ou desvio na transmissão dos dados.

Família	Proteção da Informação Residual
Sigla	FDP_RIP
Nível 1	Deverão ser listados os tipos de informação passíveis de exclusão.
Nível 2	Dados excluídos não poderão ser acessíveis, diretamente, por usuários ou administradores. Dados temporários armazenados em memória compartilhada ou em arquivos deverão ser sobrescritos ou removidos de forma segura.
Nível 3	Para recuperar dados excluídos que tenham ficado armazenados em cópias de segurança, deverão ser estabelecidos mecanismos de autorização por supervisores.
Nível 4	Toda informação residual deverá ter um tempo limite e um procedimento automatizado para sua remoção integral. Esse procedimento deve incluir a sanitização de mídias, que pode ser realizada de forma manual, desde que controlada por auditoria independente.

Família	Reversão
Sigla	FDP_ROL
Nível 1	Alterações no sistema deverão ser precedidas de mecanismos de backup para que seja possível reverter para o último ponto reconhecidamente funcional.
Nível 2	Todas as funções do sistema deverão ter condições de serem revertidas para a última configuração válida.
Nível 3	As informações utilizadas pelo sistema devem ser atendidas por sistemas de recuperação.
Nível 4	O tempo de recuperação para a última configuração válida, incluindo os dados manipulados pelo sistema, deverá inferior a 1 (um) minuto.

Nota: O nível 1 prevê o retorno do sistema para o último ponto funcional, o que seria equivalente a reinstalação de um sistema operacional e os service packs instalados anteriormente. O nível 2 requer que as configurações também sejam recuperadas mas, somente o nível 3 exigirá o pronto retornoda aplicação sem alteração nas informações. Isso significa que, para os níveis 1 e 2, poderá ocorrer perda de dados; enquanto para o 3 e 4 essa condição pode ser inaceitável

Família	Integridade de Dados Armazenados
Sigla	FDP_SDI
Nível 2	Os dados manipulados no sistema deverão ter monitoramento constante de integridade. As falhas deverão ser informadas aos responsáveis pela recuperação.
Nível 3	Devem ser estabelecidos controles automatizados para a recuperação de dados indevidamente alterados.

Nota: Podem ser utilizados, por exemplo, verificadores de integridade por algoritmos de hash por intervalos definidos.

Família	Confidencialidade dos Dados em Trânsito dos Usuários
Sigla	FDP_UCT
Nível 2	Dados de usuários deverão ser mantidos sob sistemas de controle contra acesso indevido e vazamento.
Nível 3	Os dados de usuários deverão ser protegidos de modo que, mesmo em caso de falha dos controles dos canais, não sejam expostos.
Nível 4	Dados de usuários deverão ser permanentemente criptografados, bem como os meios de transmissão.

Família	Integridade de Dados em Trânsito dos Usuários
Sigla	FDP_UIT
Nível 2	Dados de usuários deverão ser avaliados na origem e verificados no destino.
Nível 3	Em caso de falha, o sistema deverá efetuar correções automatizadas no canal de comunicação, nas partes ou na própria informação transmitida.

Nota: O nível 2 não obriga o sistema a recuperar automaticamente, mas recomenda-se que o mesmo falhe com elegância, solicitando ao usuário que tente realizar a operação novamente.

3.3.6. Identificação e Autenticação (FIA)

A classe FIA contém as famílias:

- Falhas de Autenticação (AFL)
- Definição de Atributos de Usuários (ATD)
- Especificação de Segredos (SOS)
- Autenticação de Usuários (UAU)
- Identificação de Usuários (UID)
- Ligação usuário-assunto (USB)

Família	Falhas de Autenticação
Sigla	FIA_AFL
Nível 1	Após a segunda falha de autenticação com a mesma credencial, o usuário deverá ser encaminhado para sistema de recuperação de credenciais.
Nível 2	Após a terceira tentativa de autenticação sem sucesso, com uma mesma credencial, o usuário deverá ter seu acesso bloqueado por 15 minutos, ou conforme a política local.
Nível 3	Após a quinta tentativa de acesso sem sucesso, independente da credencial utilizada, o acesso ao sistema deverá ser bloqueado por 15 minutos para o sistema/ computador de origem.
Nível 4	O desbloqueio de credenciais poderá ser realizado somente mediante procedimento administrativo formal.

Nota: Essas medidas previnem ataques de força bruta.

Família	Definição de Atributos de Usuários
Sigla	FIA_ATD
Nível 2	Os atributos de segurança de um usuário deverão ser relacionados diretamente a esse. Dessa forma, espera-se que, uma alteração em um usuário não gere impactos em outros.

Nota: Isso significa que, por exemplo, as chaves de um usuário são exclusivas.

Família	Especificação de Segredos
Sigla	FIA_SOS
Nível 1	O usuário poderá definir segredos que serão utilizados para autenticação no sistema. Esses segredos devem ter complexidade mínima e prazo de validade definida por política, e verificação pelo sistema.
Nível 2	A geração de segredos também poderá ser realizada pelo sistema, de forma aleatória, obedecendo minimamente aos critérios definidos por política.
Nível 4	Segredos deverão ser gerados em hardware de segurança ou em ambiente que ofereça segurança de mesmo nível.

Nota: Os segredos poderão ser gerados pelo sistema, por exemplo, nos casos de senhas temporárias. A partir do nível 2, podem ser estabelecidos (pelo sistema) segredos de identificação exclusiva do usuário.

Família	Autenticação de Usuários
Sigla	FIA_UAU
Nível 1	A autenticação de usuários deverá ocorrer no momento em que for realizada uma solicitação que exija o acesso a áreas consideradas restritas. (**Aplicável somente ao nível 1 de segurança**). Todo segredo entrado pelo usuário deverá ser oculto ou exibido de forma mascarada. Os segredos devem ser sempre armazenados/ manipulados em forma criptografada ou em hash. A verificação de uma credencial, para que seja concedido acesso a um recurso, deverá obedecer a sequência: 1. Se o modo de acesso utilizado para acessar o recurso é proibido, o acesso deverá ser negado; 2. Se o modo de acesso utilizado para acessar o recurso é permitido, o acesso deverá ser concedido; 3. Se o modo de acesso utilizado para acessar o recurso é proibido para todos os grupos os quais o usuário é membro, o acesso deverá ser negado; 4. Se o modo de acesso utilizado para acessar o recurso é permitido para qualquer grupo o qual o usuário é membro, o acesso deverá ser concedido; 5. Se o modo de acesso utilizado para acessar o recurso é proibido ao público, o acesso deverá ser negado; 6. Se o modo de acesso utilizado para acessar o recurso é permitido, o acesso deverá ser concedido; 7. Qualquer outra tentativa de acesso deverá ser negada.
Nível 2	Usuários anônimos não serão autorizados. A autenticação de usuários deverá ocorrer imediatamente após a apresentação de um aviso sobre o uso e as responsabilidades do usuário no sistema. Nenhuma outra função de consulta ou navegação poderá ser realizada anteriormente à autenticação. Um mesmo usuário não poderá ter acesso concomitante ao sistema. Ao acessar, o mesmo será avisado sobre o número de acessos efetuados, data e hora do último acesso válido e a validade de suas credenciais.
Nível 3	O sistema deverá monitorar as origens do acesso do usuário para registrar se, em curto espaço de tempo, há acessos em diferentes origens. Tal registro deverá gerar um alerta de suspeita de roubo ou compartilhamento de credenciais. Deverá ser exigida re-autenticação do usuário após determinado período de tempo ou para a confirmação de operações que gerem alterações em dados.

Nível 4	O uso de múltiplos fatores de autenticação (pelo menos duplo) será obrigatório para acessar o sistema e realizar operações de alterações em dados.

Nota: O nível 1 permite o uso anônimo de determinadas funções do sistema, que deverão ser documentadas, conforme solicita a família FIA_UID.

Família	Identificação de Usuários
Sigla	FIA_UID
Nível 1	Deverá ser preparada uma lista de ações que o sistema poderá realizar pelo usuário antes de sua identificação. Para todas as outras ações, o usuário deverá ser identificado antes de qualquer execução.

Família	Ligação usuário-assunto
Sigla	FIA_USB
Nível 1	Por padrão, nenhum usuário poderá realizar uma ação em nome de outro.
Nível 2	Todas as ações realizadas pelos usuários serão vinculadas aos mesmos. As operações administrativas que envolvam a alteração de atributos de segurança dos usuários deverão ser listadas e verificadas.

Nota: Essas medidas evitam, por exemplo, que um administrador execute uma ação em nome de outro usuário.

3.3.7. Gerenciamento de Segurança (FMT)

A classe de Gerenciamento de Segurança contém as famílias:

- Gerenciamento de funções do sistema (MOF)
- Gerenciamento de atributos de segurança (MSA)
- Gerenciamento de dados das funções de segurança (MTD)
- Revogação (REV)
- Validade dos atributos de segurança (SAE)
- Especificação das funções de gerenciamento (SMF)
- Regras de gerenciamento de segurança (SMR)

Família	Gerenciamento de funções do sistema
Sigla	FMT_MOF
Nível 1	Deverá haver uma matriz de funções de usuários, segregando atividades de administração, configuração de segurança e auditoria.
Nível 3	O sistema deverá contar, nativamente, com usuários de administração de segurança independentes, ou seja, que possuam funções distintas do administrador do sistema e que não possam receber interferência desse. Assim, por exemplo, novos usuários de administração de segurança poderão ser criados apenas pelo primeiro administrador de segurança.

Nota: As atividades de administração, gerenciamento de segurança e auditoria não devem ser acumuladas pelos mesmos usuários. Na segregação do nível 3, recomenda-se que os usuários nativos tenham direitos de criar novos usuários das mesmas categorias. Por exemplo, auditores só poderão ser criados a partir de auditores.

Família	Gerenciamento de atributos de segurança
Sigla	FMT_MSA
Nível 2	Deverão ser listadas as funções que criam, alteram ou removem atributos de segurança de usuários e funções do sistema.

Família	Gerenciamento de dados das funções de segurança
Sigla	FMT_MTD
Nível 1	Somente administradores de segurança poderão alterar dados das funções de segurança.
Nível 2	Qualquer alteração sobre as propriedades de usuários com esse poder de administração deverá ser registrada de forma indelével.

Família	Revogação
Sigla	FMT_REV
Nível 1	Deverá haver confirmação dos direitos do usuário a cada entrada (login) no sistema. Assim, por exemplo, um usuário que tenha privilégios revogados deverá perdê-los, minimamente, na próxima autenticação no sistema.
Nível 2	Deverá haver confirmação dos direitos do usuário a cada ação no sistema. Assim, por exemplo, um usuário que tenha privilégios revogados deverá perdê-los, minimamente, no próximo acesso a arquivo.
Nível 4	Deverá haver confirmação dos direitos do usuário a cada 10 minutos. Assim, por exemplo, um usuário que tenha privilégios revogados deverá perdê-los, no máximo, em 10 minutos.

Nota: Essas configurações permitem que um usuário não realize ações caso seja excluído do sistema, ou tenha seus direitos revogados. No nível 1, no entanto, essa revogação valerá apenas a partir da próxima autenticação no sistema.

Família	Validade dos atributos de segurança
Sigla	FMT_SAE
Nível 1	Atributos de segurança deverão ter limitação de uso por tempo, de acordo com a política institucional. Dentre os atributos de segurança com validade, podem ser observadas senhas, certificados e prazo de uso em um sistema.
Nível 2	Quando aplicável, os usuários deverão ser avisados previamente sobre a necessidade de renovação dos atributos de segurança. Esse aviso deverá obedecer a política institucional, ou, minimamente, 15 dias antes da expiração. Deverão ser estabelecidos mecanismos de renovação automática.

Família	Especificação das funções de gerenciamento
Sigla	FMT_SMF
Nível 1	Deverá ser criada uma lista com todas as funções de gerenciamento de segurança disponibilizada pelo sistema desenvolvido.

Nota: Essa família gerará um documento que será utilizado na validação do sistema.

Família	Regras de gerenciamento de segurança
Sigla	FMT_SMR
Nível 1	Deverão ser definidas regras para auditores, administradores, administradores de segurança e outros usuários.

Nota: Essa família gerará um documento que será utilizado na validação do sistema. Essas regras, após implantadas, devem ter condições de serem auditadas.

3.3.8. Privacidade (FPR)

Essa classe contém as famílias:

- Anonimato (ANO)
- Pseudoanonimato (PSE)
- Desvinculação (UNL)
- Restrição à interceptação (UNO)

Família	Anonimato
Sigla	FPR_ANO
Nível 1	As funções de segurança não poderão guardar rastros dos usuários, preservando assim, suas identidades perante o sistema e outros usuários.
Nível 3	As funções de segurança deverão ser proibidas de solicitar as credenciais do usuário.
Observações	Essa medida previne ataques de replay, bem como mantém sob anonimato consultas a dados provados, por exemplo.

Nota: As evidências de utilização dos sistemas deverão estar registradas somente para fins de auditoria. As funções de segurança deverão ser incapazes de informar as ações do usuário. A partir do nível 3, o sistema deverá conter mecanismo interno de autenticação do usuário, evitando que as funções de segurança solicitem diretamente ao mesmo. Essa medida reduz o risco de uma escuta, por exemplo.

Família	Pseudoanonimato
Sigla	FPR_PSE
Nível 2	Os usuários deverão ser tratados por referências internas para garantir a auditabilidade. Somente auditores poderão ter acesso aos registros de uso do sistema e suas funções por um usuário.

Família	Desvinculação
Sigla	FPR_UNL
Nível 2	Múltiplas atividades de um usuário não poderão ser correlacionadas, a não ser que configurem um perfil de atividade maliciosa.

Nota: Por padrão, um usuário não deverá ser "perseguido" pelo sistema. No entanto, se suas ações configurarem um ataque (conforme definido na classe FAU), deverá haver registro.

Família	Restrição à interceptação
Sigla	FPR_UNO
Nível 1	As credenciais de acesso nunca poderão trafegar em formato aberto, desprotegido.
Nível 2	O sistema deverá garantir que as informações dos usuários, bem como dados sigilosos sejam trafegados de forma segura, de modo a impedir que usuários não autorizados ou convidados interceptem esses dados.
Nível 3	De forma adicional, o sistema poderá ser estruturado para não solicitar informações sigilosas a outros sistemas.
Nível 4	Deverão ser definidos auditores com a capacidade de analisar a utilização de recursos pelos usuários.

Nota: Esses requerimentos restringem, por exemplo, a escuta com *sniffers*.

3.3.9. Proteção das Funcionalidades de Segurança (FPT)

As famílias dessa classe são:

- Falha segura (FLS)
- Disponibilidade de dados exportados da aplicação (ITA)
- Confidencialidade de dados exportados da aplicação (ITC)
- Integridade dos dados exportados da aplicação (ITI)
- Segurança nas transferências internas de dados (ITT)
- Segurança física para a aplicação (PHP)
- Recuperação confiável (RCV)
- Detecção de repetição (replay) (RPL)
- Protocolo de sincronia de estado (SSP)
- Carimbos de tempo (STM)
- Consistência de dados (TDC)
- Teste de sistemas externos (TEE)
- Consistência na replicação de dados (TRC, incorporado ao TDC)
- Auto-teste da aplicação (TST)

Família	Falha segura
Sigla	FPT_FLS
Nível 1	O sistema deverá cuidar das falhas sem externalizar os erros, e sem fornecer informações que podem ser usadas para exploração. Dados pessoais e confidenciais nunca deverão ser incluídos em mensagens de erro.
Nível 3	Mensagens de erro deverão ser exibidas apenas para pessoal autorizado.

Nota: Essas medidas reforçam a segurança do sistema a aumentam o conforto do usuário.

Família	Disponibilidade de dados exportados da aplicação
Sigla	FPT_ITA
Nível 2	Dados exportados para outros sistemas confiados deverão ser mantidos sob as mesmas condições de disponibilidade que o sistema de origem.

