INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO PAULO

Campus São João da Boa Vista

Trabalho Final de Curso

4º ano – Curso Técnico em Informática

Prof. Roan Simões da Silva

APLICAR FERRAMENTAS PARA ANÁLISE DE VULNERABILIDADE DE SEGURANÇA NO PROJETO GERAÇÕES

Aluno: Rita de Cássia Simão Camelo

Prontuário: 1620576

São João da Boa Vista – SP

2019

Resumo

Considerando a implementação do Projeto Gerações, destinado as instituições de longa permanência, atendendo o gerenciamento local a partir de uma plataforma via Internet que têm como dever confidenciar os dados da referida entidade, comtempla-se a necessidade de um escaneamento do sistema como um todo, haja vista, que nem sempre é dada atenção adequada à questão da segurança dessas informações, sendo que muitas vezes a instalação é feita apenas utilizando as configurações padrão, sem realizar ajustes mais avançados. Como o número de vulnerabilidades é alto e muitas vezes possuindo fácil exploração, há um grande risco na concretização de incidentes de segurança. Diante disso, o presente trabalho tem como objetivo realizar uma análise de segurança, através do escaneamento de vulnerabilidades no sistema, fazendo uso dos temas mais populares em segurança da informação e uma ferramenta de scanner como auxílio. O objetivo dos testes é verificar quais as vulnerabilidades encontradas, analisa-las, indicar quais os pontos à serem modificados atingindo a segurança ideal e como fim, serão apresentados os pontos positivos e negativos observados ao longo de seu desenvolvimento, além de sugestões como forma de melhoria para projetos posteriores.

ÍNDICE DE ILUSTRAÇÕES

SUMÁRIO

1 Introdução

1.1 Contextualização / Motivação

São João da Boa Vista, um município do estado de São Paulo, destaca-se pelo alto índice de qualidade de vida em aspectos sociais, culturais, educacionais e ambientais. A "Cidade dos Crepúsculos Maravilhosos", slogan atribuído pela vista ao lado leste da encantadora Serra da Mantiqueira [1], ressalta-se na preocupação com sua população de forma a oferecer benefícios, em especial a parte idosa local.

Somado a isso, a localidade recebeu a colocação de melhor cidade para viver após os sessenta anos de idade segundo um estudo realizado pela Fundação Getúlio Vargas [2]. A pesquisa feita no ano de dois mil e dezessete levou em conta as baixas ocorrências de mortes por armas de fogo e o número de estabelecimentos com atendimento ambulatorial de ordem pública distribuídos em larga escala pela cidade.

A prefeitura local dedica-se a terceira idade proporcionando o bem-estar a partir de instituições de longa permanência. Nesses ambientes a população idosa carente ganha um lar e torna-se membro da grande família anciã residente. A seguir tem-se alguns exemplos de lares destinados à essa parcela da população: Centros de Convivências do Idosos (CCI) Dona Beloca - localizado no bairro do Pratinha; José Peres Castelhano - no Jardim Nova República; Morada Onofre Inocentini - no Solário da Mantiqueira; [3] o Lar São José da Sociedade São Vicente de Paulo - na Vila Santo Antônio [4] e outros.

Além disso, a distribuição de recursos que São João oferece engloba o meio de ensino, contando com um elevado número de escolas municipais e estaduais e, ainda, uma única organização de âmbito federal. A entidade localizada no Bairro Fazenda Nossa Senhora Aparecida do Jaguari, o Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP) campus São João da Boa Vista, [5] reconhecido pela sua qualidade e pelo seu legado oferece ensino público federal gratuito, em especial aos estudantes local e regional.

Dentre os cursos oferecidos pelo IFSP apresenta-se as seguintes modalidades: técnicos, tecnologias, engenharias, licenciaturas, bacharelados e pós-graduação. [6]. Evidencia-se uma maior presença de alunos e enfoque na área técnica voltado ao integrado, isso é, a junção do ensino médio a um ramo técnico profissional específico. As duas possíveis formação, Técnico Integrado em Informática e Técnico Integrado em Eletrônica, desfrutam de uma duração de quatro anos, período de aprendizagem e aperfeiçoamento. [7]

Com enfoque no primeiro curso descrito, Integrado em Informática, percebe-se que os educadores da instituição se apropriam de algumas matérias divididas de forma a

proporcionar o desenvolvimento do grupo integrante da instituição. Explorando-se a parte de informática básica, hardware, software e desenvolvimento nas linguagens java, c, php, html entre outros destaques pedagógicos.

