

Plano de Vulnerabilidades

Este plano descreve todas as categorias de vulnerabilidades em nível de aplicação web e as determinadas medidas adotadas para inibir os ataques.

ENTRADA DE DADOS:

ATAQUE:	Buffer Overflow
DESCRIÇÃO:	Consiste no atacante explorar o sistema de entrada de informações da solução enviando alto volume de dados inconsistentes e maliciosos.
RISCO:	1-Indisponibilidade da solução (crashes)
SOLUÇÃO:	Usar uma política de validação sistemática voltadas a tipo de dados que expresse tamanho, formato e intervalo. Use expressão regular para validar: - campos de tipos de dados. - whitelist de caracteres permitidos na validar campos textos puros.

ATAQUE:	XSS Script
DESCRIÇÃO:	Consiste no atacante explorar o sistema de entrada de informações da solução enviando comandos maliciosos de HTML, DHTML e JavaScript para serem executados nos navegadores .
RISCO:	1-Sequestro do identificador da sessão autenticada e consequentemente acesso ao sistema como se fosse o usuário autenticado. 2- Indisponibilidade da solução.
SOLUÇÃO:	1-Usar uma política de validação sistemática voltadas a tipo de dados que expresse tamanho, formato e intervalo. Use expressão regular para validar: - campos de tipos de dados. - whitelist de caracteres permitidos na validar campos textos puros. 2- Todas as informações do banco de dados devem renderizados nas páginas como "texto puro" e não como tecnologias de apresentação. (Outros sistemas legados que também acessem o mesmo banco podem ter e essa vulnerabilidade) 3-Configurar especificação do cookies useHttpOnly, inibindo o sequestro do identificador de sessão.

ATAQUE:	SQL Injection
DESCRIÇÃO:	Consiste no atacante explorar o sistema de entrada de informações da solução enviando instruções SQL maliciosos.
RISCO:	1-Acessar informações não permitidas nos processos da solução. 2-Acesso total nas estruturas e os dados do banco de dados.
SOLUÇÃO:	1-Utilizar somente comandos SQL pré-compilados na camada de persistência. 2-Usar uma política de validação sistemática voltadas a tipo de dados que expresse tamanho, formato e intervalo. Use expressão regular para validar: - campos de tipos de dados. - whitelist de caracteres permitidos na validar campos textos puros.

ATAQUE:	JavaScript Off
DESCRIÇÃO:	Consiste no atacante desabilitar os JavaScript durante as requisições.

RISCO:	Desvio das validações e condições pré-estabelecidas pela solução.
SOLUÇÃO:	As validações e condições devem ser replicadas ou exclusivamente centralizadas no servidor usando tecnologia “server-side” (centralizadas na camada visão ou de negócio da solução). Nunca confie nos dados vindos do protocolo HTTP.

ATAQUE:	Malicious File Execution
DESCRIÇÃO:	Consiste no atacante enviar um arquivo em algum processo de upload da solução que possa ser executado dentro do servidor, comprometendo os fontes da aplicação ou o S.O. do servidor.
RISCO:	Indisponibilidade da solução uma vez que a execução pode comprometer os fontes da solução e ou o sistema operacional hospedeiro.
SOLUÇÃO:	1 - Deve ser feitas todos os tipos de validações, restringindo estritamente os tipos de arquivos esperado pela solução. 2 – O usuário de execução do serviço da solução não pode ter acesso total aos recursos de S.O. hospedeiro.

ATAQUE:	Cross Site Request Forgery - CSRF
DESCRIÇÃO:	Consiste na execução de requisições HTTP fantasmas durante uma sessão autenticada sem o consentimento do usuário. Acontece através de links por spam de e-mails maliciosos, sites maliciosos e vírus alocado na estação do usuário.
RISCO:	Executar operações transacionais maliciosas.
SOLUÇÃO:	1 – Aplicar synchronized token em cada operação crítica transacional. 2 – Requisitar uma senha de identificação (a mesma de senha de login ou outra senha de operação, cartão de senhas ou token randômicos) a cada operação crítica transacional.

AUTENTICAÇÃO:

ATAQUE:	Dictionary
DESCRIÇÃO:	O atacante usa uma base de informações previamente configuradas de acordo com o perfil do usuário, juntamente com algum robô (algoritmo computacional) que gera combinações de possíveis credenciais.
RISCO:	Descobrir as credenciais e ter acesso à solução.
SOLUÇÃO:	1 – Uso sistemático de políticas de senhas - force strong password, password expiration, password reset, password history (pelo menos as três últimas).