Família	Confidencialidade de dados exportados da aplicação
Sigla	FPT_ITC
Nível 2	Dados exportados para outros sistemas confiados deverão ser mantidos sob as mesmas condições de privacidade que o sistema de origem.
Nível 3	Deverão ser implementados controles criptográficos para a exportação de dados.
Nível 4	Também deverão ser definidos controles físicos para a exportação de dados. O sistema deverá manter a privacidade dos dados em casos de agregação, empacotamento e transformação.

Família	Integridade dos dados exportados da aplicação
Sigla	FPT_ITI
Nível 2	Dados exportados deverão ser verificados no seu destino quanto a sua integridade e, em caso de falhas, deverá haver um mecanismo automatizado de correção.
Tecnologia	Podem ser utilizados controles por algoritmos de hash, como o SHA1 ou o MD5

Família	Segurança nas transferências internas de dados
Sigla	FPT_ITT
Nível 1	Os dados manipulados pelo sistema deverão ser sempre mantidos sob o mesmo nível de integridade e confidencialidade exigidos pelo nível de segurança.

Família	Segurança física para a aplicação
Sigla	FPT_PHP
Nível 1	Deverá ser estabelecido um levantamento de riscos para definir as áreas que serão definidas como públicas e restritas. Deverá haver controle de acesso físico às áreas em que o sistema será instalado, excetuando-se os casos em que o sistema será liberado ao público. Dentre os controles, deverão ser estabelecidos registros de entrada e saída de pessoas, com denotação de um responsável interno pelo visitante. Deverá haver controles de inventários sobre os ativos físicos, incluindo mídias, bem como de chaves e fechaduras. São obrigatórias as especificações dos procedimentos que serão adotados em caso de perda/ roubo desses.
Nível 2	Deverá ser estabelecido controle de acesso compartimentalizado, com verificação de permissões de acesso dos usuários e visitantes aos locais de instalação dos sistemas. Serão mandatórios controles de climatização (temperatura, umidade) e de combate a incêndio. Recomenda-se o uso de sensores de invasão física.
Nível 3	Deverão ser estabelecidos controles de acesso físico que incluam registro em circuito fechado de TV, ou similar, nas áreas seguras. Também deverão ser implantados múltiplos fatores de controle de acesso físico, como crachás, biometria, senhas e seguranças.
Nível 4	O sistema deverá ser instalado em sala segura. Para acessar o sistema em que foi instalada a aplicação, serão necessárias, no mínimo, duas pessoas. Cabe ressaltar que as funções e as permissões de acesso dessas pessoas deverá ser replicado.

Família	Recuperação confiável
Sigla	FPT_RCV
Nível 1	Deverá haver um processo bem definido de recuperação do sistema em caso de falha. Deverão ser definidos responsáveis pela recuperação.
Nível 2	Deverão ser empregados mecanismos de reversão para a última configuração válida em sistemas de transação.
Nível 3	Deverá haver controles para a geração de imagens dos sistemas e seus ambientes operacionais. Prazos de recuperação deverão ser definidos por política.
Nível 4	Deverão ser utilizados sistemas de redundância e supressão de erros em tempo real.
Tecnologia	Rollback de transações e journaling são medidas que podem ser implementadas para suportar operações de recuperação em sistemas baseados em bancos de dados.

Nota: Considera-se que as aplicações de nível 4 possuem disponibilidade total, enquanto as de nível 1 podem ficar indisponíveis de forma temporária. De qualquer modo, deverá haver documentação formal e instrumentos técnicos adequados para a recuperação dos sistemas.

Família	Detecção de repetição (replay)
Sigla	FPT_RPL
Nível 2	Cada mensagem transmitida pelo sistema deverá conter um identificador único, processado somente uma vez. O sistema deverá verificar se há tentativas de envio de mensagens com os mesmos identificadores.

Nota: Os identificadores únicos em uma transação podem, por exemplo, ser gerados de forma aleatória, e o sistema deverá aceitar aquela sequência somente uma única vez.

Família	Protocolo de sincronia de estado
Sigla	FPT_SSP
Nível 2	As ações de segurança deverão ser monitoradas e replicadas em tempo real.
Nível 3	Deverão ser implantados controles que avaliem a integridade do sistema de forma periódica e, em caso de alteração indevida, deverá haver mecanismos automatizados de recuperação.

Nota: Para evitar inconsistências de permissões, configurações ou demais atributos ou ações de segurança, deverá haver controles que sincronizem múltiplas cópias do sistema.

Família	Carimbos de tempo
Sigla	FPT_STM
Nível 2	Deverão ser empregados mecanismos de carimbo de tempo, com base em relógios confiáveis.
Nível 4	Recomenda-se o uso de carimbos de tempo com sincronização no horário oficial brasileiro, fornecido pelo Observatório Nacional. Deverão ser observadas as regulamentações do ITI sobre o assunto.
Observações	Mais informações: - RFC 3628: http://www.faqs.org/rfcs/rfc3628.html - ICP BR documento 10: http://www.iti.gov.br/twiki/pub/Certificacao/DocIcp/DOC-ICP-11_-_Versao_1.2.pdf

Nota: Consideram-se relógios confiáveis aqueles adotados, de maneira formal, como a referência de tempo oficial da empresa. A regulamentação dos carimbos de tempo digitais é de responsabilidade do ITI, que deverá ser consultado para mais informações.

Família	Consistência de dados
Sigla	FPT_TDC
Nível 1	Deverão ser implementados controles que evitem a edição simultânea de dados.
Nível 2	Controles de replicação deverão ser estabelecidos a partir de uma única origem de dados.

Nota: Dentre os controles, podem ser adotadas medidas em que um usuário possa editar um documento e outros tenham somente acesso como leitura. Para replicação, deverá ser adotada uma origem primária dos dados para que, a partir dessa, sejam feitas as cópias.

Família	Teste de sistemas externos
Sigla	FPT_TEE
Nível 1	Deverão ser estabelecidos levantamentos de riscos quando forem utilizados sistemas externos, como firewalls e sistemas operacionais. Deverão ser conduzidos, por exemplo, testes de invasão nos sistemas externos para validar sua confiabilidade. Além disso, deverão ser estabelecidos documentos afirmando as responsabilidades do cedente do sistema.
Nível 2	Sistemas externos deverão ser avaliados quanto a compatibilidade de suas funções de segurança com os requerimentos de segurança do nível da aplicação.

Família	Auto-teste da aplicação
Sigla	FPT_TST
Nível 1	O sistema deverá ser contemplado com funções automatizadas de testes de integridade. Caso os testes falhem, deverão ser acionados os mecanismos de recuperação segura.
Nível 2	Todos as funções de segurança deverão ser testadas automaticamente em intervalos regulares.

Nota: No nível 1, deverá ser possível testar as funcionalidades de segurança quanto a sua integridade. A partir do nível 2, os testes serão automatizados, em intervalos regulares, podendo haver recuperação automática em caso de falhas.

3.3.10. Utilização de Recursos (FRU)

Essa classe contém as famílias:

- Tolerância a falhas (FLT)
- Prioridade do serviço (PRS)
- Alocação de recursos (RSA)

Família	Tolerância a falhas
Sigla	FRU_FLT
Nível 2	O sistema deverá contar com mecanismos de tolerância a falhas em caso de falha de hardware e/ou energia.
Nível 3	O sistema deverá contar com mecanismos de redundância do ambiente de software e conectividade.
Nível 4	O sistema deverá contar com mecanismos que o mantenha operacional em caso de falha própria.

Nota: De um modo geral, deverão ser empregados mecanismos de redundância de hardware, energia, sistemas operacionais, links de rede, e de sistemas.

Família	Prioridade do serviço
Sigla	FRU_PRS
Nível 1	O sistema deverá limitar o uso aos recursos de acordo com a prioridade de execução. Deverá ser criada uma lista de prioridades das funções do sistema

Família	Alocação de recursos
Sigla	FRU_RSA
Nível 1	Deverão ser estabelecidas cotas de uso de recursos, como memória, disco e processamento para usuários e funções.
Observações	Essa medida previne contra ataques de negação de serviço (DoS)

3.3.11. Acesso ao Sistema (FTA)

A classe FTA contém as famílias:

- Limitação da sessão (LSA)
- Limitações em sessões concorrentes (MCS)
- Bloqueio e encerramento de sessão (SSL)
- Avisos de acesso (TAB)
- Histórico de acesso (TAH)
- Estabelecimento de sessão (TSE)

Família	Limitação da sessão
Sigla	FTA_LSA
Nível 1	O sistema deverá limitar o escopo da sessão do usuário com base em seus direitos e regras.

Nota: Essa família solicita que um usuário de direitos restritos não possa navegar, por exemplo, em telas administrativas, mesmo que não consiga salvar suas alterações.

Família	Limitações em sessões concorrentes
Sigla	FTA_MCS
Nível 1	Os sistema deverá ter um número limitado de sessões concorrentes, definido com base na disponibilidade de recursos do ambiente.
Nível 2	Um usuário poderá ter somente uma sessão por acesso, não sendo permitidas sessões concorrentes.

Nota: Essas medidas restringem o compartilhamento de credenciais, que deixariam um usuário sem poder trabalhar, caso outra pessoa utilize seu login. Da mesma forma, essa medida permite identificar se houve roubo de credenciais, caso o usuário conectado tenha prioridade sobre o que está tentando acessar.

Família	Bloqueio e encerramento de sessão
Sigla	FTA_SSL
Nível 1	As conexões deverão ser encerradas após 30 minutos de inatividade, ou conforme política local.
Nível 2	O sistema deverá ser bloqueado após, pelo menos, 15 minutos de inatividade. O usuário poderá restabelecer a sessão mediante reautenticação, e seu trabalho em andamento poderá ser reaproveitado.
Nível 3	O bloqueio do sistema deverá alterar a exibição da tela, para que não seja possível visualizar o conteúdo que estava sendo trabalhado.
Observações	Os tempos poderão ser alterados conforme a necessidade ou aplicação de políticas locais.

Nota: A partir do nível 2, espera-se que o usuário possa se autenticar e retomar seu trabalho a partir do ponto em que parou. No nível 1, as informações poderão ser perdidas.

Família	Avisos de acesso
Sigla	FTA_TAB
Nível 2	Antes de fornecer acesso à aplicação, deverá ser exibida uma mensagem, informando que: 1. O usuário está acessando um sistema governamental, que deverá ser utilizado de acordo com a legislação vigente; 2. O sistema é monitorado e que, ao acessar, o usuário concorda que algumas de suas ações possam ficar registradas para aplicações legais; 3. O uso não autorizado do sistema é proibido.

Família	Histórico de acesso
Sigla	FTA_TAH
Nível 2	O sistema deverá informar ao usuário a data e hora do último acesso realizado em seu nome.

Nota: O histórico de acesso permite que os usuários monitorem o uso das suas credenciais. Recomenda-se a realização de seminários de conscientização em segurança, para que o usuário entenda que deverá relatar incidentes.

Família	Estabelecimento de sessão
Sigla	FTA_TSE
Nível 2	De acordo com a política local, o sistema deverá ter restrições, considerando: 1. Origem do acesso (ex. proibido terminal, celular, remoto, etc.); 2. Horário do acesso (ex. proibido das 19 às 6 h); 3. Método de acesso (ex. x-windows);

3.3.12. Canais de Confiança (FTP)

Há duas famílias na classe FTP:

- Canais de confiança entre sistemas (ITC)
- Canais seguros (TRP)

Família	Canais de confiança entre sistemas
Sigla	FTP_ITC
Nível 1	Sistemas externos deverão ser compatíveis com as políticas definidas para o nível de segurança da aplicação. Essa compatibilidade deverá ser formalizada com documentos gerenciais.

Família	Canais seguros
Sigla	FTP_TRP
Nível 1	O sistema deverá estabelecer um canal seguro entre o usuário e suas funções de segurança, como por exemplo, no momento da autenticação.

4. Avaliação e validação de aplicações

4.1. Avaliação de sistemas

4.1.1. Objetivos

A avaliação de sistemas tem como objetivos:

- Verificar se o sistema funciona conforme previsto/ documentado;
- Corrigir falhas antes da implantação;
- Validar os controles de segurança.

4.1.2. Preparação para a avaliação de sistemas

A etapa de avaliação consiste na verificação entre o produto desenvolvido/ adquirido e os requerimentos de segurança para o nível estipulado para esse.

Por padrão, para que um sistema seja validado dentro do seu nível, todos os requerimentos deverão ser atendidos. Quando o atendimento pleno não ocorrer, o sistema deverá estar respaldado por documentos que evidenciem as escolhas adotadas por desenvolvedores e gestores.

O principal documento que alinhará as expectativas entre desenvolvedores e avaliadores é o de objetivos de avaliação, que contém todas as ressalvas e peculiaridades do sistema que precisarão ser avaliadas.

4.1.3. Critérios de avaliação

Um avaliador poderá indicar um sistema para autorização somente caso todos os requerimentos listados para serem avaliados no documento de objetivos de avaliação tenham sido plenamente atendidos.

4.1.4. *Framework* de avaliação

O *framework* de avaliação de aplicações pode ser aplicado em todas as fases do ciclo de vida do desenvolvimento de aplicações, além de ser utilizado para a validação final do sistema.

A partir da revisão dos objetivos de segurança, há três etapas de avaliação a serem percorridas:

1) Validação dos pontos de controle;

2) Avaliação da documentação do sistema;

3) Testes do sistema de acordo com a documentação.

Em cada etapa, os resultados devem ser pareados para verificar se o sistema foi construído e é executado conforme previsto pela documentação, além de verificar se a descrição das funcionalidades está correta.

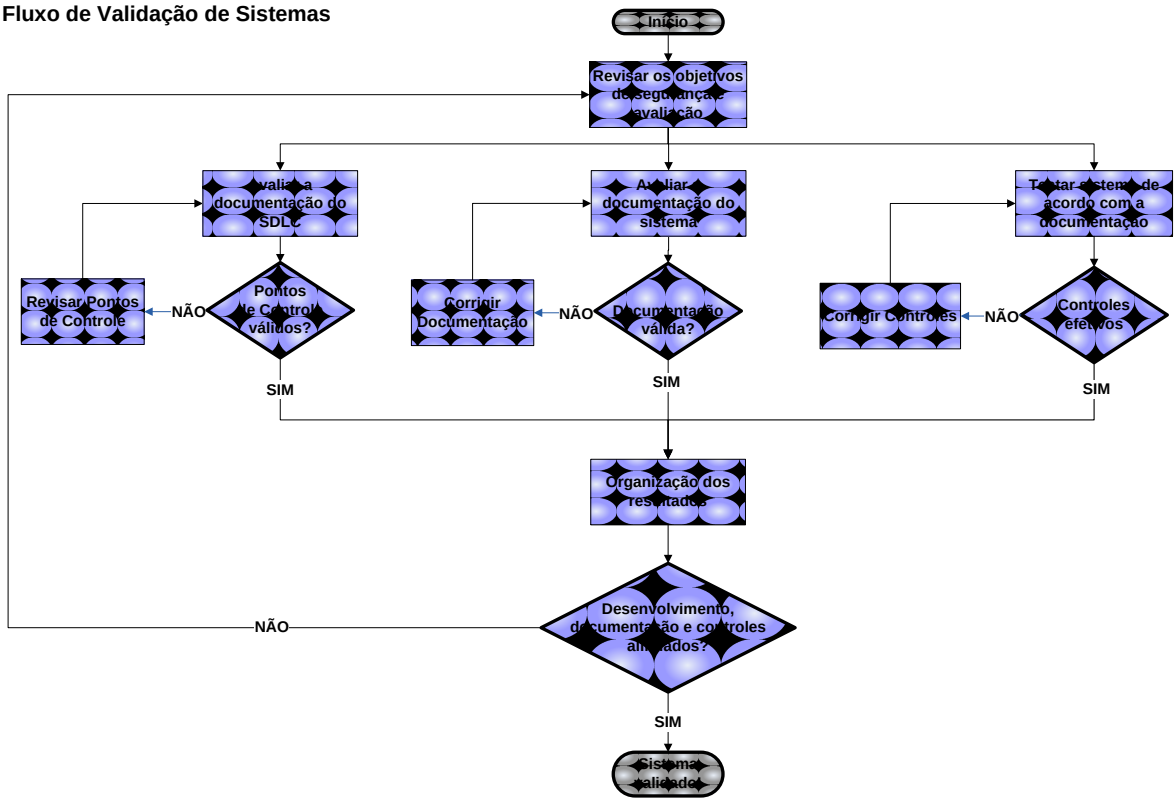


Figura 12- Fluxo de validação de sistemas.