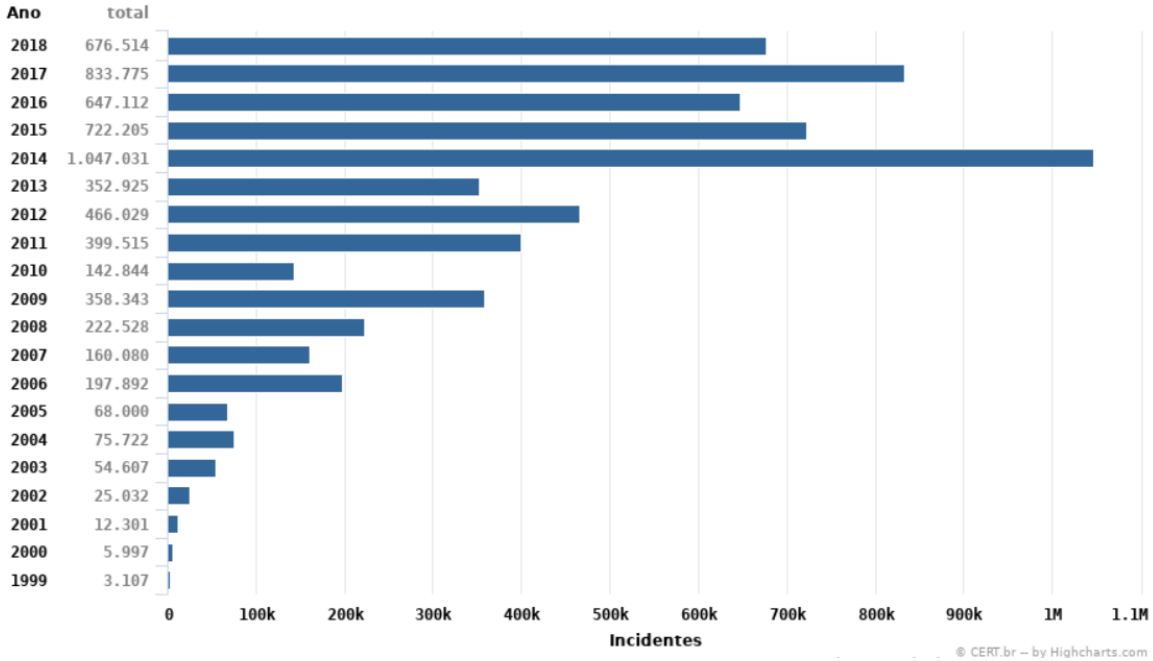
Os estudantes do IFSP contam com uma base preparatória que engloba desde o 1° até o 4° ano de ensino, momento em que se encontram capazes de realizar o projeto destinado aos últimos anos do Integrado. Colaborando com a cidade local realiza-se uma seleção para a escolha do órgão à ser beneficiado afim de atingir um propósito. No corrente ano os alunos dedicam-se à construção de uma plataforma dinâmica cujo objetivo é promover a integração entre os cuidadores e os idosos de Instituições de Longa Permanência. Dessa forma, o sistema possibilita um acompanhamento online das atividades desenvolvidas pelos instruendos, bem como seu histórico médico, alimentação, prática desportiva, e outras informações relevantes.

Tendo em vista, o quão abrangente encontra-se o projeto uma divisão em setores faz-se necessário. A cada “Módulo” atribui-se uma função específica, sendo repartidos em uma soma que juntos denominam um “Subsistema”. Os três primeiros, Usuários; Prontuário dos Idosos e Acompanhamento pelos Familiares voltam-se ao acesso externo. Enquanto isso, os quatro seguintes, Cuidado Diário dos Idosos; Prescrições Médicas e Controle de Incidentes; Nutrição, e Atividades Físicas e Recreativas, ambos destinados ao gerenciamento dos idosos. Por fim, a seção de controle administrativo, com Controle administrativo e Relatórios Especializados.

Concluída a repartição, cabe uma verificação do sistema como um todo, afim de garantir o sigilo e o não vazamento de informações. Uma vez que, a ocorrência de casos em que dados confidentes de uma determinada organização migram para o ambiente externo por falhas na segurança tem-se apresentado em larga escala.

O gráfico expõe número de incidentes no período de 1999 – 2018, sendo os casos apresentados sob autorização das instituições, que ao encontrar o erro o divulgaram. No entanto, vale ressaltar a existência de organizações que possuem falhas na segurança de suas informações e que não se encontram explícitas. [8]

Figura 1- Total de Incidentes Reportados ao CERT



540 milhões de dados de usuários do Facebook ficam expostos em servidores da Amazon

Além disso, um ocorrido em abril de 2019 resume o risco de ocorrer vazamentos em escala sigilosa. O fato resultou em 540 milhões de dados pessoais expostos de usuários do *Facebook* nos servidores da *Amazon* na nuvem, divulgado pela empresa cibersegurança UpGuard.

Segundo o Portal G1, entre os dados continham curtidas, comentários, fotos, músicas, reservas de hotéis e outros, sendo apoderado pela Cultura Criativa, instituição mexicana, segundo a UpGuard. Assim, medidas foram tomadas e a rede social trabalhou com a Amazon para retirar as bases de dados e ressaltou o compromisso de atuar com os desenvolvedores para proteger os dados dos usuários da plataforma. Segundo eles os dados já estão em segurança.

Com mais um caso explicitado, mais um capítulo nos escândalos de privacidade envolvendo redes sociais. Vale ressaltar um outro episódio do ano 2018, na qual o fundador do Facebook, Mark Zuckerberg, assumiu no Senado dos Estados Unidos a responsabilidade pelo uso inadequado de informações pela Cambridge Analytica e afirmou que a segurança é uma prioridade da empresa. [9]

Diante disso, evidencia-se a necessidade de um plano de segurança, na qual todos os elementos confidenciais relativos ao projeto sintam-se seguros quanto ao vazamento de dados.

1.2 Objetivo Geral da Pesquisa

O objetivo geral dessa obra consiste em fazer um levantamento da importância da segurança da informação, realizando uma análise de como é tratada no projeto Gerações.

1.3 Objetivos específicos

Procurando atender o objetivo supracitado realizar-se-ão testes de verificação da segurança geral do sistema, com o intuito de reconhecer possíveis erros da aplicação que possam comprometer a integridade ou a confidencialidade. Logo, inclui-se uma avaliação das soluções adotadas no projeto e se necessário propor correções à mesma.

A ferramenta utilizada com funcionalidade de scanner de segurança de aplicações web de código aberto, o OWASP Zed Attack Proxy (ZAP) é uma ferramenta gratuita e que automaticamente realiza suas devidas funções. [10]

1.4 Estrutura do Documento

Capítulo 1 – Introdução

Neste capítulo são apresentadas, uma breve descrição da cidade e sua relação com as questões de Âmbito social, a instituição de ensino IFSP e a estrutura do projeto. Além de, exemplificações fomentando a necessidade da segurança dos dados.