ATAQUE:	Brute Force
DESCRIÇÃO:	Consiste no atacante utilizar de recursos automatizados para descobrir as credenciais de acesso, executando esse robô na solução. O atacante usa uma “base de senhas” previamente gerada ou configurada em um “dictionary”.
RISCO:	Descobrir as credenciais e ter acesso à solução
SOLUÇÃO:	Bloquear logicamente o usuário após exceder o numero limite de tentativas. (try block password)

ATAQUE:	Men in the Middle – MITM e
DESCRIÇÃO:	Consiste no atacante de posse de determinadas ferramentas interceptar o trafico HTTP, tendo acesso a todas as informações trafegadas dentro do protocolo HTTP.
RISCO:	1 – Sequestro do identificador da sessão e ter acesso à solução. 2 – Acesso total as informações trafegadas. 3 – Adulteração de qualquer informação trafegada (Ataque chamado de “Data Tampering”)
SOLUÇÃO:	Usar canal criptografado HTTPS,

ATAQUE:	Session Tampering
DESCRIÇÃO:	Consiste no atacante adivinhar a sequencia de geração do ID de sessões da solução.
RISCO:	Sequestro do identificador da sessão e ter acesso à solução.
SOLUÇÃO:	A solução deve usar geração forte de id de sessões não sequencias.

ATAQUE:	Session Replay
DESCRIÇÃO:	Consiste no atacante conseguir sequestrar de alguma forma (vírus, trojan, horse ou worms, Men in the Middle, Cross Site Request Forgery) o identificador da sessão do usuário autenticado. O atacante que esta fora da rede do usuário vitima, tenta usar a solução com o ID da sessão sequestrada.
RISCO:	Acesso à solução como se fosse o usuário autenticado.
SOLUÇÃO:	A solução deve vincular o identificador da sessão do usuário autenticado com o seu IP origem autenticada, verificando a consistência a cada requisição HTTP. (Esse solução não funcionara caso existe algum usuário que use um rede que tenha load-balancer de proxy)

ATAQUE:	Session Replay
DESCRIÇÃO:	Consiste no atacante conseguir sequestrar de alguma forma (vírus, trojan, horse ou worms, Men in the Middle, Cross Site Request Forgery) o identificador da sessão do usuário autenticado. O atacante que esta dentro da mesma rede do usuário vitima, tenta usar a solução com o ID da sessão sequestrada.
RISCO:	Acesso à solução como se fosse o usuário autenticado.
SOLUÇÃO:	Requisitar uma senha de identificação (a mesma de senha de login ou outra senha de operação, cartão de senhas ou token randômicos) a cada operação critica transacional.

ATAQUE:	Session Fixation
DESCRIÇÃO:	Consiste no atacante conseguir sequestrar de alguma forma (vírus, trojan, horse ou worms, Men in the Middle, Cross Site Request Forgery) o identificador da sessão do usuário autenticado durante o uso sem SSL. Depois do usuário se autenticar, a solução ativa o canal SSL, mas mantém o mesmo numero de identificador da sessão.
RISCO:	Acesso à solução como se fosse o usuário autenticado.
SOLUÇÃO:	O ID da sessão deve ser alterado cada vez que a partes da solução estabelecer comunicação SSL em páginas restritas.

ATAQUE:	Key Logger
DESCRIÇÃO:	Consiste no atacante conseguir de alguma forma (vírus, trojan, horse ou worms) a senha do usuário autenticado. O vírus instalado no PC do usuário intercepta todas as teclas pressionadas no teclado.
RISCO:	Obtenção de credenciais e conseqüentemente acesso à solução como se fosse o usuário autenticado.
SOLUÇÃO:	A solução deve fazer o usuário digitar a senha de autenticação de forma que não use o teclado – Teclado Virtual.

ATAQUE:	Mouse Logger
DESCRIÇÃO:	Consiste no atacante conseguir de alguma forma (vírus, trojan, horse ou worms) a senha do usuário autenticado. O vírus instalado no PC do usuário intercepta e copia os pedaços das áreas visuais pressionadas pelo mouse.
RISCO:	Obtenção de credenciais e conseqüentemente acesso à solução como se fosse o usuário autenticado.
SOLUÇÃO:	1 - O teclado virtual oferecido pela solução deve ser embaralhado a cada acesso. 2 – As informações do teclado virtual (letras, números ou qualquer caracteres) oferecido pelo sistema devem ser apagadas quando o usuário passar e clicar o mouse sobre eles.