O documento base para realizar a avaliação do sistema deverá ser gerado durante o desenvolvimento, entitulado neste manual como “Plano detalhado para certificação”, que orientará o avaliador quanto aos itens do sistema que devem ser testados.

É um documento que requer atenção do desenvolvedor, porque todos os itens de segurança implantados têm que estar discriminados, bem como todas as ressalvas.

4.2. Sistemas externos ou adquiridos

Caso um sistema esteja sendo adquirido, dificilmente haverá acesso à documentação do SDLC.

Portanto, sistemas adquiridos deverão ser avaliados quanto a sua documentação e deverão ser realizados os testes de segurança para confirmar a existência dos controles requeridos.

Em alguns casos, o fornecedor poderá entregar a documentação dos controles e seus testes, cabendo ao comprador a verificação da veracidade das informações.

4.3. Auditoria do SDLC

4.3.1. Considerações para a preparação do plano de auditoria

Um avaliador deverá considerar a inclusão dos seguintes itens ao criar um plano de auditoria:

- O modo de compra ou desenvolvimento da aplicação, tecnologia, porte, objetivos e uso do sistema;
- Estrutura do projeto para aquisição e implantação;
- Competência e experiência da equipe de desenvolvimento;
- Possíveis riscos;
- Considerações dos gestores;
- O estágio de desenvolvimento;
- Avaliações anteriores, mesmo em outros estágios.

4.3.1.1. Aspectos a serem revisados

O avaliador deverá estudar os itens relevantes para cada fase, e oferecer propostas de melhorias, dentre os seguintes aspectos:

- Planejamento do projeto (plano, entregáveis, calendário), e os benefícios trazidos pelo sistema desenvolvido;
- Viabilidade financeira do projeto;
- A estrutura do projeto, incluindo pessoas e suas responsabilidades;
- A metodologia de gerenciamento de projetos adotada;
- A metodologia de desenvolvimento aprovada;
- Contratos com fornecedores de aplicações e serviços;
- Processos de controle do modelo de SDLC: revisões, validações e autorizações em cada fase do desenvolvimento;
- Estrutura dos entregáveis de cada fase do SDLC avaliada;
- Atas de reuniões;
- Os entregáveis, suas revisões e autorizações;
- Documentos de acompanhamento do projeto (recursos, tempo, custos);
- Gerenciamento de recursos;
- Dados da gestão de riscos;
- Gerenciamento de qualidade;
- Gerenciamento de mudanças;
- Gerenciamento de performance e problemas, incluindo SLAs;
- Gerenciamento de configurações;
- Conversão/ migração de dados;
- Leis, normas e padrões que devem ser utilizados como referência para o desenvolvimento.

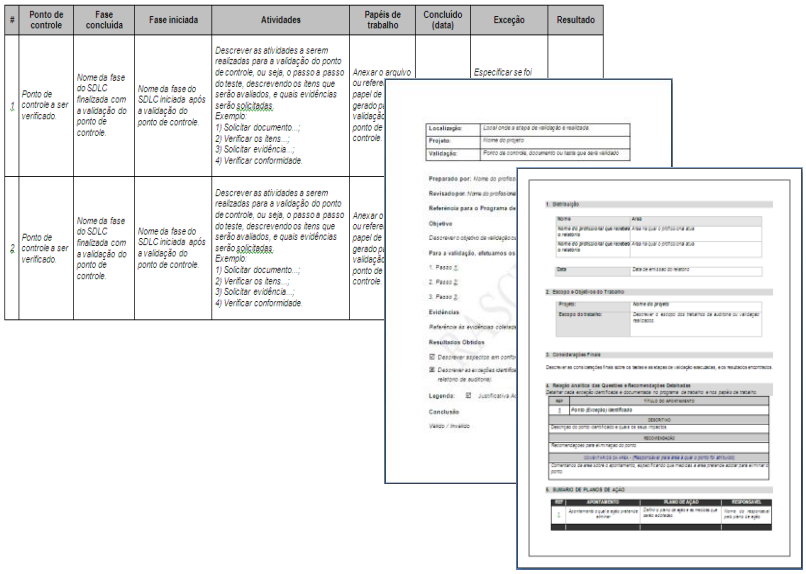
4.3.1.2. Revisão dos pontos de controle

Os pontos de controle ao final de cada fase geram documentação para o acompanhamento da auditoria. Esses documentos viabilizam que a avaliação dos sistemas acompanhe o desenvolvimento desde a fase de concepção.

4.3.2. Documentos da auditoria

Os documentos de suporte à auditoria são:

- **Programa de trabalho**: Visão geral da avaliação dos pontos de controle, da documentação do sistema e da realização dos testes. Deverá ser criado para cada avaliação, de forma específica. De um modo geral, o programa de trabalho marca as aprovações e exceções dos itens avaliados.
- **Papéis de trabalho**: Descrevem as etapas de avaliação de documentos e se os objetivos dessas foram alcançados.
- **Relatório de auditoria**: Documento que aponta as falhas encontradas durante a execução de uma determinada atividade ou conjunto de atividades, e indica responsáveis pela execução de planos de ação.



#	Ponto de controle	Fase concluída	Fase iniciada	Atividades	Papéis de trabalho	Concluído (data)	Exceção	Resultado
1.	*Ponto de controle a ser verificado.*	*Nome da fase do SDLC finalizada com a validação do ponto de controle.*	*Nome da fase do SDLC iniciada após a validação do ponto de controle.*	*Descrever as atividades a serem realizadas para a validação do ponto de controle, ou seja, o passo a passo do teste, descrevendo os itens que serão avaliados, e quais evidências serão solicitadas. Exemplo: 1) Solicitar documento...; 2) Verificar os itens ...; 3) Solicitar evidência...; 4) Verificar conformidade.*	*Anexar o arquivo ou referê papel de gerado pe validação ponto de controle.*		*Especificar se foi*	
2.	*Ponto de controle a ser verificado.*	*Nome da fase do SDLC finalizada com a validação do ponto de controle.*	*Nome da fase do SDLC iniciada após a validação do ponto de controle.*	*Descrever as atividades a serem realizadas para a validação do ponto de controle, ou seja, o passo a passo do teste, descrevendo os itens que serão avaliados, e quais evidências serão solicitadas. Exemplo: 1) Solicitar documento...; 2) Verificar os itens ...; 3) Solicitar evidência...; 4) Verificar conformidade.*	*Anexar o ou referê papel de gerado pe validação ponto de controle.*			

Modelos desses documentos estão contidos nos anexos deste manual.

Dentre os fatores a serem abordados, o relatório de auditoria do SDLC deverá conter:

- Viabilidade econômica do projeto (quando a avaliação é realizada no momento do desenho/ concepção do sistema;
- Riscos ao SDLC, suas causas e efeitos;
- Ações de mitigação dos riscos nos diferentes estágios do SDLC.

4.4. Avaliação da documentação do sistema

4.4.1. Objetivos

A avaliação da documentação deve:

- Verificar se a documentação é clara, direta e descreve, precisamente, objetivos, funções e procedimentos do sistema;
- Confirmar que o sistema foi desenvolvido de acordo com o planejamento e dentro de um sistema formal de gestão de mudanças;
- Validar manuais de administração e uso do sistema, com orientações de interpretação única.

4.4.2. Documentação

Assim como os requerimentos de segurança há classes para a validação de sistemas, que exigem a criação de documentos. São as classes:

- Avaliação das Metas de Segurança (ASE)
- Desenvolvimento (ADV)
- Documentação (AGD)
- Ciclo de Vida (ALC)
- Testes (ATE)
- Vulnerabilidades (AVA)

A documentação exigida pode ser distribuída em três modalidades, conforme mostra a tabela:

SDLC	Sistema	Testes
[ALC] Avaliação de riscos do desenvolvimento	[ASE] Expectativas de segurança	[ATE] Avaliação de riscos: lista de ameaças e recomendação de controles
Direcionadores de negócio	[ASE] Categorização do sistema e requerimentos de segurança	[ATE/ AVA] Plano de testes do sistema
[ALC] Definição de padrões de desenvolvimento e codificação	[ADV] Funcionalidades do sistema e de segurança	[ATE] Resultados dos testes funcionais
Plano detalhado de avaliação	Plano de segurança do sistema	[ATE/ AVA] Resultados dos testes de segurança
[ALC] Relatórios de validação dos pontos de controle	Lista de serviços e riscos compartilhados e identificação de controles comuns	Relatório de avaliação dos controles de segurança
Autorização de execução	Especificações dos controles de segurança implementados	
	[AGD] Documentação final do sistema: guia de implantação; manual de administração; manual de uso do sistema.	

Tabela 17 – Documentação exigida para sistemas desenvolvidos.

Para sistemas adquiridos, a documentação pode se resumir a:

Sistema	Testes
Plano de segurança do sistema	[ATE] Avaliação de riscos: lista de ameaças e recomendação de controles
[ASE] Categorização do sistema e requerimentos de segurança	[ATE/ AVA] Plano de testes do sistema
[ADV] Lista de funções do sistema e de segurança	[ATE] Resultados dos testes funcionais
Especificações dos controles de segurança implementados	[ATE/ AVA] Resultados dos testes de segurança
[AGD] Documentação do sistema: guia de implantação; manual de administração; manual de uso do sistema.	Relatório de avaliação dos controles de segurança

Tabela 18 - Documentação exigida para sistemas adquiridos.

Nota: A documentação de testes de aplicações adquiridas poderá ser substituída por um termo de responsabilidade do fornecedor, afirmando que todos os requerimentos de segurança estabelecidos para o nível estipulado para a aplicação foram atendidos e, caso não atenda a um ou mais requerimentos, deverá haver justificativa formal. Contudo, a documentação do sistema deverá ser integralmente entregue.

A maioria dos documentos listados é gerada no decorrer das atividades de segurança no desenvolvimento do sistema, seguindo os requerimentos descritos no capítulo 2.

4.4.2.1. Requerimentos

Os requerimentos para avaliação dos documentos podem ser definidos pelo nível de segurança da aplicação, de forma cumulativa:

Nível	Requerimentos
1	Os controles de segurança descritos na documentação atendem explicitamente os requerimentos de segurança para o nível. São listadas as ameaças sobre o sistema. Tópicos abordados na documentação: • Visão geral do sistema • Resumo das funcionalidades do sistema • Resumo das funcionalidades de segurança • Identificação do tipo de sistema (principal ou de suporte) • Identificação dos requerimentos do sistema • Visão geral da proteção física do sistema • Descrição do escopo lógico do sistema
2	O desenvolvedor fornece informações precisas sobre as funcionalidades do sistema e de segurança, permitindo análise e facilitando os testes dos controles. Também há diagramas de funcionamento e tabelas de correlação entre os controles implantados e as ameaças avaliadas. Os controles são desenvolvidos de modo a combaterem todas as ameaças descritas anteriormente.
3 e 4	O desenvolvedor deve prover documentação da implantação dos controles. Além disso, deve ser fornecida a descrição dos controles contra violação do sistema. Testes de invasão devem ser realizados e documentados.

4.5. Testes do sistema

4.5.1. Panorama

Os testes de segurança de software validam as configurações ao longo do desenvolvimento. Principalmente, são reduzidas as falhas de segurança antes do lançamento do sistema, otimizando recursos.

Os testes verificam se o sistema:

- É estável, robusto e tem comportamento previsível;
- Expõe vulnerabilidades ou falhas;
- Trata de forma segura as falhas e exceções de funcionamento;
- Atende a todos os requerimentos funcionais de acordo com seu nível de segurança esperado;
- Viola alguma regra de segurança.

Dentre as técnicas para testes, destacam-se:

- Análise de riscos;
- Revisão de código;
- Análise estática;
- Injeção de falhas em código;
- Análise de código binário;
- Análise de vulnerabilidades;
- Teste nebuloso;
- Teste de penetração.

Alguns desses testes serão discutidos ao longo deste capítulo.

4.5.2. Planejamento dos testes

Os testes têm como base três objetos: pessoas, processos e tecnologia. São verificados, respectivamente se: desenvolvedores e usuários tem consciência da segurança; há políticas adequadas para serem adotadas; a implantação da tecnologia efetiva os processos.

Os testes verificam se o sistema:

- É estável, robusto e tem comportamento previsível;
- Expõe vulnerabilidades ou falhas;
- Trata de forma segura as falhas e exceções de funcionamento;
- Atende a todos os requerimentos funcionais de acordo com seu nível de segurança;
- Viola alguma regra de segurança.

O plano de testes deverá incluir:

- Casos de uso das funções de segurança, incluindo: uso normal, imperícia e utilização mal intencionada;
- Teste de dados (significativos e nebulosos);
- Identificação das ferramentas de testes da aplicação e seu ambiente operacional;
- Critérios de aprovação/ reprovação;
- Modelo do relatório de testes.

4.5.3. Exemplos de falhas a serem procuradas nos testes

Falha	Explicação	Formas de procura
Falsificação de identidades	Obter acesso ao sistema com uma conta fraudada ou roubada	• Tentativa de forçar usar o aplicativo sem qualquer autenticação • Verificar se há configurações que permitem autenticação anônima • Forçar o acesso da aplicação utilizando uma versão legada menos segura • Procurar dados de usuários no sistema ou autenticações permanentes • Verificar se indícios persistentes de conexão (ex. cookies) podem ser reutilizados para novas autenticações válidas • Verificar se ataques de força bruta forçam o sistema a mostrar mensagens ou dificultar o acesso à aplicação • Testar a segurança de mecanismos de recuperação de credenciais
Adulteração de dados	Tentativa de ferir a integridade de informações do sistema ou de usuários	• Testar burlar os mecanismos de autenticação • Verificar se dados adulterados podem ter seus hashes alterados posteriormente • Verificar se o aplicativo permite retornar ao uso de configurações antigas, menos seguras • Criar hashes e assinaturas fraudulentas para verificar se o sistema continua funcionando
Repúdio	Fraudar registros para que eventos não tenham responsabilização	• Verificar se há impedimentos para registros de logs ou sua auditoria • Testar a alteração de dados para que sejam gerados logs incorretos • Verificar se ações sigilosas ou administrativas podem ser efetuadas sem autenticação

Vazamento de informações	Expor informações privadas através de falhas do sistema ou erros de implementação de controles	• Avaliar arquivos e componentes do sistema para verificar a possibilidade de burlar controles de acesso obrigatórios para informações sigilosas • Encerrar os processos da aplicação para tentar acessar diretamente os dados • Utilizar sniffers para analisar o tráfego de rede em busca de dados confidenciais • Fazer ataques de injeção na tentativa de fazer com que o sistema exponha informações confidenciais ou de configuração
Negação de serviço	Buscar interromper o funcionamento do sistema, tornando-o indisponível	• Tentar inundar um processo com grande número/ volume de dados até que esse pare de responder a solicitações válidas • Verificar se é possível entrar com dados malformados • Forçar a parada do aplicativo com alterações nas variáveis externas, como espaço em disco e consumo de memória
Elevação de privilégios	Buscar, através de contas de usuários comuns, obter acesso com contas administrativas	• Verificar quantos processos são realizados com o uso de contas privilegiadas • Avaliar a possibilidade de executar dados como código • Tentar obter um shell de comando com uma conta privilegiada