Capítulo 2 – Desenvolvimento

Na parte voltada ao desenvolvimento, serão analisadas as medidas de segurança inseridas no Projeto, contando com a verificação do mesmo com base em conhecimentos de segurança da informação.

Capítulo 3 - Conclusões e Recomendações

Após todo o levantamento e análise os associados desse imenso projeto poderão evidenciar a importância de tais meios a partir de fatos e ações a serem realizadas ao longo deste documento.

2 Desenvolvimento

2.1 Levantamento bibliográfico

O meio informatizado, presente na economia, carrega consigo grande troca de informações a partir da criação, armazenagem e transferência. Utilizando os recursos de sistema de informação milhares de ações são realizadas simultaneamente facilitando e promovendo o mundo dos negócios e a forma com que gozamos da vida.

A ideia de assegurar um mecanismo que proteja a comunicação envolve conceitos de termos relacionados ao tema. A seguir verificar-se-ão o que distingue dado de informação, duas palavras com conteúdo diferentes, geralmente confundidas. O que conceitua um sistema de informação aliado aos seus componentes, a abordagem da arquitetura de segurança OSI (Open Systems Interconnection), como se dá o modelo de cifra simétrica entre tantos outros pontos à serem exibidos, a fim de atingir as formas de criptografia e qual a indicada.

2.1.1 Dado em comparação com informação

Os dados se caracterizam como fatos crus, subdividindo-se em dados alfanuméricos representado por números, letras e outros caracteres; em imagem, a partir de imagens gráficas e figuras; em áudio, o que inclui som, ruído ou tom, e ainda em dados de vídeo envolvendo imagens e figuras dinâmicas.

Quando um fato se torna organizado a informação é concebida e com papel de possuidora de valor adicional, se diferencia daquilo considerado como dado.

Pode ocorrer o estabelecimento de regras e relações, organizando assim os dados em informações preciosas e de grande utilidade. Há exemplo disso, admitindo que os dados são partes de um trilho em uma certa ferrovia. Compreende-se que cada parte do trilho apresenta valor específico limitado e único, porém ao definir um vínculo entre as partes o mesmo adquire valor. Logo, ao estruturar as peças, o traçado da ferrovia começa a ganhar forma.

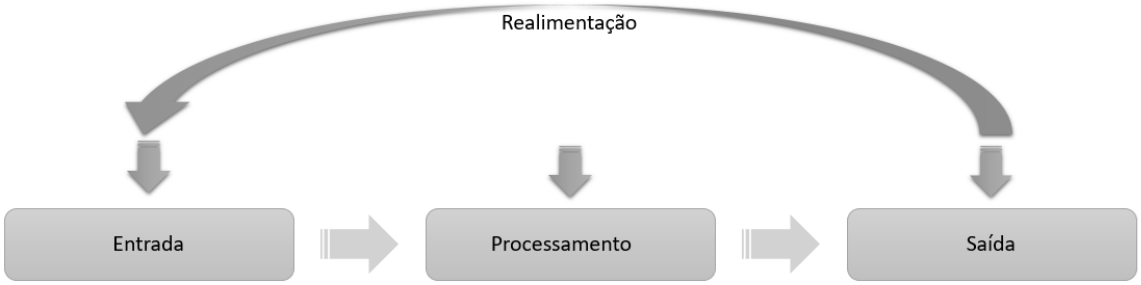
Além disso, destaca-se a opção de redefinir as relações e criar novas informações. No caso exemplificado anteriormente, o acréscimo de fragmentos ao trilho atribui-lhe valor, possibilitando a criação de um traçado mais elaborado da ferrovia.

Vale ressaltar, que o processo, nome atribuído à transformação dos dados em informação, desfruta do conhecimento para compreender os conjuntos de informações e o modo como elas podem ser utilizadas no apoio de uma tarefa em específico ou na tomada de decisão.

2.1.2 O que é um sistema de informação?

Classifica-se como sistema de informação (SI) todo conjunto de elementos ou componentes correlacionados, com a função de coletar (entrada), manipular (processo), armazenar e disseminar os dados (saída) e informações; e fornece uma relação corretiva (mecanismo de realimentação) afim de atingir um objetivo. No caso de uma organização, a realimentação encontra-se como chave para aumento de lucros ou melhora de serviço ao cliente. Segue abaixo os componentes que contribuem para o sistema de informação, de forma ilustrativa. [11]

Figura 2 - Componentes de um sistema de informação



Outrossim, convém destacar a inclusão de dois componentes adicionais: feedback e controle, ambos se apresentam como autorregulado (conhecido como sistema cibernético).

O primeiro diz respeito ao desempenho de um determinado sistema, enquanto que o outro trata da monitoração e avaliação do feedback, verificando que não haja nenhum desvio da meta à ser realizado pelo sistema. Logo após, realiza-se às correções necessária aos componentes de entrada e processamento, garantindo o alcance da produção adequada. [12]

2.1.3 A organização de segurança OSI

A arquitetura de segurança OSI volta-se aos ataques, mecanismos e serviços de segurança. Representar-se-ão a seguir a composição de cada uma dessas vertentes, organizado com precisão a fim de prover segurança.

2.1.3.1 Ataques à segurança

Ataques à segurança, podem classificar-se em passivo ou ativo. Um ataque passivo se dedica a descobrir ou ainda, aproveitar-se da informação do sistema sem que seus recursos sejam afetados. Já um ataque ativo, dispõe-se a modificar os recursos do sistema afetando sua operação.