ATAQUE:	Insecurity Storage
DESCRIÇÃO:	Consiste no atacante ser alguém interno ter acesso ao banco de dados de credenciais ou informações críticas da solução.
RISCO:	1-Acessar as credenciais e ter acesso à solução. 2-Acessar informações da solução com cartões de credito, registros médicos, informações pessoais, cupons de desconto, etc.
SOLUÇÃO:	1-Criptografar todas as informações sensíveis a solução.

AUTORIZAÇÃO:

ATAQUE:	Unrestricted URL Access
DESCRIÇÃO:	Consiste no atacante descobrir que existe na solução URL's de recursos esquecidos de ser restringindo com a devida autenticação e autorização.
RISCO:	1-Visualizar informações confidenciais. 2-Fazer transações indevidas.
SOLUÇÃO:	1-Efetuar testes sistemáticos de autenticação e autorização em todas os recursos (públicas e privadas). 2-Implementar no mecanismo de autenticação e autorização de recursos que automaticamente não permita o acesso nas URL's da solução sem nenhum tipo de declaração(esquecidas), gerando um log de aviso em runtime. Ou seja, todas as URL's da solução devem ser mapeadas como "pública" ou "privada" com sua devida autorização. Caso aconteça de uma URL ser esquecida, esse mecanismo não permite ninguém acessar, gerando um log de aviso.

ATAQUE:	Forceful Browsing
DESCRIÇÃO:	Consiste no atacante descobrir que existe na solução URL's para determinadas recursos não utilizadas pela solução, sem as devidas configurações de autenticação e autorização.
RISCO:	1-Visualizar informações confidenciais. 2-Fazer transações indevidas.
SOLUÇÃO:	1-Recursos em desuso ou descontinuados devem estar sobre todas as regras de autenticação e autorização. 2-Deleção destes recursos da solução. 3-Implementar no mecanismo de autenticação e autorização de recursos que automaticamente não permita o acesso nas URL's da solução sem nenhum tipo de declaração(esquecidas), gerando um log de aviso em runtime. Ou seja, todas as URL's da solução devem ser mapeadas como "publica" ou "privada" com sua devida autorização. Caso aconteça de uma URL ser esquecida, esse mecanismo não permite ninguém acessar, gerando um log de aviso.

ATAQUE:	Workflow Undue
DESCRIÇÃO:	Consiste no atacante descobrir que pode furar uma seqüência de páginas pertencente a um "workflow de negócio" através de qualquer tipo e bookmark .
RISCO:	1-Fazer transações indevidas. 2-Gerar erros de runtime.
SOLUÇÃO:	Todos os casos de workflows da solução devem usar sistematicamente o padrão "Sincronized Token".

ATAQUE:	Overlapping Transaction
DESCRIÇÃO:	Consistem no usuário valido sobrepor informações de outro usuário valido indevidamente em processos transacionais que podem acontecer concorrentemente.
RISCO:	1-Perca de informações. 2-Solução ficar com informações inconsistentes.
SOLUÇÃO:	Todos os casos de processos transacionais concorrentes devem usar sistematicamente a abordagem de "Optimistic lock" em suas transações.

ATAQUE:	Client Side Authorization
DESCRIÇÃO:	Consiste na solução impedir de dar acesso a "recursos de função" usando javascript nas páginas. O atacante pode manipular e ou desabilitar o javascript.
RISCO:	Acesso aos recursos.
SOLUÇÃO:	A validação de recursos por função deve ser feitas no servidor "server-side ", fazendo com que as partes das páginas que apresentem "recursos de função" não seja enviadas para o navegador.

MANIPULAÇÃO DE PARÂMETROS:

ATAQUE:	Query String Manipulation
----------------	---------------------------

DESCRIÇÃO:	Consiste no atacante enviar ou alterar as informações de forma maliciosas dentro da URL do HTTP GET.
RISCO:	1-Transações com informações inconsistentes. 2-Furo nas regras de negócios. 3-Gerar erros de runtime.
SOLUÇÃO:	1 – Criptografe os dados do GET. 2 – Faça validações em todos os dados vindouros do GET. Mesmo aqueles pré-definidos pela solução. 3 – Não use esta abordagem, armazenando as informações na sessão do usuário.