4.5.4. Estresse de sistemas

Os sistemas podem ser estressados com a alteração dos dados, em si, ou quanto ao seu conteúdo, tamanho, ou contêiner de recepção da mutação dos dados conforme apresenta o diagrama (Howard & Leblanc, 2005):

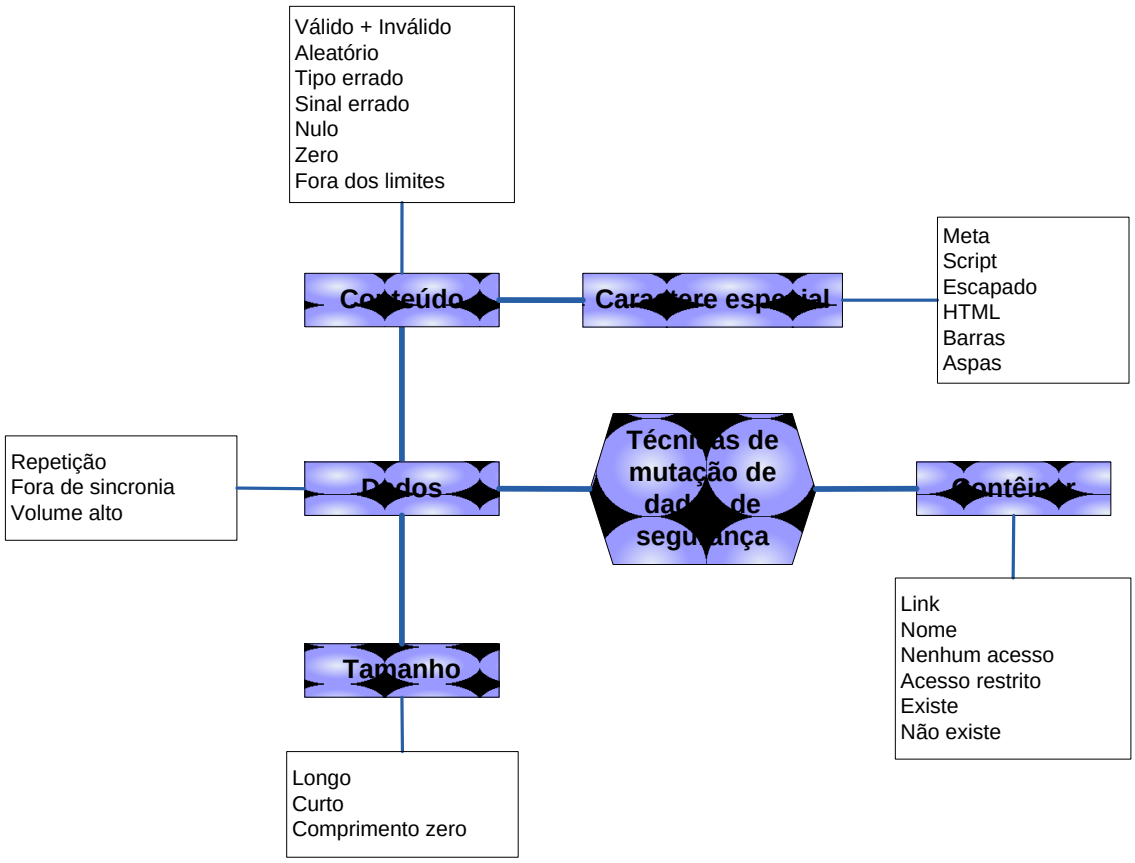


Figura 13 - Mecanismos de estresse de sistemas.

Normalmente, a alteração do contêiner acarretará na negação de serviço, pelo fato do sistema não contar com o local adequado para a utilização dos dados.

4.5.5. Testes ao longo do SDLC

Os testes podem ocorrer ao longo de todas as fases do SDLC:

Fase	Testes	Nível de Segurança			
		1	2	3	4
Concepção	Casos de uso, erro e abuso do sistema	■	■	■	■
Elaboração	Revisão do desenho	■	■	■	■
	Análise de riscos	■	■	■	■
	Análise de impacto	■	■	■	■
Construção	Revisão do código		■	■	■
	Testes nebulosos			■	■
	Análise de código binário			■	■
	Análise de vulnerabilidades			■	■
Transição (Implantação)	Análise estática	■	■	■	■
	Injeção de falhas em código fonte			■	■
	Injeção de falhas em código binário			■	■
	Testes nebulosos		■	■	■
	Análise de código binário			■	■
	Análise de vulnerabilidades	■	■	■	■
	Teste de penetração	■	■	■	■
Manutenção	Análise estática		■	■	■
	Análise de vulnerabilidades		■	■	■
	Teste de penetração		■	■	■

Tabela 19 - Testes ao longo do SDLC.

4.5.6. Avaliações

4.5.6.1. Análise de riscos

A análise de riscos deverá ser conduzida na concepção e na elaboração do sistema. É uma forma eficiente de identificar ameaças e direcionar as contramedidas apropriadas.

A modelagem de ameaças é um processo sistemático que deve ser empregado para identificar ameaças e vulnerabilidades em sistemas prontos ou em estágio de concepção. Recomenda-se a leitura do documento NIST SP 800-30 para desenvolver as atividades de avaliação de riscos, que envolvem as seguintes etapas:

- **Decompor da aplicação**: entender funcionalidades, ativos e conectividade do sistema;
- **Definir e classificar ativos**: classificar os ativos tangíveis e intangíveis de acordo com sua importância para o negócio;
- **Listar potenciais vulnerabilidades**: técnicas, operacionais ou gerenciais;
- **Definir potenciais ameaças**: montar cenários realistas dos vetores de ataque;
- **Criar estratégias de mitigação**: estabelecer controles para cada ameaça que tenha probabilidade real de ocorrer.

4.5.6.2. Revisão de código

Consiste no processo de revisar e verificar manualmente o código fonte desenvolvido, na busca por falhas e vulnerabilidades.

- Vantagens
  - Precisão;
  - Efetividade;
  - Abrangência.
- Desvantagens
  - Pouco prática para grandes volumes de códigos;

- Requer revisores altamente qualificados;
- Requer muitas horas de trabalho;
- Difícil de detectar erros de execução.

4.5.6.3. Análise estática

A análise estática consiste na avaliação do software sem executá-lo, e geralmente envolve o uso de ferramentas de análise estática. Normalmente, é realizado no código-fonte. Sua execução pode ocorrer ao longo de todo o desenvolvimento, e deve começar o mais cedo possível.

Esse tipo de análise pode ser muito útil na detecção de violações de programação, como estouros de pilhas, uso incorreto de bibliotecas e erros de digitação.

- Vantagens
  - Mais rápido que a revisão de código;
  - Cobertura completa e consistente;
  - Detecção de falhas grosseiras e relativas à linguagem de programação;
  - Eficiente para códigos grandes;
  - Os revisores não precisam ser experts.
- Desvantagens
  - Grande número de falsos positivos;
  - Detecção de falsos negativos;
  - Incapacidade em detectar erros sutis;
  - Pode requerer muitas horas de trabalho para a revisão dos resultados;
  - Não detecta erros de execução.

4.5.6.4. Injeção de falhas em código fonte

Utilizada para estressar o sistema, gerar problemas de interoperabilidade, simular falhas em ambientes operacionais e revelar falhas não aparentes. Os testes são conduzidos pela execução de operações anormais (imperícia ou uso mal intencionado) da aplicação ou em seu ambiente.

Há duas técnicas fundamentais de injeção de falhas em código, com o objetivo de avaliar os impactos resultantes da propagação de falhas no código fonte e os sistemas de auto-correção da aplicação:

- **Análise de propagação estendida**: o avaliador deve gerar uma árvore de falhas a partir do código fonte, e inserir falhas de forma programada. Essas falhas deverão ser executadas e avaliadas para verificar como os erros são propagados pelo sistema.
- **Análise de propagação de interface**: nessa modalidade, os códigos dos módulos são alterados para avaliar como os erros se propagam através das interfaces entre componentes da aplicação e com o ambiente.

Esses testes são úteis na detecção de ponteiros e vetores incorretos, chamadas perigosas e condições de competição. Como os outros testes, deve ser utilizado de forma iterativa ao longo do desenvolvimento do sistema.

4.5.6.5. Injeção de falha em código binário

Os testes de injeção de falhas em código binário têm como objetivo a detecção de erros de execução da aplicação. Por exemplo, são verificados se é possível detectar os nomes, parâmetros e o retorno de cada chamada de sistema.

A realização desses testes requer a injeção de falhas no ambiente operacional, o que traz semelhança com as situações reais de operação e ataque. Para melhores resultados, a injeção de falha em binários deve ser realizada em conjunto com testes de penetração.

Dentre os benefícios desse tipo de teste, destacam-se:

- A simulação de anormalidades do ambiente operacional mesmo sem ter o entendimento de sua ocorrência no mundo real;
- Os avaliadores podem decidir quais falhas ambientais devem ser testadas, em ambientes controlados, isolando os resultados e podendo avaliar as interações entre os problemas externos do sistema;
- A facilidade de automatização dos testes.

4.5.6.6. Teste nebuloso

Os testes nebulosos consistem na entrada aleatória de dados (geralmente gerados pela modificação em entradas válidas) no sistema, por sua interface ou por seus componentes. São utilizados em testes específicos, como de entrada de dados em HTTP.

As falhas encontradas por esses testes são utilizadas como base para formas mais específicas de testes, como a revisão de código.

4.5.6.7. Análise de código binário

A análise de código binário utiliza ferramentas de engenharia reversa como:

- ***Scanners* de executáveis**: técnica menos intrusiva, que analisa comportamento, fluxos de controle e dados, e chamadas externas do sistema. Normalmente, ferramentas de análise de vulnerabilidades realizam testes em binários;
- ***Disassemblers***: técnica de engenharia reversa que remonta os binários em um estágio intermediário de compilação. É medianamente intrusivo, mas requer analistas altamente qualificados para a avaliação dos resultados;
- **Descompiladores**: técnica extremamente intrusiva de engenharia reversa que tem como objetivo a obtenção do código fonte que originou o binário, para que possa ser alterado e recompilado. Contudo, os códigos obtidos são raramente navegáveis e a qualidade do código é normalmente baixa.

Precauções devem ser tomadas com esse tipo de análise, porque a engenharia reversa pode vir a ser considerada violação aos direitos autorais do desenvolvedor do sistema.

4.5.6.8. Análise de vulnerabilidades

Sistemas web, bancos de dados e alguns sistemas operacionais podem ter sua segurança avaliada com ferramentas de varreduras de vulnerabilidades. Essas análises consistem em verificar padrões (assinaturas) de vulnerabilidades. A eficácia de um scanner de vulnerabilidades pode ser medida pelo tamanho da base de dados de assinaturas e pela frequência de atualização dessas.

Além de avaliar aplicações, os *scanners* podem verificar falhas em redes, servidores, sistemas operacionais e configurações, favorecendo a segurança do ambiente de software.

Embora as análises sejam automatizadas, faz-se necessária análise posterior para definir o nível de risco das falhas encontradas, com base no negócio. Como vantagem, a maioria dos aplicativos de análise de vulnerabilidades conta com relatórios de correções a serem aplicadas por atualizações de sistema ou ajuste de configurações.

A análise de vulnerabilidades tem maior efetividade quando usada:

- Durante a análise de componentes binários;
- Antes do teste de penetração, para resolver as falhas de segurança mais simples.

4.5.6.9. Teste de penetração

Os testes de penetração têm como foco a exploração de brechas de segurança em sistemas para o ganho de controle ou alteração no funcionamento de aplicações prontas, em execução.

Os testes podem ser realizados como se o avaliador estivesse na posição do usuário, mas os melhores resultados são obtidos quando a aplicação é testada diretamente. O ambiente operacional também deve ser testado.

O plano de teste de invasão deve imaginar o “pior caso”, com ataques que trariam impacto à confidencialidade, integridade e disponibilidade. O plano deve abordar:

- A política e/ou os requerimentos de segurança a serem reforçados;
- A antecipação de ameaças ao sistema;
- As sequências mais prováveis para os ataques ao sistema.

Cabe ressaltar que os testes de invasão não deverão ser considerados como a principal ou única forma de teste, principalmente por ser realizado em momentos finais do desenvolvimento.

4.6. Interpretação e comunicação dos resultados

Assim que cada teste é concluído, os resultados e a versão exata testada devem ser marcados no sistema de gerenciamento de configurações.

Os relatórios devem conter:

- Avaliação da documentação
    - Análise de consistência das atividades realizadas no desenvolvimento
    - As falhas encontradas em procedimentos do manual do sistema, bem como suas inconsistências
    - O grau de alinhamento entre o sistema testado e a documentação
    - O grau de alinhamento entre o sistema e os requerimentos de segurança
- Testes de segurança
    - A usabilidade e a precisão das informações dos documentos de usuário e de operação do sistema
    - Requerimentos de segurança faltantes e a necessidade de correções
- Considerações
    - Orientação de alterações necessárias para a documentação
    - Determinação do volume e da natureza de reengenharia para atender aos novos requerimentos

4.7. Aprovação do sistema

Para que um sistema seja aprovado, o mesmo deverá atender a todos os requerimentos de segurança e de documentação definidos neste manual, de acordo com o seu nível de segurança.

Caso um sistema não atenda a todos os requerimentos do seu nível, duas situações podem ocorrer:

1) O sistema atenderá a um nível inferior, com características do nível superior;
2) O sistema atenderá ao nível de objetivo, mas terá toda a documentação das ressalvas para não atendimento de todos os requerimentos.

Caberá ao avaliador definir se o sistema atende ou não aos objetivos de segurança definidos durante o desenvolvimento. Por fim, o sistema terá que ser autorizado pelo gerente de autorização para entrar em produção.

5. Referências e Literatura Complementar

Adaikkappan, A. (2009). Application Security Controls: An Audit Perspective. *ISACA Journal* .

Agência Brasil. (22 de 08 de 2009). *Folha Online.* Acesso em 04 de 2010, disponível em Folha: http://www1.folha.uol.com.br/folha/brasil/ult96u613407.shtml

Albuquerque, R. (2002). *Segurança no Desenvolvimento de Software.* São Paulo: Campus.

Basham, R. (2006). Procedure Guidelines and Controls Documentation: SDLC Controls in Cobit 4.0. *ISACA* .

Croll, P., & Moss, M. (2008). Leveraging Models and Standards for Assurance. *SSTC.* Las Vegas.

CWE - Common Weakness Enumeration. (2010). 2010 CWE/SANS Top 25 Most Dangerous Software. *http://cwe.mitre.org/top25/* .

FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION. (2006). Minimum Security Requirements for Federal Information and Information Systems. *FIPS PUB 200* .

Governo Brasileiro: Comitê Executivo de Governo Eletrônico. (2009). e-PING: Padrões de Interoperabilidade de Governo Eletrônico. *http://www.eping.e.gov.br* .

Graff, M. G., & Wyk, K. R. (2003). *Secure Coding: Principles and Practices.* Sebastopol, CA, Estados Unidos: O'Reilly.

Greene, F. (2002). A Survey of Application Security in Current International Standards. *http://www.isaca.org* .

Howard, M., & Leblanc, D. (2005). *Escrevendo Código Seguro: estratégias e técnicas práticas para codificação segura de aplicativos em um mundo em rede.* (2a. ed.). Porto Alegre: Bookman.

Huey, P. (2010). Oracle® Database Security Guide. *Oracle Corporation* .

IBM. (2008, 01). Understanding Web application security challenges. *Web application security management* . Estados Unidos: IBM.

IEEE Computer Society. (2006). IEEE Standard for Developing a Software Project Life Cycle Process. *http://standards.ieee.org* .

IG. (22 de 07 de 2009). *Último Segundo.* Acesso em 04 de 2010, disponível em IG: http://ultimosegundo.ig.com.br/brasil/2008/07/22/policiais_civis_fazem_mega_operacao_no_rio_de_janeiro_1460079.html

Information Systems Audit and Control Association. (2003). Control Risk Self-Assessment. *www.isaca.org* .