2.1.3.2 Mecanismos de segurança

Atribui-se como um mecanismo de segurança, todo tipo de processo (ou um dispositivo introduzindo tal processo) direcionado a detectar, impedir ou permitir a recuperação de um ataque à segurança. Vale destacar alguns exemplos de mecanismos, como algoritmos de criptografia, assinaturas digitais, protocolos de autenticação entre outros meios específicos. Os mecanismos de criptografia classificam-se em reversíveis e irreversíveis, o primeiro define-se em um algoritmo que permite que os dados sejam criptografados e posteriormente decriptografado. Enquanto isso, os irreversíveis incluem algoritmos de hash e códigos de autenticação de mensagens, útil em aplicações de assinatura digital.

2.1.3.3 Serviços de segurança

No que diz respeito aos serviços de segurança, segundo à RFC 2828 (acrônimo de request for comments), que são documentos técnicos desenvolvidos e mantidos pelo IETF (Internet Enginnering Task Force) - instituição que específica os padrões que serão implementados e utilizados em toda a internet – [13] entende-se como um serviço de processamento ou comunicação fornecido por um sistema para prover um tipo específico de proteção aos recursos do sistema. Os serviços de segurança incluem políticas (ou diretrizes) de segurança e são implementados por mecanismos de segurança. Dividindo-se em cinco categorias os serviços, vale destacar cada um examinando-os.

- Autenticação – garante que a comunicação seja autentica, dividido em duas tendências. A autenticação de entidade par, utilizada em associação com uma conexão lógica para confiabilidade da identidade das entidades vinculadas. Já, quando se tratar de uma transferência sem conexão, a autenticação de origem de dados irá garantir que a origem é a apontada.
- Controle de acesso – impede qualquer tipo de uso não autorizado de um determinado recurso, ou seja, o serviço em questão controla quem tem acesso, sob quais condições o acesso torna-se liberado e as liberdades permitidas para os que acessam o mesmo recurso.

- Confidencialidade de dados – destinada a proteção dos dados transmitidos contra ataques passivos. Um serviço considerado mais amplo, protege todos os dados circulado entre dois usuários por um determinado período de tempo. Além disso, outra questão da confidencialidade é a proteção do fluxo de tráfego contra análise, o que impede a visualização do atacante sobre a origem e o destino, a frequência, o tamanho ou outras características do tráfego em um sistema de comunicação.
- Integridade de dados – assegura que os dados recebidos se encontram exatamente da mesma forma que enviados por uma entidade autorizada. Sem nenhuma modificação, inserção, exclusão ou repetição o serviço classificado com e sem recuperação está voltado a detecção ao invés da prevenção. Assim, caso uma violação for detectada, o serviço terá como função informar e alguma outra parte do software ou interferência humana irá ser necessário para recuperar-se da violação. Como opção, existem os mecanismos, já citados anteriormente, para recuperar da perda de integridade de dados.
- Irretratabilidade – tende a impedir que o emissor ou o receptor negue uma mensagem transmitida. Logo, quando enviada uma mensagem o receptor poderá provar que o emissor apontado realmente enviou tal mensagem, o mesmo vale para a recepção de mensagens, sendo verificado se o destino corresponde ao receptor alegado.
- Serviço de disponibilidade – segundo a RFC 2828 definem a disponibilidade como sendo a propriedade de um sistema ou de um sistema ser acessível e utilizável por uma entidade que seja autorizada do sistema, a partir de uma especificação de desempenho. Os ataques a disponibilidade podem ser favoráveis a contramedidas automatizadas, como autenticação e criptografia, enquanto isso outros necessitam de algum tipo de ação física para impedir ou recuperar da perda de disponibilidade dos elementos de um sistema distribuído. Vale ressaltar, que que esse serviço depende do gerenciamento e controle dos recursos do sistema e, sendo assim, depende do serviço de controle de acesso e outros serviços de segurança. [14]

2.1.4 Modelo de cifra simétrica

A criptografia simétrica (também chamada de criptografia convencional ou de chave única) disponibiliza de cinco elementos, sendo:

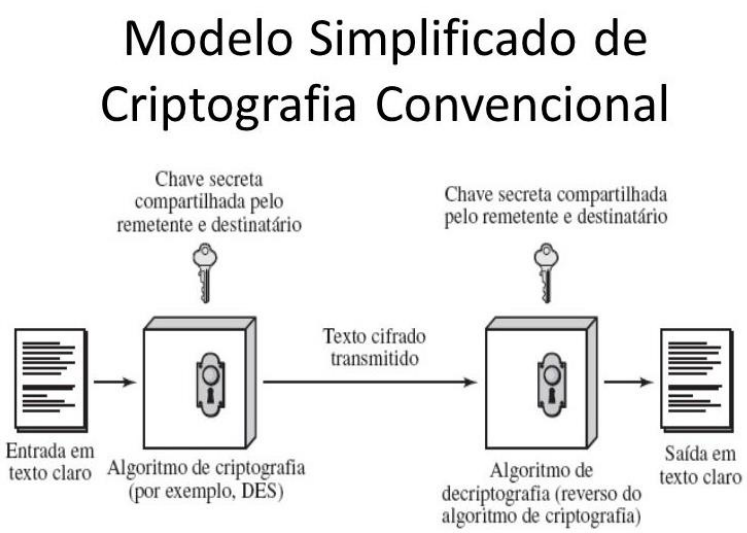
- Texto claro: destinado a entrada no algoritmo, conceitua como a mensagem ou o dado original.