ATAQUE:	Form Field Manipulation
DESCRIÇÃO:	Consiste no atacante enviar informações maliciosas dentro de campos HTML hidden e read-only, não esperado pela aplicação.
RISCO:	1-Transações com informações inconsistentes. 2-Furo nas regras de negócios. 3-Gerar erros de runtime.
SOLUÇÃO:	1 – Repita sistematicamente as validações de campos que utilize esta abordagem a cada interação HTTP. 2 – Não use esta abordagem, armazenando as informações na sessão do usuário.

ATAQUE:	Cookie Manipulation
DESCRIÇÃO:	Consiste no atacante enviar informações maliciosas dentro dos cookies usados pela aplicação.
RISCO:	1-Transações com informações inconsistentes. 2-Furo nas regras de negócios. 3-Gerar erros de runtime.
SOLUÇÃO:	1 – Use criptografia HMAC cookie para guardar valores importantes para aplicação. 2 – Use cookies para guardar informações que não tenham impacto na aplicação. 3 – Ou nunca use esta abordagem, armazenando as informações na sessão do usuário.

ATAQUE:	Header Manipulation
DESCRIÇÃO:	Consiste no atacante enviar ou manipular informações dentro do HTTP Header.
RISCO:	1-Transações com informações inconsistentes. 2-Furo nas regras de negócios. 3-Gerar erros de runtime.
SOLUÇÃO:	A solução não pode executar decisões transacionais baseadas nos headers HTTP por que eles podem ser facilmente falsificados.

ATAQUE:	Insecure Direct Object References
DESCRIÇÃO:	Consiste no atacante manipular informações dentro do HTTP Header enviando valores não esperados pela solução. Típico caso da solução

	carregar uma combobox com valores de chaves ID do banco de dados esperando o usuário selecionar 1 opção, sendo que um atacante pode manipular a requisição HTTP enviado um identificar inconsistente.)
RISCO:	1-Transações com informações inconsistentes. 2-Furo nas regras de negócios. 3-Gerar erros de runtime.
SOLUÇÃO:	A solução não usar a abordagem de mapeamento gerenciado na sessão do usuário, aplicando sistematicamente validações de autorização e consistências nos valores vindouros da requisição HTTP.

ATAQUE:	Unvalidated Redirects and Forwards
DESCRIÇÃO:	Consiste no atacante manipular informações dentro da requisição HTTP enviando endereços absolutos de páginas ou recursos maliciosos automaticamente redirecionados pela solução.
RISCO:	1-O atacante pode enviar endereços de recursos que contenham instalações de vírus na estação do usuário. 2- O atacante pode enviar endereços de recursos restritos da solução.
SOLUÇÃO:	Não use recursos de redirecionamentos que pegam o endereço da recurso via valores HTTP.

GERENCIAMENTO DE ERROS

ATAQUE:	Flaw Error Handling and Information Disclosure
DESCRIÇÃO:	Consiste na solução expor detalhes infraestruturais como: S.O., servidor de aplicação, nomes de servidores, tipo/filosofia de mecanismos de persistências, comandos SQL, configurações de banco de dados, versionamento e etc na ocorrência de erros de runtime.
RISCO:	A exposição destas informações pode servir como fator decisivo, habilitando um possível ataque futuro.
SOLUÇÃO:	A aplicação deve apresentar uma mensagem de erro de serviço genérico quando qualquer exception de runtime vier acontecer, usando a abordagem de "Logger" para registrar os determinados de erros técnicos e infraestruturais.

AUDITORIA E LOGGING

ATAQUE:	Desconhecer o aconteceu na aplicação
DESCRIÇÃO:	Consiste na aplicação não ter como identificar as ocorrências de ações efetuadas pelos usuários finais.
RISCO:	1-A solução não pode ser auditada em casos de ocorrências de furos de negocio. 2-Impossibilidade de responsabilizar os usuários culpados por determinadas transações indevidas, errôneas, maliciosas e ou criminosas.
SOLUÇÃO:	A aplicação deve usar algum mecanismo de log que registre (usuário, processo, data e hora, sucesso ou falha) as operações críticas mostrando o histórico de uso de cada usuário.