Information Systems Audit and Control Association. (2001). Generic Application Review. *www.isaca.org* .

International Organization for Standardization. (2008). *Information technology - Security techniques - Evaluation criteria for IT security - Part 2: Security functional components.* Gênova: ISO/IEC.

ISACA. (2007). Systems Development Life Cycle and IT Audits. *www.isaca.org* .

IT Governance Institute. (2008). Aligning CobiT® 4.1, ITIL® V3 and ISO/IEC 27002 for Business Benefit. *www.itgi.org* .

IT Governance Institute. (2007). CobiT 4.1. *www.itgi.org* .

IT Governance Institute. (2007). CobiT Mapping: Mapping of NIST SP800-53 Rev 1 With COBIT® 4.1. *www.itgi.org* .

IT Governance Institute. (2006). CobiT Mapping: Overviewof International IT Guidance, 2nd Edition. *www.itgi.org* .

Kissel, R. (2008). Security Considerations in the System Development Life Cycle. *NIST Special Publication 800-64 Revision 2* .

Kurth, H. (2009). Security Target for Oracle Database 11g Release 1 (11.1.0) with Oracle Database Vault. *Oracle Corporation* .

Mead, N. R., Hough, E. D., & Stehney II, T. R. (2005). Security Quality Requirements Engineering (SQUARE) Methodology. *Carnegie Mellon University* .

Mellado, D., Fernández-Medina, E., & Piattini, M. (2007). A common criteria based security requirements engineering process for the development of secure information systems. *Computer Standards & Interfaces , 29*, pp. 244-253.

Microsoft Corporation. (2003). Design Guidelines for Secure Web Applications.

Microsoft Corporation. (2010). Microsoft Security Development Lifecycle. *http://www.microsoft.com/sdl* .

Microsoft Corporation. (2005). Microsoft SQL Server 2005: Security-Enhanced Database Platform. *http://www.microsoft.com/sql* .

MySQL Security Guide extract from the MySQL 5.1 Reference Manual. (2010).

National Infrastructure Security Co-ordination Centre. (2006). Secure web applications: Development, installation and security testing. *www.niscc.gov.uk* .

NIST. (1995). An Introduction to Computer Security. *Special Publication 800-12* .

NIST. (02 de 2004). FIPS 199: Standards for Security Categorization of Federal Information and Information Systems. *FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION* , p. 13.

NIST. (2010). Guide for Applying the Risk Management Framework to Federal Information Systems. *NIST Special Publication 800-37* .

NIST. (2008). Guide for Assessing the Security Controls in Federal Information Systems. *NIST Special Publication 800-53A* .

NIST. (2007). Guide to Secure Web Services. *Special Publication 800-95* .

NIST. (2003). Guideline on Network Security Testing. *NIST Special Publication 800-42* .

NIST. (2009). Recommended Security Controls for Federal Information Systems and Organizations. *NIST Special Publication 800-53 Revision 3* .

NIST. (2008). Security Considerations in the System Development Life Cycle. *NIST Special Publication 800-64 Revision 2* .

NIST. (2002, Junho). SP 800-34: Contingency Planning Guide for Information Technology Systems. *Special Publication Series 800* , p. 107.

NIST. (10 de 2008). SP 800-64: Security Considerations in the System Development Life Cycle. *NIST Special Publications 800 Series* , p. 67.

NIST. (2008). Technical Guide to Information Security Testing and Assessment. *Special Publication 800-115* .

OWASP Foundation. (2008). OWASP TESTING GUIDE. *http://www.owasp.org* .

OWASP Foundation. (s.d.). Software Assurance Maturity Model: A guide to building security into software development. *http://www.opensamm.org* .

Paul, M. (s.d.). A Kaleidoscope of Perspectives. *(ISC)2* .

Paul, M. (s.d.). Software Security: Being Secure in an Insecure World. *(ISC)2* .

Paul, M. (n.d.). The Need for Secure Software. (ISC)2.

Paul, M. (s.d.). The Ten Best Practices for Secure Software Development. *(ISC)2* .

PC Guardian. (2003). Encryption Plus® Hard Disk 7.0 Security Target. *http://www.pcguardian.com* .

Pessoa, M. (2007). *Segurança em PHP: desenvolva programas PHP com alto nível de segurança e aprenda como manter os servidores web livres de ameaças.* São Paulo: Novatec Editora.

Peter, W. (2008). CC and CMMI: An Approach to Integrate CC with Development. *TÜV Informationstechnik GmbH* .

RAGEN, A. (2007). Manager's Guide to the Common Criteria.

Richardson, R. (2008). CSI Computer Crime & Security Survey. *Computer Security Institute* .

UK Law Enforcement Community. (2009/2010). The United Kingdom Threat Assessment of Organised Crime. *SOCA* .

UOL. (22 de 10 de 2009). *UOL Educação.* Acesso em 04 de 2010, disponível em UOL: http://educacao.uol.com.br/ultnot/2009/10/22/ult1811u454.jhtm

Vaz, L. (21 de 12 de 2009). *Clipping.* Acesso em 04 de 2010, disponível em Ministério do Planejamento: https://conteudoclippingmp.planejamento.gov.br/cadastros/noticias/2009/12/21/fraude-na-previdencia-chega-a-r-1-6-bilhao

Woody, C. (2008). Strengthening Ties Between Process and Security. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.

6. Anexos

6.1. Glossário

Acreditação: Decisão oficial de gerenciamento para autorizar a operação de um sistema de informação e aceitar explicitamente os riscos residuais.

Ambiente: Agregação de procedimentos, condições e objetos externos afetando o desenvolvimento, operação e manutenção de um sistema de informação.

Análise de impacto na privacidade: Uma análise de como a informação é manuseada: 1) para garantir seu manuseio em conformidade com requerimentos políticos, regulatórios e legais; 2) para determinar os riscos e efeitos da coleta, manutenção e distribuição da informação; 3) para examinar e avaliar proteções e processos alternativos de manuseio de informações para mitigar potenciais riscos sobre a privacidade.

Análise de Impacto no Negócio: Análise de requerimentos, processos e interdependências de um sistema de informação, usada para caracterizar prioridades e requerimentos de contingência do sistema para o caso de uma interrupção significante.

Análise de Riscos: Processo de identificação de riscos à segurança do sistema e determinação da probabilidade de ocorrência, do impacto, e de medidas adicionais para mitigar esse impacto.

Ataque de Buffer Overflow: Método para sobrecarregar uma quantidade pré-definida de espaço em um *buffer*, o que pode sobrescrever ou corromper dados na memória.

Ataque de força bruta: Método utilizado para acessar um dispositivo através da tentativa de combinações múltiplas de senhas numéricas e alfanuméricas.

Ativo: Recurso, aplicação principal, sistema de suporte geral, programa de alto impacto, planta física, sistema de missão crítica, ou um grupo de sistemas relacionados logicamente.

Auditoria: Revisão e exame, independente, de registros e atividades, para avaliar a adequação de controles de sistemas, para assegurar a conformidade com políticas e procedimentos operacionais estabelecidos, e recomendar mudanças necessárias em controles, políticas e procedimentos.

Autenticação: Verificação da identidade de um usuário, processo ou dispositivo, geralmente como pré-requisito para permissão de acesso a recursos de um sistema de informação.

Autenticar: Confirmar a identidade de uma entidade quando esta identidade é apresentada.

Autoridade Certificadora: Uma entidade confiável, que emite ou revoga certificados de chave pública.

Autorização: Decisão oficial de gerenciamento para autorizar a operação de um sistema de informação e aceitar explicitamente os riscos residuais.

Backup: Cópia de arquivos e programas feita para facilitar a recuperação dos mesmos, caso necessário.

Certificado digital: Representação digital de uma informação que, no mínimo: 1) identifica a autoridade certificadora que a emitiu; 2) nomeia ou identifica o assinante; 3) contém uma chave pública do assinante; 4) identifica seu período operacional; e 5) é assinada digitalmente pela autoridade certificadora que a emitiu.

Chaves assimétricas: Duas chaves relacionadas, uma pública e outra privada, que são usadas para executar operações complementares, tais como cifragem e decifragem, ou geração e verificação de assinatura.

Cliente (Aplicação): Uma entidade do sistema, geralmente um processo de computador, agindo em nome de um usuário humano, que faz uso de um serviço prestado por um servidor.

Código malicioso: Código intencionado a executar um processo não autorizado, que pode causar um impacto adverso na confidencialidade, integridade ou disponibilidade de um sistema de informação.

Confidencialidade: Restrições autorizadas para preservação do acesso e descarte de informações, incluindo meios de proteger informações pessoais privadas e proprietárias.

Controles compensatórios: Controles técnicos, operacionais e gerenciais empregados por uma organização em lugar dos controles recomendados, que oferecem proteção equivalente ou comparável para um sistema de informação.

Controles gerenciais: Controles de segurança para um sistema de informação que focam no gerenciamento do risco e da segurança do sistema de informação.

Controles operacionais: Controles de segurança do sistema de informação que primeiramente são implementados e depois são executados por pessoas.

Cookie: Um pedaço de uma informação fornecida por um servidor web a um browser, juntamente com o recurso requisitado, para que browser armazene temporariamente e retorne ao servidor em qualquer visita ou requisição subseqüente.

Criptografia: a disciplina que incorpora os princípios, meios e métodos para a transformação de dados com a finalidade de ocultar seu conteúdo semântico e prevenir sua utilização não autorizada ou sua modificação não detectada.

Criptografia: Série de transformações que transforma um texto plano em um texto cifrado.

Decifragem: O processo de transformar texto cifrado em texto plano.

Disponibilidade: Garantia de tempestividade e confiabilidade no acesso e uso da informação.

Firewall: Um portão que limita o acesso entre redes de acordo com a política de segurança local.

Hashing: O processo de uso de um algoritmo matemático contra um dado para produzir um valor numérico que represente aquele dado.

IDS (Intrusion Detection System): Sistema de Detecção de Intrusos. Software que procura por atividades suspeitas e alerta administradores.

IPS (Intrusion Prevention System): Sistema de Prevenção de Intrusos. Software que toma ações perante alertas gerados pelo IDS. Quando integrado ao IDS, pode ser chamado de IDPS (Intrusion Detection and Prevention System).

Impacto: Magnitude do dano que pode ser causado por conseqüência de operações não autorizadas em informações, tais como descarte, modificação, destruição, perda, entre outras.

Integridade do dado: A propriedade que um dado tem de não ter sido alterado de maneira não autorizada durante seu armazenamento, seu processamento e sua transmissão.

Interrupção: Evento não planejado que torna o sistema inoperante por um espaço de tempo inaceitável.

Menor Privilégio: Garantia aos usuários apenas dos acessos aos quais estes necessitam para realizar suas funções oficiais.

Não repúdio: Garantia de que o remetente da informação recebe a prova da entrega e o destinatário recebe a prova da identidade do remetente, não podendo posteriormente negar ter processado a informação.

Objeto: Entidade passiva que contém ou recebe informação.

PDSOO: Processo de Desenvolvimento de Software Orientado a Objeto. Ciclo de vida de desenvolvimento de sistemas da Prodemge.

Plano de contingência: Política e procedimentos de gerenciamento, desenhados para manter ou recuperar operações de negócio, incluindo operações de computadores, no caso de eventuais emergências, desastres ou falhas de sistemas.

Plano de Continuidade de Negócios (PCN): Documentação de uma série de instruções ou procedimentos pré-determidados, que descrevem como as funções de negócios da organização serão sustentadas durante e após uma interrupção significante.

Política de Segurança: Um documento que descreve a estrutura de gerenciamento de segurança e claramente designa responsabilidades de segurança e estabelece as bases necessárias para medir o cumprimento e o progresso.

Proxy: Aplicação que quebra a conexão entre cliente e servidor. O Proxy aceita certos tipos de tráfego entrando ou saindo de uma rede, processa e os encaminha, fechando o caminho direto entre redes internas e redes externas.

Risco residual: O potencial risco remanescente depois que todas as medidas de segurança são aplicadas. Para cada ameaça, há um risco residual relacionado.

Risco: Nível de impacto em operações ou ativos, resultantes da operação de um sistema de informação, dado o potencial risco de uma ameaça e a probabilidade da ameaça vir a ocorrer.

SDLC: Systems Development Lifecycle. Ciclo de vida do desenvolvimento de sistemas. Conjunto de metodologias e práticas para o desenvolvimento de aplicações e hardware. Pode ser adaptado para a aquisição de sistemas.

Segurança da Informação: A proteção da informação e de sistemas de informação contra acesso, uso, descarte, modificação ou destruição não autorizada, com a finalidade de fornecer confidencialidade, integridade e disponibilidade.

Senha: Um segredo que alguém memoriza e usa para se autenticar ou autenticar sua identidade.

Sistema de Informação: Um conjunto discreto de recursos de informação, organizados de forma a coletar, processar, manter, usar, compartilhar, disseminar e disponibilizar informação.

Software antivírus: Programa que monitora computadores ou redes para identificar aplicações maliciosas e prevenir incidentes.

Texto cifrado: Texto em sua forma criptografada.

Trilha de auditoria: Um registro mostrando quem acessou um sistema de informação, e quais operações o usuário executou em um determinado período.

Vulnerabilidade: Fragilidade em um sistema de informação, procedimentos de segurança do sistema, controles internos ou implementação, que poderia ser explorada por uma ameaça.

6.2. Mapeamento de normas e padrões ao CobiT 4.1

Cobit x CMMI

		Cobit AI2 - Adquirir e Manter Software Aplicativo										Cobit AI6 - Gerenciar Mudanças		
		AI2.1. Design de Alto Nível	AI2.2. Design Detalhado	AI2.3. Controle de Aplicação e Auditabilidade	AI2.4. Segurança de Aplicação e Disponibilidade	AI2.5. Configuração e Implementação de Software Aplicativo Adquirido	AI2.6. Mudança em Sistemas Existentes	AI2.7. Desenvolvimento de Software Aplicativo	AI2.8. Controle de Qualidade de Software	AI2.9. Gerenciamento de Requisitos para Aplicações	AI2.10. Manutenção de Software Aplicativo	AI6.1. Procedimentos e padrões de mudanças	AI6.2 Avaliação de impacto, prioridades e autorização	AI6.3 Mudanças emergenciais
CMMI para Desenvolvimento v1.2 - Maturidade nível 2	Requisitos de Negócio	■	■						■					
	Gerenciamento de configuração (CM)	■	■			■		■		■	■	■	■	■
	Controle e monitoramento de projeto (PMC)							■			■			
	Planejamento de projeto (PP)										■			
	Controle de qualidade de processo e produto (PPQA)		■					■	■	■				
	Gerenciamento de requisição (REQM)	■	■					■		■	■			
	Gerenciamento de acordo com fornecedor (SAM)					■					■			

Cobit x ISO 15408

		Cobit AI2 - Adquirir e Manter Software Aplicativo										Cobit AI6 - Gerenciar Mudanças		
		AI2.1. Design de Alto Nível	AI2.2. Design Detalhado	AI2.3. Controle de Aplicação e Auditabilidade	AI2.4. Segurança de Aplicação e Disponibilidade	AI2.5. Configuração e Implementação de Software Aplicativo Adquirido	AI2.6. Mudança em Sistemas Existentes	AI2.7. Desenvolvimento de Software Aplicativo	AI2.8. Controle de Qualidade de Software	AI2.9. Gerenciamento de Requisitos para Aplicações	AI2.10. Manutenção de Software Aplicativo	AI6.1. Procedimentos e padrões de mudanças	AI6.2 Avaliação de impacto, prioridades e autorização	AI6.3 Mudanças emergenciais
ISO 15408	2. Requerimentos funcionais de segurança	■	■	■	■	■	■	■		■		■		
	3. Requerimentos de validação de sistemas			■			■		■		■		■	