- Algoritmo de criptografia: tende a realizar substituições e transformações no texto claro (mensagem original).
- Chave secreta: considerada como entrada para o algoritmo de criptografia é um valor independente do texto claro e do algoritmo. As substituições e transformações exatas dependem da chave.
- Texto cifrado: tida como mensagem embaralhada, ela é produzida como saída. Depende do texto claro e da chave secreta e de acordo com a mensagem duas chaves diferentes produzirão dois textos cifrados diferentes. Vale destacar, que ele é um fluxo de dados aleatórios e, nesse formato, ininteligível.
- Algoritmo de descriptografia: executado de modo inverso funciona basicamente como o algoritmo de criptografia, responsável por pegar o texto cifrado e a chave secreta produzindo o texto original.

Diante disso, faz-se necessário a existência de dois requisitos. Contando com algoritmo de criptografia forte, de forma que o oponente seja incapaz de descriptografar o texto cifrado ou descobrir a chave, logo, o uso da criptografia convencional tornar-se-ão seguro.

Não obstante, o emissor e receptor necessitam de uma cópia da chave secreta seguramente e precisam mantê-la protegida. Caso ocorra uma descoberta da chave e exista um conhecimento do algoritmo toda a comunicação poderá ser visualizada. Abaixo verifica-se de forma elucidativa a criptografia convencional a partir de um esquema.

Figura 3 - Modelo simplificado da criptografia convencional



2.1.4.1 Criptografia

Os sistemas criptográficos dividem-se em três dimensões independentes. A seguir verifica-se quais são elas e suas respectivas responsabilidades:

- O tipo das operações usadas para transformar texto claro em texto cifrado: admitindo-se que todos os algoritmos de criptografia são baseados em dois princípios, substituição – em que cada elemento no texto claro é mapeado em outro elemento – e transposição, em que os elementos no texto claro são reorganizados. Vale lembrar, o objetivo (requisito fundamental), sendo de extrema importância que nenhuma informação seja perdida.
- O número de chaves usadas: caso ocorra de o emissor e o receptor usufruírem da mesma chave, sistema se torna como criptografia simétrica. No entanto, se o emissor e o receptor utilizarem de chaves diferentes, trata-se de um sistema com criptografia assimétrica (de duas chaves ou chave pública).
- O modo como o texto é processado: uma cifra de bloco responsabiliza-se pela entrada de um bloco de elementos de cada vez. Sendo um de saída para cada bloco de entrada, enquanto isso, uma cifra em fluxo processa os elementos da entrada continuamente, produzindo assim um elemento por vez ao mesmo tempo prosseguindo.

2.1.4.2 Criptoanálise

Geralmente, o propósito de atacar um sistema de criptografia está voltado a recuperação da chave em uso ao invés de recuperar o texto claro de um único texto cifrado. Logo analisar-se-ão as duas técnicas gerais utilizado em um ataque de esquema de criptografia convencional.

A criptoanálise conta com a natureza do algoritmo e, por vezes, algum conhecimento das características gerais do texto claro (ou pares de amostra de texto claro e cifrado). Esse modelo de ataque volta-se as características do algoritmo para concluir um texto claro específico ou ainda a chave utilizada.

Outrossim, convém mencionar o ataque por força bruta, na qual o atacante usufrui de todas as chaves possíveis em uma parte do texto cifrado continuamente até obter uma tradução inteligível para texto claro. Sendo, na média, metade do total de chaves precisam ser experimentadas para alcançar o sucesso. [14]

2.1.5 OWASP Top 10 2017 - Os dez riscos mais críticos de segurança de aplicativos Web

Tendo como principal objetivo educar desenvolvedores, designers, arquitetos, gerentes e organizações sobre as consequências que sistemas falhos em segurança de aplicativos da Web podem vir a ocorrer a OWASP Top 10 do ano 2017, momento de atualização, fornece informações básicas técnicas para se proteger contra essas áreas problemáticas de alto risco e fornece orientações sobre como seguir pelo caminho seguro. Dessa forma, apresentar-se-ão uma breve descrição dos dez riscos de segurança de aplicativos existentes.

- A1:2017- Injection: ocorre quando existem falhas de injeção, como SQL, NoSQL, OS e LDAP, sendo que dados não confiáveis são enviados para um intérprete como parte de um comando ou consulta. Os dados hostis do invasor podem enganar os intérpretes a executar comandos não intencionais ou acessar dados sem a devida autorização.
- A2:2017-Broken Authentication: as funções de aplicativo relacionadas à autenticação e gerenciamento de sessões são frequentemente implementadas incorretamente, permitindo que os invasores comprometam senhas, chaves ou tokens de sessão ou explorem outras falhas de implementação para assumir a identidade de outros usuários temporária ou permanentemente.
- A3:2017- Sensitive Data Exposure: por existir muitos aplicativos da Web e APIs que não protegem adequadamente dados confidenciais, como dados financeiros, cuidados de saúde e PII, os invasores podem roubar ou modificar esses dados fracamente protegidos para realizar fraude de cartão, roubo de identidade entre outros crimes. Dados sensíveis podem ser comprometidos sem extra proteção, como criptografia em repouso ou em trânsito, e requer precauções especiais quando trocado com o navegador.
- A4:2017-XML External Entities (XXE): grande parte dos processadores XML antigos ou mal configurados avaliam referências de entidades externas no XML documentos. Essas entidades podem ser usadas para divulgar arquivos internos usando o manipulador de URI de arquivo, compartilhamentos de arquivos internos, verificação interna de portas, execução remota de código e ataques de negação de serviço.
- A5:2017-Broken Access Control: diz respeito às restrições sobre o que os usuários autenticados têm permissão para fazer que geralmente não são aplicadas

corretamente. Os invasores podem explorar essas falhas para acessar funcionalidades e / ou dados não autorizados, como acesso contas de outros usuários, visualizar arquivos confidenciais, modificar dados de outros usuários, alterar direitos de acesso etc.

- A6:2017-Security Misconfiguration: considerado como um dos problemas mais comuns, a configuração incorreta da segurança é resultado de insegurança das configurações padrão, configurações incompletas ou ad hoc, armazenamento em nuvem aberta, configuração incorreta Cabeçalhos HTTP e mensagens de erro detalhadas que contêm informações confidenciais.
- A7:2017- Cross-Site Scripting (XSS): as falhas existentes no XSS ocorrem sempre que um aplicativo inclui dados não confiáveis em uma nova página da Web sem validação ou escape adequados ou atualiza uma página da web existente com dados fornecidos pelo usuário usando um API do navegador que pode criar HTML ou JavaScript. O XSS permite que os invasores executem scripts no diretório navegador da vítima, que pode sequestrar sessões do usuário, desfigurar sites ou redirecionar o usuário para sites maliciosos.
- A8:2017- Insecure Deserialization: A desserialização insegura geralmente leva à execução remota de código. Mesmo que falhas de desserialização não resultem no mesmo, eles podem ser usados para realizar ataques, incluindo ataques de repetição, ataques de injeção e ataques de escalonamento de privilégios.
- A9:2017-Using Components with Known Vulnerabilities: componentes como bibliotecas, estruturas e outros módulos de software, são executados com o mesmo privilégio como o aplicativo. Se um componente vulnerável for explorado, esse ataque poderá facilitar perda de dados grave ou aquisição do servidor. Aplicativos e APIs usando componentes com conhecimento vulnerabilidades podem minar as defesas de aplicativos e permitir vários ataques e impactos.
- A10:2017- Insufficient Logging & Monitoring: registro e monitoramento insuficientes, juntamente com a integração ausente ou ineficaz com a incidente resposta, permite que os atacantes atacem mais os sistemas, mantenham a persistência, giram para mais sistemas e adulterar, extrair ou destruir dados. A maioria dos estudos de violação mostra que o tempo para detectar uma violação acabou 200 dias, normalmente detectados por partes externas, em vez de processos ou monitoramento internos. [15]

2.1.6 Modelos de Criptografia – Qual o indicado?

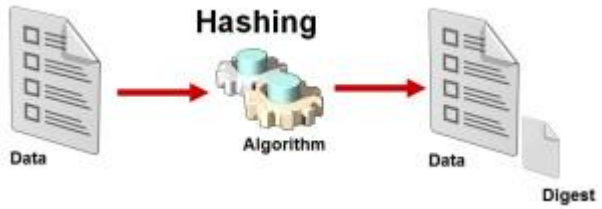
Toda rede disposta a interagir com um sistema necessita de uma segurança geral, a fim de comprometer-se de forma direta a assegurar o sigilo das informações do possuinte. Segundo a *Open Web Application Security Project* (*OWASP*) um dos ataques mais comuns às aplicações Web é a injection, que consiste em o usuário mal-intencionado inserir código malicioso para conseguir ter acesso a dados sensíveis, logo qualquer formulário Web pode servir para iniciar ataques como esse [16].

Considerado uma boa prática de segurança, o armazenamento do hash da senha no lugar da senha original, o processo pode ser unidirecional ou bidirecional. Sendo o primeiro considerado como mais forte, pois para validar a senha informada pelo usuário, torna-se necessário calcular o seu hash com o mesmo algoritmo criptográfico para só então comparar com a versão armazenada. Nesse sentido, faz-se necessário uma breve descrição do que se trata um hash seguido de suas funções.

A Criptografia é um conjunto de princípios e técnicas utilizadas para codificar uma mensagem e torná-la ininteligível para os que não tenham acesso às convenções utilizadas na codificação.

Uma função de dispersão criptográfica ou função hash criptográfica é uma função considerada praticamente impossível de inverter, isto é, de recriar o valor de entrada utilizando somente o valor de dispersão. Os dados de entrada são chamados de mensagem e o valor de dispersão é chamado de mensagem resumida ou simplesmente resumo.

Figura 4 - A Hashing



Uma função de dispersão criptográfica deve possuir as seguintes propriedades:

- Resistência à pré-imagem: dado um valor *hash* deve ser difícil encontrar qualquer mensagem m tal que h = hash(m). Este conceito está relacionado ao da função de mão única (ou função unidirecional). Funções que não possuem essa propriedade estão vulneráveis a ataques de pré-imagem.
- Resistência à segunda pré-imagem: Dada uma entrada m1 deve ser difícil encontrar outra entrada m2 tal que hash(m1) = hash(m2). Funções que não possuem essa propriedade estão vulneráveis a ataques de segunda pré-imagem.

- Resistência à colisão: Deve ser difícil encontrar duas mensagens diferentes m1 e m2 tal que hash(m1) = hash(m2). Tal par é chamado de colisão *hash* criptográfica. Essa propriedade também é conhecida como forte resistência à colisão. Ela requer um valor *hash* com pelo menos o dobro do comprimento necessário para resistência à pré-imagem.

Funções hash criptográficas possuem várias aplicações em segurança da informação como por exemplo os certificados digitais. Elas também podem ser utilizadas para indexar dados em tabelas, para detectar dados duplicados, identificar arquivos únicos e como checksum para detectar dados corrompidos.

Uma função de hash gera uma cadeia de caracteres de tamanho fixo a partir de uma sequência de qualquer tamanho. Considerado uma equação matemática que a partir de um texto cria um código chamado message digest (resumo de mensagem). O resumo criptográfico é o resultado retornado por uma função de hash que pode ser comparado a uma impressão digital, pois cada documento possui um valor único de resumo e até mesmo uma pequena alteração no documento, como a inserção de um espaço em branco, resulta em um resumo completamente diferente.