Cobit x ISO 27002

		Cobit AI2 - Adquirir e Manter Software Aplicativo										Cobit AI6 - Gerenciar Mudanças		
		AI2.1. Design de Alto Nível	AI2.2. Design Detalhado	AI2.3. Controle de Aplicação e Auditabilidade	AI2.4. Segurança de Aplicação e Disponibilidade	AI2.5. Configuração e Implementação de Software Aplicativo Adquirido	AI2.6. Mudança em Sistemas Existentes	AI2.7. Desenvolvimento de Software Aplicativo	AI2.8. Controle de Qualidade de Software	AI2.9. Gerenciamento de Requisitos para Aplicações	AI2.10. Manutenção de Software Aplicativo	AI6.1. Procedimentos e padrões de mudanças	AI6.2 Avaliação de impacto, prioridades e autorização	AI6.3 Mudanças emergenciais
ISO 27002:2005	5. Política de Segurança	▣	▣	▣	▣	▣	▣	▣	▣	▣	▣			
	6.1.4. Processo de autorização para os recursos de processamento de informação				▣									
	7.2.1. Recomendações para classificação				▣									
	10.1.2 Gerenciamento de mudanças											▣	▣	▣
	10.3.2. Aceitação de sistemas				▣				▣					
	10.10.1. Registros de auditoria			▣										
	10.10.5. Registros (logs) de falhas			▣										
	11.5.4 Uso de utilitários de sistemas													▣
	11.6.2. Isolamento de sistemas sensíveis				▣									
	12.1.1. Análise e especificação dos requisitos de segurança				▣									
	12.2.1. Validação dos dados de entrada			▣										
	12.2.2. Controle de processamento interno			▣										
	12.2.3. Integridade de mensagens			▣	▣									
	12.2.4. Validação de dados de saída			▣										
	12.3.1. Política para o uso de controles criptográficos				▣									
	12.4.3. Controle de acesso ao código-fonte de programa				▣									
	12.5.1. Procedimentos para controle de mudanças						▣						▣	▣
	12.5.2. Análise crítica técnica das aplicações após mudanças no sistema operacional				▣									
	12.5.3. Restrições sobre mudanças em pacotes de software					▣						▣	▣	▣
	12.5.4. Vazamento de informações				▣									
	12.5.5. Desenvolvimento terceirizado de software							▣						
	12.6.1 Controle de vulnerabilidades técnicas												▣	▣
	13.2.3. Coleta de evidências			▣										
	15.1.1. Conformidade com requisitos legais	▣	▣						▣					
	15.1.2. Direitos de propriedade intelectual	▣	▣						▣					
	15.1.3. Proteção de registros organizacionais	▣	▣						▣					
	15.1.4. Proteção de dados e privacidade de informações pessoais	▣	▣						▣					
	15.3.1. Controles de auditoria de sistemas de informação			▣										
	15.3.2. Proteção de ferramentas de auditoria de sistemas de informação			▣	▣									

Cobit x NIST SP 800-53

		Cobit AI2 - Adquirir e Manter Software Aplicativo										Cobit AI6 - Gerenciar Mudanças		
		AI2.1. Design de Alto Nível	AI2.2. Design Detalhado	AI2.3. Controle de Aplicação e Auditabilidade	AI2.4. Segurança de Aplicação e Disponibilidade	AI2.5. Configuração e Implementação de Software Aplicativo Adquirido	AI2.6. Mudança em Sistemas Existentes	AI2.7. Desenvolvimento de Software Aplicativo	AI2.8. Controle de Qualidade de Software	AI2.9. Gerenciamento de Requisitos para Aplicações	AI2.10. Manutenção de Software Aplicativo	AI6.1. Procedimentos e padrões de mudanças	AI6.2 Avaliação de impacto, prioridades e autorização	AI6.3 Mudanças emergenciais
NIST SP 800-53	AC-3. Reforço do controle de acesso				■									
	AU-2. Eventos auditáveis			■										
	CM-1. Procedimentos e políticas da gestão de configurações											■		
	CM-3. Controle de mudanças de configurações											■		■
	IA-2. Identificação e autenticação (de usuários)				■									
	MA-2. Manutenção controlada										■			
	SA-1. Políticas e procedimentos de aquisição de sistemas e serviços					■								
	SA-3. Suporte ao ciclo de vida							■						
	SA-4. Aquisições				■									
	SA-8. Princípios de Engenharia Segura				■									
	SA-11. Testes de segurança do desenvolvedor								■					
	SC-2. Particionamento de Aplicações				■									
	SI-7. Integridade de software e informações				■									
	SI-10. Validação de entrada de dados			■	■									

6.3. Modelo de ameaças e vulnerabilidades

O modelo de ameaças deve apresentar atributos mínimos para caracterizar cada ameaça identificada. A seguir, são listados atributos essenciais para a composição do modelo:

- **Identificador:** Número único usado para referenciar a ameaça;
- **Ameaça:** Nome da ameaça;
- **Descrição:** Breve descrição da ameaça, incluindo possíveis técnicas de ataque;
- **Potenciais alvos:** Ativos, recursos ou componentes do sistema que podem ser alvos da ameaça;
- **Motivação:** Possíveis motivos pelos quais a ameaça pode ser utilizada contra o sistema (ganhos econômicos, ganhos políticos, curiosidade, ego, erros não intencionais);

A tabela a seguir pode ser utilizada para documentar as ameaças identificadas:

ID	Ameaça	Descrição	Potenciais alvos	Motivação

O próximo passo consiste na identificação das vulnerabilidades (falhas ou fraquezas) que poderiam ser exploradas pelas ameaças. Para a documentação das vulnerabilidades, podem ser destacados os seguintes atributos:

- **Identificador:** Número único usado para referenciar a vulnerabilidade;
- **Vulnerabilidade:** Breve descrição da vulnerabilidade, e como a mesma pode ser explorada;
- **Ameaças:** Referência às ameaças que podem explorar a vulnerabilidade;
- **Ativo:** Ativo, recurso ou componente do sistema que apresenta a vulnerabilidade;

A tabela a seguir pode ser utilizada para documentar as vulnerabilidades:

ID	Vulnerabilidade	Ameaças	Ativo

6.4. Análise de riscos

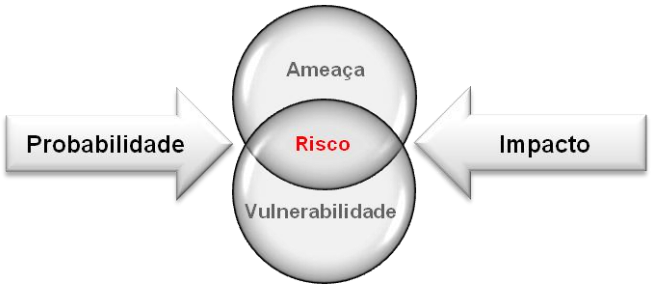
O risco é composto pelo par ameaça / vulnerabilidade. Quando há conseqüência, o risco é significativo e deve ser avaliado.

Os elementos de avaliação de riscos são:

- Ativo (informações, sistemas, dispositivos, pessoas, etc.)
  - Contém vulnerabilidades
- Ameaça (atacante, usuário insatisfeito, criminoso, etc.)
  - Explora as vulnerabilidades
- Impacto (danos de imagem, financeiros, físicos, legais, etc.)
  - Causa exposição após a exploração bem sucedida

O cálculo básico do risco pode ser obtido em:

risco = probabilidade x impacto



Para a análise da probabilidade do risco se concretizar, devem ser considerados os controles compensatórios relativos à vulnerabilidade do sistema (Quanto mais efetivos forem os controles, menor a probabilidade).

Portanto, a probabilidade de um risco se concretizar pode ser classificada em 3 (três) níveis:

Probabilidade	Descrição
Alta	Os controles compensatórios são insuficientes para a mitigação do risco
Média	Os controles compensatórios reduzem a probabilidade de exploração da vulnerabilidade
Baixa	Os controles compensatórios são efetivos, e reduzem consideravelmente a probabilidade de exploração da vulnerabilidade

A análise de impacto se refere à magnitude do potencial dano que pode ser causado pela exploração bem sucedida de uma vulnerabilidade. O impacto é determinado pelo valor do ativo em risco, levando-se em consideração o custo para sua substituição, sua importância para o negócio e a sensibilidade do dado nele contido.

O nível de segurança, definido durante a atividade de categorização do sistema, também deve ser considerado no momento de determinar o impacto de cada risco identificado.

O impacto do risco também pode ser classificado em alto, médio e baixo:

Impacto	Descrição
Alto	A materialização do risco pode causar danos severos ou catastróficos às operações e aos ativos individuais ou organizacionais
Médio	A materialização do risco pode causar danos significativos às operações e aos ativos individuais ou organizacionais
Baixo	A materialização do risco pode causar danos limitados às operações e aos ativos individuais ou organizacionais

Com a determinação da probabilidade e do impacto do risco, é possível definir o nível do risco. A seguinte matriz de nível de risco pode ser utilizada para auxiliar nesta tarefa:

		Impacto		
		Alto	**Médio**	**Baixo**
Probabilidade	**Alta**	Alto	Alto	Médio
	Média	Alto	Médio	Baixo
	Baixa	Médio	Baixo	Baixo

Após a definição do nível do risco, devem ser determinadas as recomendações para mitigação do risco. As recomendações devem levar em consideração aspectos como nível de esforço, custos, tecnologias emergentes, restrições de tempo, e viabilidade.

Para mitigar ou eliminar riscos, podem ser implementados controles que:

- Bloqueiam ameaças (firewalls e muros, por exemplo);
- Eliminam vulnerabilidades (atualizações de sistema, correção de falhas, etc.);
- Reduzem impacto;
- Transferem o risco (como seguros).

A documentação dos resultados da análise de riscos deve fornecer:

- **Identificador:** Número único usado para referenciar cada vulnerabilidade;
- **Origem:** Referência à origem (documento ou atividade de avaliação de riscos) onde a vulnerabilidade foi identificada;
- **Risco:** Breve descrição do risco;
- **Possíveis Impactos ao negócio:** Possíveis impactos que o negócio pode sofrer em caso de uma exploração bem sucedida da vulnerabilidade;
- **Recomendações:** Breve descrição das ações recomendadas para a mitigação do risco;
- **Probabilidade:** Probabilidade de uma ameaça explorar uma vulnerabilidade;
- **Impacto:** Nível de impacto do risco;
- **Nível do risco:** Nível definido para o risco (alto, médio, baixo).

A tabela a seguir pode ser utilizada para documentar os resultados da análise de riscos:

ID	Origem	Risco	Possíveis Impactos ao negócio	Recomendações	Probabilidade	Impacto	Nível de Risco

A avaliação de riscos tem objetivos específicos em cada fase do ciclo de vida do desenvolvimento de sistemas, conforme mostra a tabela:

Fases do SDLC	Características	Suporte de Atividades de Gerenciamento de Risco
Fase 1 – Concepção	É manifestada a necessidade de um sistema de TI e a proposta e o escopo são documentados.	Riscos identificados são utilizados para embasar o desenvolvimento dos requerimentos de sistema, incluindo requerimentos de segurança e uma concepção de segurança das operações (estratégias).
Fase 2 – Elaboração	O sistema de TI é comprado ou desenvolvido.	Os riscos identificados durante essa fase podem ser utilizados para embasar as análises de segurança do sistema de TI, que poderão gerar mudanças no desenvolvimento do projeto.
Fase 3 - Transição (Implantação)	As características de segurança do sistema deverão estar configuradas, habilitadas, testadas e verificadas.	O processo de gerenciamento de risco embasa o acompanhamento da implantação do sistema diante seus requerimentos dentro de seu ambiente operacional modelado.
Fase 4 - Operação e Manutenção	O sistema está em funcionamento. Geralmente estão em constante mudança devido a alterações de hardware, software, políticas e processos.	As atividades de gerenciamento de risco são realizadas por um sistema periódico de validação ou quando grande mudanças são realizadas no ambiente de produção.
Fase 5 – Desativação	Essa fase pode envolver a disposição de informação, hardware e software. Atividades podem incluir transferência, arquivamento, descarte ou destruição da informação para eliminar evidências do hardware e do software.	As atividades de gerenciamento de risco são realizadas por componentes de sistema que irão ser descartados ou repostos para garantir que o descarte de evidências ocorra de maneira segura.

6.5. Ferramentas para testes em sistemas

Há uma série de ferramentas disponíveis para a realização de testes em sistemas. Foram listadas algumas ferramentas gratuitas ou com versão demo, mas é altamente recomendável que seja realizada uma pesquisa prévia para determinar a ferramenta mais adequada para cada teste.

Ferramenta	**MSAT (Microsoft Security Assessment Tool)**
Descrição	Ferramenta gratuita para avaliar pontos fracos no ambiente de segurança de TI atual, revelar listas de problemas com prioridade, além de fornecer diretrizes específicas para minimizar esses riscos.
Onde obter	http://www.microsoft.com/downloads/details.aspx?FamilyId=CD057D9D-86B9-4E35-9733-7ACB0B2A3CA1&displaylang=em

Ferramenta	**SDL Threat Modeling Tool**
Descrição	Ferramenta gratuita que ajuda arquitetos de software, mesmo que não sejam peritos em segurança, a entenderem riscos inerentes aos programas que estão criando.
Onde obter	http://www.microsoft.com/downloads/details.aspx?FamilyID=A48CCCB1-814B-47B6-9D17-1E273F65AE19&displaylang=em

Ferramenta	**Microsoft Threat Analysis & Modeling**
Descrição	Ferramenta gratuita que permite que especialistas em assuntos não relacionados à segurança insiram informações conhecidas, incluindo requisitos de negócios e arquitetura de aplicativo, que são usadas para produzir um modelo contra ameaças.
Onde obter	http://www.microsoft.com/downloads/details.aspx?FamilyID=59888078-9daf-4e96-b7d1-944703479451&displaylang=em

Ferramenta	**Netsparker Community Edition**
Descrição	Aplicativo gratuito para detectar SQL Injection + cross-site scripting. Realiza varredura no site e apresenta soluções para as possíveis questões apresentadas.
Onde obter	http://www.mavitunasecurity.com/communityedition/

Ferramenta	**Websecurify**
Descrição	Ferramenta *open source* que identifica automaticamente as vulnerabilidades da aplicação web. Pode criar relatórios simples, que podem ser exportados para vários formatos.
Onde obter	http://www.websecurify.com/

Ferramenta	**Wapiti**
Descrição	Ferramenta web *open source* que verifica as páginas de sites e aplicativos web procurando por scripts e formulários que podem injetar dados. Consegue detectar erros de manuseio de arquivos, *SQL Injection*, *Cross-Site Scripting*, fixação de sessões, entre outros.
Onde obter	http://www.ict-romulus.eu/web/wapiti/home

Ferramenta	**N-Stalker**
Descrição	Edição gratuita para verificar até 100 páginas de uma só vez, para realização de vários testes de segurança, incluindo *Cross-Site Scripting.*

Onde obter	http://nstalker.com/products/free

Ferramenta	**Skipfish**
Descrição	Ferramenta gratuita para teste de segurança em sites. Possui diversos tipos de testes de segurança, incluindo *Blind SQL Injection.*
Onde obter	http://code.google.com/p/skipfish/

Ferramenta	**Scrawlr**
Descrição	Software gratuito para escanear vulnerabilidades de *SQL Injection* em aplicações web. Desenvolvida pelo *HP Web Security Research Group* juntamente com o *Microsoft Security Response Center.*
Onde obter	http://www.communities.hp.com/securitysoftware/blogs/spilabs/archive/2008/06/23/finding-sql-injection-with-scrawlr.aspx

Ferramenta	**Fiddler**
Descrição	Ferramenta para depuração de páginas da internet, que grava um histórico de todo o tráfego HTTP entre o computador e a internet. Permite acompanhamento em tempo real, enquanto a página é aberta.
Onde obter	http://www.fiddlertool.com/