O SHA-2 classificado como um conjunto de funções hash criptográficas projetadas pela NSA (Agência de Segurança Nacional dos EUA) com o significado de secure hash algorithm (algoritmo de hash seguro) destaca-se pelo SHA-256, extremamente útil com um total em Java de 256 bits que fornece 128 bits de segurança contra-ataques de colisão.

As aplicações que usam hashes criptográficos para armazenar senhas passam à ser minimamente afetado por um ataque de colisão. Se o algoritmo de hash for fraco, pode-se fazer um ataque de pré-imagem. Nesse caso, pode-se também tentar um ataque de força bruta a partir da inversão da encriptação das senhas.

Á seguir verifica-se um exemplo em Java. O código abaixo utiliza o algoritmo SHA-256 para codificar um array de bytes em um array de tamanho fixo (16 bytes). Cada byte desse array é convertido para uma versão hexadecimal:

Figura 5- Exemplo em Java utilização do algoritmo SHA-256

```java
public String digest(String password) throws NoSuchAlgorithmException,
      UnsupportedEncodingException {
   MessageDigest algoritmo = MessageDigest.getInstance("SHA-256");
   byte digestMessage[] = algoritmo.digest(password.getBytes("UTF-8"));
   StringBuilder hexPassword = new StringBuilder();
   for (byte aByte : digestMessage) {
      hexPassword.append(String.format("%02X", 0xFF & aByte));
   }
   return hexPassword.toString();
}
```

O valor retornado pode ser armazenado no banco de dados aumentando assim a dificuldade de êxito do atacante. Vale ressaltar a possibilidade de concatenar um valor aleatório (sal) à String original antes da conversão [17].

Além disso, convém citar a existência de outras duas fontes caracterizadas menos confiável, em que se destacam sistemas muito antigos com as senhas em hash feitas por algoritmos de hash que assumem papel inseguros, são eles o MD5 (Message-Digest algorithm 5) e o SHA1 (Secure Hash Standard em inglês), fáceis de descobrir a partir de ataques de força bruta.

O MD5 caracteriza-se por um algoritmo de hash de 128 bits unidirecional desenvolvido pela RSA Data Securit, que por ser um algoritmo unidirecional (one-way), foi feito para não ser transformada novamente no texto que lhe deu origem, logo o seu método de verificação é, então, feito pela comparação das duas hash (uma da mensagem original confiável e outra da mensagem recebida). Assim, quando a senha é definida, o usuário geralmente não recebe sua senha de texto simples em troca, isso porque o serviço online não armazena senhas em texto plano e sim um valor de hash para a senha evitando o conhecimento da senha real. E por fim, se os dois hash são idênticos, então a transmissão é autêntica. [18]

Aliado a isso, segundo a plataforma de tecnologia ZDNet em maio desse ano os algoritmos de hash SHA1 tem-se tornado alvo de ataques a partir da descoberta de um "ataque de colisão de prefixo escolhido", uma versão mais prática do ataque de colisão SHA-1 realizado pela Google há dois anos atrás, na qual acadêmicos da Google produziram dois arquivos que tinham o mesmo hash SHA-1, no primeiro ataque de colisão SHA-1 do mundo, conhecido como "SHAttered".

Mas na semana passada, uma equipe de acadêmicos da França e Cingapura deram a pesquisa SHAttered um passo adiante, demonstrando um ataque de colisão SHA-1 "prefixo escolhido", em um novo estudo intitulado "From Collisions to Chosen-Prefix Collisions – Application to Full SHA-1."

Os fornecedores de navegadores há muito tempo começaram a suspender o suporte para o tráfego TLS assinado pelo SHA-1 dentro de seus produtos; no entanto, outros aplicativos ainda contam com isso.

"Os ataques contra o SHA-1 só vão melhorar", disse Scott Arciszewski, diretor de desenvolvimento e principal criptógrafo da Paragon Initiative Enterprises

Todos devem mudar para (em ordem de preferência): BLAKE2b / BLAKE2s, SHA-512/256, SHA3-256 ou SHA-384. [19]

2.2 Etapas para o desenvolvimento da pesquisa

2.2.1 Apresentação do Projeto Gerações

Inicialmente, verifica-se no Projeto Gerações dos alunos do curso técnico integrado em informática, já mencionado anteriormente, suas aplicações utilizadas ao longo de toda a estrutura, incluindo linguagem de programação, framework, banco de dados e toda infraestrutura de segurança dos dados.

Faz-se uso de linguagens de programação HTML5 – um conjunto de regras e códigos que define como os elementos da página serão exibidos - CSS - utilizada para definição de estilos, indicando o layout de documentos HTML - JavaScript - fornecendo às páginas web a possibilidade de programação, transformação e processamento de dados enviados e recebidos, interagindo com a marcação e exibição dos conteúdos da linhagem HTML e com a estilização feita com o uso do CSS – e PHP, na qual o código executado do lado do servidor passa a ser enviado para o cliente somente o resultando no formato HTML puro. [20]

Com o intuito de facilitar o desenvolvimento do projeto quando presentes componentes mais de uma vez, torna-se útil a utilização de um framework pelo desenvolvedor. A reutilização de códigos que o framework proporciona originam conjunto de bibliotecas ou componentes que são usados para criar uma base onde sua aplicação será construída. No caso em questão utilizou-se do Bootstrap contribuindo na interface a fim de melhorar a experiência visual do usuário. [21]

O sistema que abrange cerca de nove módulos, destacados em subdivisões, aponta a presença de medidas cautelosas que rege a segurança dos dados. Esses são guardados em banco de dados SQL, estruturado em colunas e linhas e logo, armazenados em tabelas. Recorrendo ao MySQL Workbench, uma marca de software que utiliza o modelo cliente-servidor sendo que o SQL informa ao servidor o que deve ser feito com o dado e assim gerenciando toda a estrutura de dados do sistema. [22]

A fim de garantir total sigilo das informações presentes no banco do Gerações, faz-se uso de algoritmos de hash SHA-256, proporcionando um meio criptográfico eficaz e assegurando-o contra ataques de colisão. Essa característica assume papel responsável no momento do login no site, por exemplo, evitando qualquer tipo de investida hacker sob o cadastro de usuários.