Ferramenta	**Watcher**
Descrição	*Plugin* para o Fiddler. Funciona como uma ferramenta de análise passiva para aplicações web baseadas em HTTP. Identifica questões como *POSTs cross-domain*, comutação perigosa entre HTTP e HTTPS, dentre outros.
Onde obter	http://websecuritytool.codeplex.com/

Ferramenta	**x5s**
Descrição	*Plugin* para o Fiddler. Projetado para trabalhar com falhas de segurança relativas à vulnerabilidades de *XSS*. Realiza testes de input com caracteres especiais como “<” e “>”, e analisa a saída resultante.
Onde obter	http://xss.codeplex.com/

Ferramenta	**Exploit-Me**
Descrição	Plugin para o Firefox que realiza testes pelo próprio browser, sem o uso de proxy. É um “combinado” de 3 plugins de segurança: • XSS-Me: Para testes de XSS. (https://addons.mozilla.org/en-US/firefox/addon/7598); • SQL Inject Me: Para testes de vulnerabilidades com SQL injection. (https://addons.mozilla.org/en-US/firefox/addon/7597); • Access-Me: Para testes de segurança com vulnerabilidades de acesso. (https://addons.mozilla.org/en-US/firefox/addon/7595).
Onde obter	http://labs.securitycompass.com/index.php/exploit-me/

Ferramenta	**Acunetix**
Descrição	Versão gratuita que executa testes de segurança para identificar vulnerabilidades de *Cross Site Scripting* (XSS).
Onde obter	http://www.acunetix.com/cross-site-scripting/scanner.htm

Ferramenta	Watcher
Descrição	*Plugin* para o Fiddler. Funciona como uma ferramenta de análise passiva para aplicações web baseadas em HTTP. Identifica questões como *POSTs cross-domain*, comutação perigosa entre HTTP e HTTPS, dentre outros
Onde obter	http://websecuritytool.codeplex.com/

Ferramenta	x5s
Descrição	*Plugin* para o Fiddler. Projetado para trabalhar com falhas de segurança relativas à vulnerabilidades de *XSS*. Realiza testes de input com caracteres especiais como “<” e “>”, e analisa a saída resultante
Onde obter	http://xss.codeplex.com/

Ferramenta	Exploit-Me
Descrição	Plugin para o Firefox que realiza testes pelo próprio browser, sem o uso de proxy. É um “combinado” de 3 plugins de segurança: • XSS-Me: Para testes de XSS. (https://addons.mozilla.org/en-US/firefox/addon/7598); • SQL Inject Me: Para testes de vulnerabilidades com SQL injection. (https://addons.mozilla.org/en-US/firefox/addon/7597); • Access-Me: Para testes de segurança com vulnerabilidades de acesso. (https://addons.mozilla.org/en-US/firefox/addon/7595).
Onde obter	http://labs.securitycompass.com/index.php/exploit-me/

Ferramenta	Acunetix
Descrição	Versão gratuita que executa testes de segurança para identificar vulnerabilidades de *Cross Site Scripting* (XSS)
Onde obter	http://www.acunetix.com/cross-site-scripting/scanner.htm

6.6. Modelo MVC

Durante a fase de elaboração, deve ser considerada a possibilidade de construção do sistema seguindo o modelo MVC.

O MVC (*Model – View - Controller*) é um padrão de desenvolvimento que divide, em camadas, as três partes vitais de um sistema: modelo (*Model*), visualização (*View*) e controle (*Controller*). Desta forma, o sistema torna-se flexível e com nível de segurança modular independente.

Camada de Modelo: Responsável pelo gerenciamento dos dados do sistema;

Camada de Visualização: Camada de interface com o usuário, responsável pela apresentação da informação e pela entrada de dados;

Camada de Controle: Recebe as informações da entrada e as transmite para o modelo. É a camada onde são implementadas as regras de negócio do sistema.

6.7. Modelos

6.7.1. Casos de uso

Uma forma eficiente de identificar as funções de segurança e suas ameaças é empregando casos de uso, que podem ser categorizados, por exemplo:

- Controle de acesso
  - Fingir ser um usuário válido
  - Roubar credenciais de acesso
  - Engenharia social
- Integridade
  - Alteração indevida de dados
  - Alteração de mensagens
  - Alteração do funcionamento do sistema
- Privacidade
  - Privacidade de dados
  - Privacidade de mensagens do sistema
  - Privacidade de dados do usuário

As informações prestadas nos casos de uso deverão ser claras e sintéticas. As condições para que os eventos ocorram também deverão ser enumerados.

Exemplo de tabela para o estudo de casos de uso:

<table>
<tr><td colspan="4">Caso de uso: Privacidade</td></tr>
<tr><td colspan="4">Caso específico: Privacidade de mensagens do usuário</td></tr>
<tr><td colspan="4">Ameaça: um usuário externo obter acesso a uma mensagem do usuário para o sistema</td></tr>
<tr><td colspan="4">Pré-condições: 1) o atacante tem meios de acessar as mensagens do usuário; 2) o sistema solicitou dados privados do usuário</td></tr>
<tr><td rowspan="2">Ações do usuário</td><td rowspan="2">Ações do atacante</td><td colspan="2">Sistema</td></tr>
<tr><td>Interações do sistema</td><td>Ações do sistema</td></tr>
<tr><td>Enviar mensagem privada</td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>Tornar a mensagem ilegível durante a transmissão</td></tr>
<tr><td></td><td>Interceptar a mensagem do usuário</td><td></td><td></td></tr>
<tr><td colspan="4">Pós-condições: o sistema tem que garantir que um usuário não autorizado não pode ler mensagens privadas</td></tr>
</table>

6.7.2. Qualificação do sistema e plano de segurança

1. Nome do sistema

Identificador único e o nome do sistema. EX: SistemaX_01.

2. Data da elaboração do plano de segurança

Data da criação do plano.

3. Data da atualização do plano de segurança

Data da atualização do plano.

4. Versão do plano de segurança

Versão do documento, de acordo com suas atualizações

5. Propostas e objetivos do sistema

Descrição das funcionalidades e propostas do sistema.

6. Status operacional do sistema

Relatar qual o status do sistema. Operacional – Em Desenvolvimento – Em Mudança.

Status	X
Operacional	
Em Desenvolvimento	
Em Mudança	
Em Planejamento	

7. Tipo do sistema

Relatar o tipo do sistema. Sistema Principal – Sistema Auxiliar.

Tipo	X
Sistema Principal	
Sistema Auxiliar	

8. Descrição do ambiente do sistema

Descrição do ambiente físico e lógico no qual o sistema será implantado.

9. Planejamento preliminar da segurança do sistema

Descrição das expectativas de segurança e calendário inicial de atividades de segurança.

10. Matriz de responsabilidades

Papel	Nome	Setor	e-mail	Telefone
Superintendência de Tecnologia da Informação				
Chief Information Officer (CIO)				
Diretor de Qualidade e Testes				
Proprietário do Sistema				
Gerente de Contratos				
Analista de Contratos				
Gerente de Autorização				
Chief Information Security Officer				
Gerente de Segurança				
Analista de Privacidade				
Gerente de Configurações				
Gerente de Desenvolvimento				
Arquiteto de Sistemas				
Desenvolvedor/ Programador				
Jurídico				
Outros Participantes				

a. Conflitos entre papéis

Listar conflitos em relação a papéis que não podem ser desempenhados por um mesmo profissional, justificando o motivo e referenciando o documento em que o proprietário do sistema aprova a decisão.

11. Categorização do sistema

Nível em que o sistema foi categorizado, após a classificação das informações;
Requerimentos de segurança de alto nível.

12. Avaliação de impacto no negócio

Linhas de serviço suportadas pelo sistema, como as mesmas serão impactadas e qual o tempo de tolerância até a recuperação em caso de interrupção;
Componentes mínimos necessários para o funcionamento do sistema.

13. Avaliação de impacto na privacidade

Detalhamento de onde e por qual motivo as informações privadas são coletadas, armazenadas e/ou criadas dentro do sistema.

Informação	Importância	Utilização	Proprietário	Onde é coletada	Onde é armazenada

14. Padrões definidos

Plano de verificação de qualidade, entregas e marcos do projeto;
Padrões definidos para desenvolvimento e codificação.

15. Legislação pertinente

Leis ou regulamentos que estabeleçam requisitos específicos de confidencialidade, integridade e disponibilidade.

16. Avaliação de riscos

a. Ameaças identificadas

Listar as ameaças identificadas.

ID	Ameaça	Descrição	Potenciais alvos	Motivação

b. Vulnerabilidades identificadas

Listar as vulnerabilidades identificadas.

ID	Vulnerabilidade	Ameaças	Ativo

c. Riscos do sistema

Documentar os resultados da análise de riscos.

ID	Origem	Risco	Possíveis Impactos ao negócio	Recomendações	Probabilidade	Impacto	Nível de Risco

17. Requerimentos de interoperabilidade.

Lista na qual serão descritos os sistemas para os quais serão aplicados controles de interoperabilidade e os requerimentos adotados.

Sistema	Organização	Versão	Requerimentos e-Ping

18. Controles de segurança

a. Níveis de segurança – Classes/Famílias

Definir implementações de acordo com as classes e famílias do nível de segurança requerido pelo sistema. Pode utilizar referências de controles técnicos do manual.

Classe Funcional		Família (ISO 15408-2)		Implementação
Auditoria de Segurança	FAU	Resposta Automática de Auditoria	ARP	
		Geração de Dados de Auditoria	GEN	
		Análise da Auditoria de Segurança	SAA	
		Revisão da Auditoria de Segurança	SAR	
		Seleção de Eventos da Auditoria de Segurança	SEL	
		Armazenamento dos Eventos de Auditoria de Segurança	STG	
Comunicação	FCO	Irretratabilidade da Origem	NRO	
		Irretratabilidade do Destinatário	NRR	
Criptografia	FCS	Gerenciamento de Chaves Criptográficas	CKM	
		Operação Criptográfica	COP	
Proteção de Dados do Usuário	FDP	Política de Controle de Acesso	ACC	
		Funções de Controle de Acesso	ACF	
		Autenticação de Dados	DAU	
		Exportação de Dados	ETC	
		Política do Fluxo de Informação	IFC	
		Funções do Fluxo de Informação	IFF	
		Importação de Dados	ITC	
		Transferência Interna de Dados	ITT	
		Proteção da Informação Residual	RIP	

		Reversão	ROL	
		Integridade de Dados Armazenados	SDI	
		Confidencialidade dos Dados em Trânsito dos Usuários	UCT	
		Integridade de Dados em Trânsito dos Usuários	UIT	
Identificação e Autenticação	FIA	Falhas de Autenticação	AFL	
		Definição de Atributos de Usuários	ATD	
		Especificação de Segredos	SOS	
		Autenticação de Usuários	UAU	
		Identificação de Usuários	UID	
		Ligação usuário-assunto	USB	
Gerenciamento de Segurança	FMT	Gerenciamento de funções do sistema	MOF	
		Gerenciamento de atributos de segurança	MSA	
		Gerenciamento de dados das funções de segurança	MTD	
		Revogação	REV	
		Validade dos atributos de segurança	SAE	
		Especificação das funções de gerenciamento	SMF	
		Regras de gerenciamento de segurança	SMR	
Privacidade	FPR	Anonimato	ANO	
		Pseudoanonimato	PSE	
		Desvinculação	UNL	
		Restrição à interceptação	UNO	
Proteção das Funcionalidades de Segurança	FPT	Falha segura	FLS	
		Disponibilidade de dados exportados da aplicação	ITA	
		Confidencialidade de dados exportados da aplicação	ITC	
		Integridade dos dados exportados da aplicação	ITI	
		Segurança nas transferências internas de dados	ITT	
		Segurança física para a aplicação	PHP	
		Recuperação confiável	RCV	
		Detecção de repetição (replay)	RPL	
		Protocolo de sincronia de estado	SSP	

		Carimbos de tempo	STM	
		Consistência de dados	TDC	
		Teste de sistemas externos	TEE	
		Consistência na replicação de dados	TRC	
		Auto-teste da aplicação	TST	
Utilização de Recursos	FRU	Tolerância a falhas	FLT	
		Prioridade do serviço	PRS	
		Alocação de recursos	RSA	
Acesso ao Sistema	FTA	Limitação da sessão	LSA	
		Limitações em sessões concorrentes	MCS	
		Bloqueio e encerramento de sessão	SSL	
		Avisos de acesso	TAB	
		Histórico de acesso	TAH	
		Estabelecimento de sessão	TSE	
Canais de Confiança	FTP	Canais de confiança entre sistemas	ITC	
		Canais seguros	TRP	

b. Controles adicionais

Listar controles adicionais, baseados em condições específicas do sistema.

c. Técnicas de programação segura (cap 4)

Relatar os controles técnicos e/ou soluções adotadas.

d. Proteção de dados (cap 5)

Relatar os controles técnicos e/ou soluções adotadas.

e. Observações

Observações gerais, problemas identificados, limitações, sugestões de melhoria.

19. Arquitetura de segurança

Esquema gráfico ou descrição de como os controles de segurança serão integrados ao sistema;

Listas de serviços compartilhados e riscos compartilhados resultantes;

Identificação de controles comuns usados pelo sistema.

20. Resultados dos testes

Relatar quais tipos de teste foram executados no sistema.

Testes	X
Casos de uso, erro e abuso do sistema	
Modelos de ataque	
Revisão de desenho	
Análise de riscos	
Análise de impacto	
Provas formais	
Revisão de código	
Detecção em tempo de compilação	
Análise estática em injeção de falhas	
Testes nebulosos	
Análise de código binário	
Análise de vulnerabilidades	
Análise estática	
Injeção de falhas em código fonte	
Injeção de falhas em código binário	
Teste de penetração	
Testes de regressão	

Documentar os resultados dos testes aplicados ao sistema, relatando qualquer variação inesperada e verificando se o sistema foi desenvolvido de acordo com os requerimentos funcionais e de segurança.

21. Riscos residuais

Descrever os riscos residuais, ou seja, que restaram após a implementação dos controles, especificando se foram aceitos pelo Gerente de Autorização.

22. Controles operacionais

Descrever os controles de segurança operacionais, e como os mesmos serão integrados ao ambiente de produção do sistema.

23. Resultado da avaliação de segurança

Descrever os resultados da avaliação de segurança do sistema.

24. Validação do sistema

Item de	Avaliador 1		Avaliador 2	

avaliação	Data	Parecer	Data	Parecer
Documentação				
SDLC				
Testes				

Registrar os resultados das atividades de validação do sistema.

25. Decisão de autorização do sistema

Identificar por quem e quando o sistema foi autorizado, e se a autorização foi endossada pelo Gerente de Autorização.

26. Ferramentas de monitoração

Listar as ferramentas utilizadas para monitorar o sistema.

27. Gestão de Mudanças (Resumo)

Relacionar as demandas de manutenção, atualização, modificação e melhorias do sistema.

Demanda	Solicitante	Autorizante	Data da Implementação	Obs. Implementação

28. Histórico de atualizações

Relacionar todas as atualizações. Autor – Data – Descrição resumida da atualização.

Autor	Data	Descrição

6.7.3. Estrutura do relatório de avaliação de sistemas

Parte I – Sumário Executivo

Texto orientado para gestores de alto nível e proprietários do sistema, para que entendam as principais ameaças do sistema e a efetiviadade das contramedidas adotadas.

Deve conter o resultado da avaliação e a indicação dos próximos passos. O texto deve evitar o linguajar técnico.

Parte II – Panorama Técnico

Texto orientado aos gerentes de áreas técnicas, abordando os controles para disponibilidade, integridade e confidencialidade do sistema, em maior profundidade que no sumário executivo.

Deverá haver uma lista das principais ameaças, seus níveis de risco e as contramedidas adotadas.

Parte III – Especificidades

Detalhamentos das falhas encontradas e necessidades de correção, que podem ser expostos em uma tabela (dados meramente ilustrativos).

Categoria	Código	Teste	Resultado	Solução	Responsável
Autenticação	*A01*	*Força bruta*	*Foram encontradas senhas fracas*	*Reforçar e implantar política de senhas*	*Desenvolvedor*

6.7.4. Programa de trabalho

- Programa de avaliação de pontos de controle

#	Ponto de controle	Fase concluída	Fase iniciada	Atividades	Papéis de trabalho	Concluído (data)	Exceção	Resultado
1	*Ponto de controle a ser verificado.*	*Nome da fase do SDLC finalizada com a validação do ponto de controle.*	*Nome da fase do SDLC iniciada após a validação do ponto de controle.*	*Descrever as atividades a serem realizadas para a validação do ponto de controle, ou seja, o passo a passo do teste, descrevendo os itens que serão avaliados, e quais evidências serão solicitadas.* *Exemplo:* *1) Solicitar documento...;* *2) Verificar os itens...;* *3) Solicitar evidência...;* *4) Verificar conformidade.*	*Anexar o arquivo ou referenciar o papel de trabalho gerado para a validação do ponto de controle.*	*Data em que foi concluída a atividade de validação.*	*Especificar se foi identificada alguma exceção, ou seja, se algo não está em conformidade com o esperado (Sim; Não).*	***Ponto:** Descrever, uma a uma, as exceções identificadas.*
2	*Ponto de controle a ser verificado.*	*Nome da fase do SDLC finalizada com a validação do ponto de controle.*	*Nome da fase do SDLC iniciada após a validação do ponto de controle.*	*Descrever as atividades a serem realizadas para a validação do ponto de controle, ou seja, o passo a passo do teste, descrevendo os itens que serão avaliados, e quais evidências serão solicitadas.* *Exemplo:* *1) Solicitar documento...;* *2) Verificar os itens...;* *3) Solicitar evidência...;* *4) Verificar conformidade.*	*Anexar o arquivo ou referenciar o papel de trabalho gerado para a validação do ponto de controle.*	*Data em que foi concluída a atividade de validação.*	*Especificar se foi identificada alguma exceção, ou seja, se algo não está em conformidade com o esperado (Sim; Não).*	***Ponto:** Descrever, uma a uma, as exceções identificadas.*

- Programa de avaliação de documentação

#	Documento	Fase de origem	Atividades	Papéis de trabalho	Concluído (data)	Exceção	Resultado
1	*Documento a ser verificado.*	*Nome da fase do SDLC onde é gerado o documento.*	*Descrever os itens e as propriedades a serem verificadas no documento para que o mesmo seja validado.* *Exemplo:* *1) Verificar os itens...;* *2) Verificar se o documento está completo;* *3) Verificar a conformidade com o processo;* *4) Verificar se o documento foi assinado por...*	*Anexar o arquivo ou referenciar o papel de trabalho gerado para a validação do documento.*	*Data em que foi concluída a atividade de validação.*	*Especificar se foi identificada alguma exceção, ou seja, se algo não está em conformidade com o esperado (Sim; Não).*	***Ponto:*** *Descrever, uma a uma, as exceções identificadas.*
2	*Documento a ser verificado.*	*Nome da fase do SDLC onde é gerado o documento.*	*Descrever os itens e as propriedades a serem verificadas no documento para que o mesmo seja validado.* *Exemplo:* *1) Verificar os itens...;* *2) Verificar se o documento está completo;* *3) Verificar a conformidade com o processo;* *4) Verificar se o documento foi assinado por...*	*Anexar o arquivo ou referenciar o papel de trabalho gerado para a validação do documento.*	*Data em que foi concluída a atividade de validação.*	*Especificar se foi identificada alguma exceção, ou seja, se algo não está em conformidade com o esperado (Sim; Não).*	***Ponto:*** *Descrever, uma a uma, as exceções identificadas.*

- Programa de testes

#	Teste	Fase	Documento base	Objetivo do teste	Atividades	Papéis de trabalho	Concluído (data)	Exceção	Resultado
1	*Tipo de teste a ser executado.*	*Nome da fase do SDLC em que o teste será aplicado.*	*Nome do documento que será utilizado como base para o teste;*	*Descrever o objetivo da realização do teste, ou seja, o que será verificado.*	*Descrever as atividades a serem realizadas para a realização do teste, ou seja, o passo a passo do teste, descrevendo os itens que serão testados, e quais evidências serão solicitadas.* *Exemplo:* *1) Aplicar técnica de...;* *2) Simular ataque de...;* *3) Coletar evidências do...;* *4) Análisar resultados.*	*Anexar o arquivo ou referenciar o papel de trabalho gerado para o teste de validação.*	*Data em que foi concluída a atividade de validação.*	*Especificar se foi identificada alguma exceção, ou seja, se algo não está em conformidade com o esperado (Sim; Não).*	***Ponto:*** *Descrever, uma a uma, as exceções identificadas.*
2	*Tipo de teste a ser executado.*	*Nome da fase do SDLC em que o teste será aplicado.*	*Nome do documento que será utilizado como base para o teste;*	*Descrever o objetivo da realização do teste, ou seja, o que será verificado.*	*Descrever as atividades a serem realizadas para a realização do teste, ou seja, o passo a passo do teste, descrevendo os itens que serão testados, e quais evidências serão solicitadas.* *Exemplo:* *1) Aplicar técnica de...;* *2) Simular ataque de...;* *3) Coletar evidências do...;* *4) Análisar resultados.*	*Anexar o arquivo ou referenciar o papel de trabalho gerado para o teste de validação.*	*Data em que foi concluída a atividade de validação.*	*Especificar se foi identificada alguma exceção, ou seja, se algo não está em conformidade com o esperado (Sim; Não).*	***Ponto:*** *Descrever, uma a uma, as exceções identificadas.*

6.7.5. Papel de trabalho

Localização:	*Local onde a etapa de validação é realizada*.
Projeto:	*Nome do projeto*
Validação:	*Ponto de controle, documento ou teste que será validado*

Preparado por: *Nome do profissional* **Data:**

Revisado por: *Nome do profissional* **Data:**

Referência para o Programa de Trabalho: *Item correspondente*

Objetivo

Descrever o objetivo da validação ou realização do teste, e o que será verificado.

Para a validação, efetuamos os seguintes procedimentos:

1. P*asso 1*;

2. *Passo 2*;

3. *Passo 3*.

Evidências

Referência às evidências coletadas.

Resultados Obtidos

☑ *Descrever aspectos em conformidade com o esperado*.

☒ *Descrever as exceções identificadas (Estas exceções se tornarão pontos do relatório de auditoria)*.

Legenda: ☑ Justificativa Aceita ☒ Justificativa Rejeitada

Conclusão

Válido / Inválido

6.7.6. Relatório de auditoria

1. Distribuição

Nome	Área
Nome do profissional que receberá o relatório	*Área na qual o profissional atua*
Nome do profissional que receberá o relatório	*Área na qual o profissional atua*

Data	*Data de emissão do relatório*

2. Escopo e Objetivos do Trabalho

Projeto:	***Nome do projeto***
Escopo do trabalho:	*Descrever o escopo dos trabalhos da auditoria ou validação realizados.*

3. Considerações Finais

Descrever as considerações finais sobre os testes e as etapas de validação executadas, e os resultados encontrados.

4. Relação Analítica das Questões e Recomendações Detalhadas

Detalhar cada exceção identificada e documentada no programa de trabalho e nos papéis de trabalho.

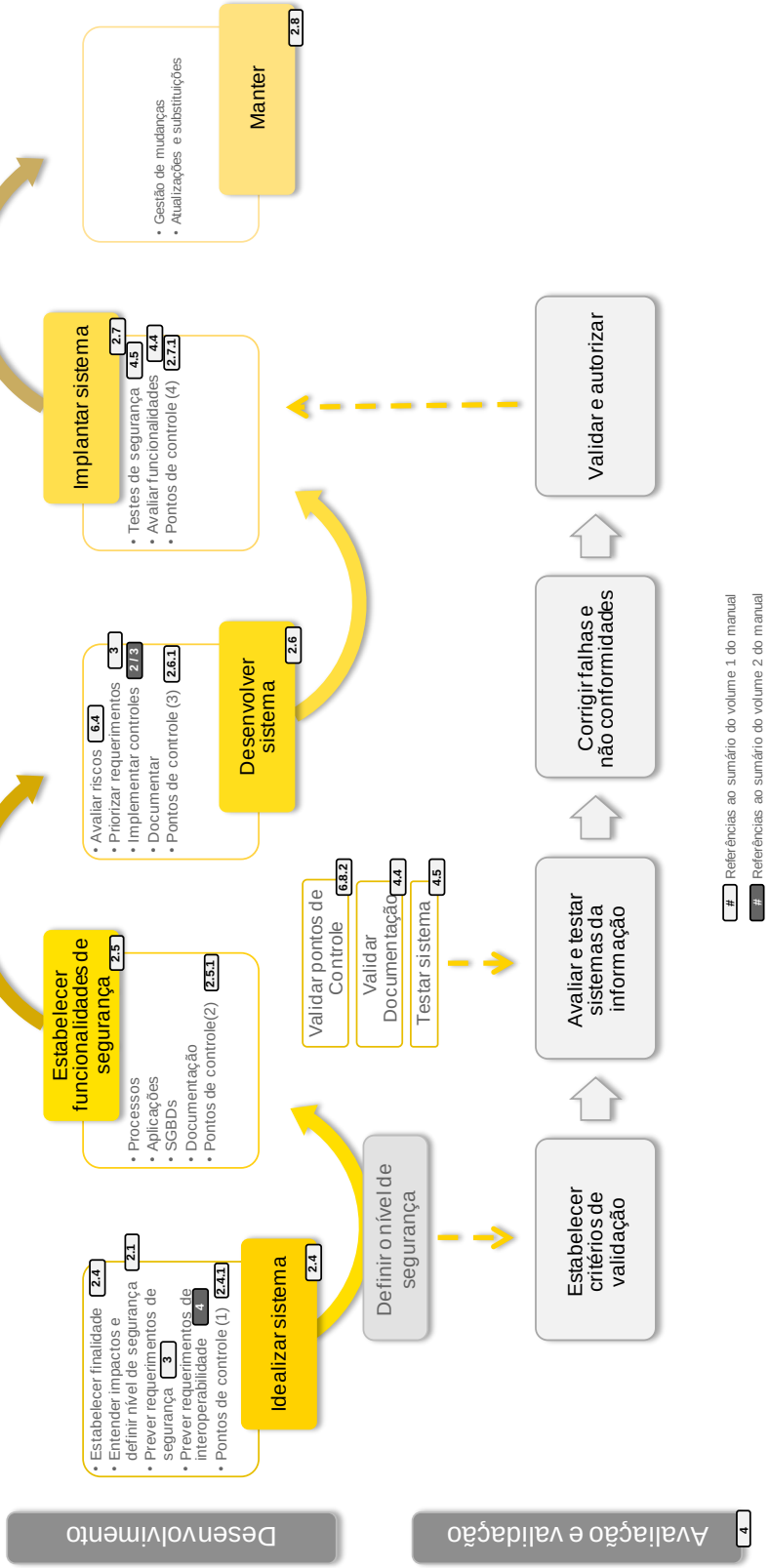
REF	TÍTULO DO APONTAMENTO
1	***Ponto (Exceção) identificado***
DESCRITIVO	
Descrição do ponto identificado e quais os seus impactos.	
RECOMENDAÇÃO	
Recomendações para eliminação do ponto.	
COMENTÁRIOS DA ÁREA - (*Responsável pela área à qual o ponto foi atribuído*)	
Comentários da área sobre o apontamento, especificando que medidas a área pretende adotar para eliminar o ponto.	

5. SUMÁRIO DE PLANOS DE AÇÃO

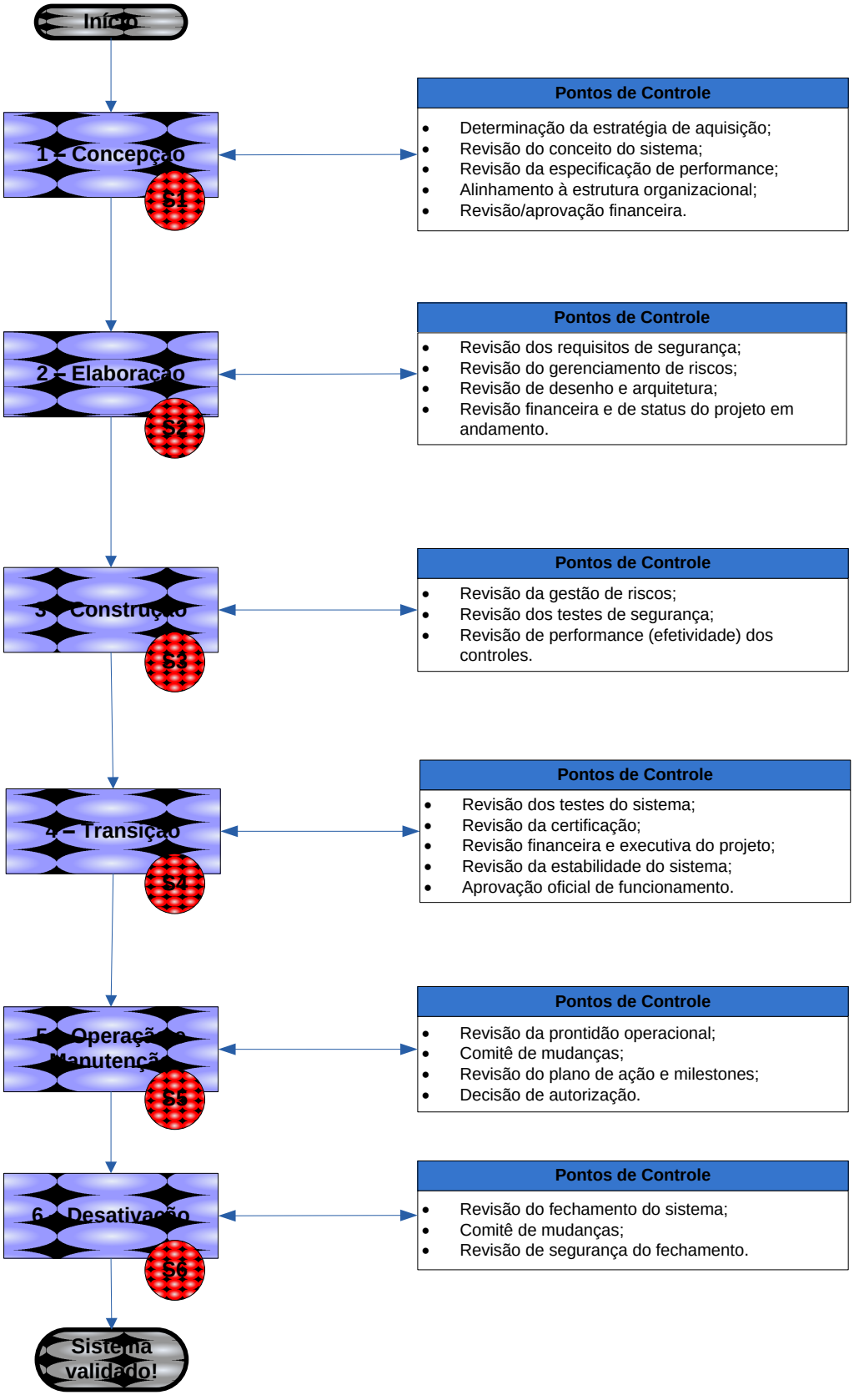
REF	APONTAMENTO	PLANO DE AÇÃO	RESPONSÁVEL
1	*Apontamento o qual a ação pretende eliminar*	*Definir o plano de ação e as medidas que serão adotadas.*	*Nome do responsável pelo plano de ação*

6.8. Diagramas

6.8.1. Guia rápido de desenvolvimento seguro



6.8.2. Validação de sistemas (com pontos de controle)



FIM DO DOCUMENTO