Diante do exposto, faz-se necessário uma verificação de possíveis vulnerabilidades no Projeto a partir de uma análise à ser realizado pelo ZAP, um projeto web scanner, admitindo

a presença ou não de vulnerabilidades e qual o nível de gravidade do problema. Somado à uma breve orientação dos pontos que podem ser corrigidos.

2.2.2 Ferramenta para Análise de Vulnerabilidade de Segurança

Foi selecionado o scanner OWASP Zed Attack Proxy (ZAP) considerado um dos softwares de segurança mais populares, atua gratuitamente baseado no top ten vulnerabilidades encontradas em aplicações web, publicado pela própria OWASP (2017). Elaborado em Java e mantido pelo OWASP, ele se encaixa no que o mercado chama de ferramentas de teste dinâmico ou blackbox.

Os testes por análise dinâmica para aplicações web funcionam basicamente em dois passos, primeiro se executa uma varredura completa na aplicação, identificando as páginas e recursos por meio de navegação nas URLs, processo conhecido pelo termo em inglês crawling. Em seguida, baseado no resultado da navegação, possíveis vulnerabilidades que o recurso possa ter são inferidas, e então realiza-se a tentativa de exploração da falha, desde a manipulação de cookies à injeção de SQL. Vale destacar, que baseado na resposta do servidor, torna-se possível identificar se a vulnerabilidade é explorável ou não.

O ZAP inclui módulos distintos voltados para cada etapa de um teste dinâmico, segue abaixo cada uma delas com suas devidas explicações:

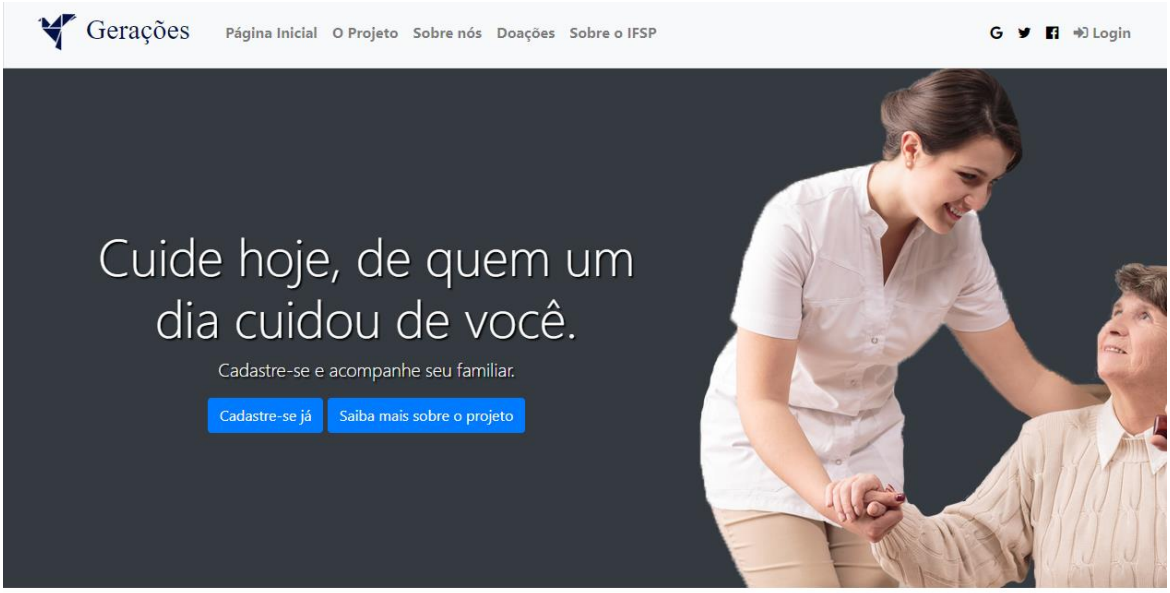
- Intercepting Proxy: permite interceptar requisições e respostas no navegador.
- Active Scanner: executa verificações de segurança automáticas contra determinada aplicação Web alvo.
- Passive Scanner: Recupera informações sobre vulnerabilidades de segurança em páginas que são baixadas usando ferramentas de varredura (crawling / spider).
- Spiders: Antes que o ZAP possa atacar uma aplicação, ele cria um mapa de navegação rastreando todas as páginas e recursos.
- REST API: Permite que o ZAP seja executado no modo headless e controlado para executar scans automatizados.

Há ainda uma versão do ZAP em contêiner Docker, com a capacidade da interface visual Java ser acessível pelo navegador, por meio WebSwing. Além disso, possui plugin para o Jenkins nativo, onde por meio da API é possível coordenar a execução automática dos testes [23]. Para utilização do scanner, foi instalado a versão ZAP 2.8.0 Standard de junho de 2019 para Windows.

2.2.3 Testagem do Projeto Gerações

Todo o processo decorre de passos extremamente relevantes que colaboram no alcance pelo sucesso. Com o acesso ao código, verifica-se a necessidade de testá-lo, no caso, utilizando o NetBeans juntamente com o banco de dados, rodado e testado a partir do aplicativo MySQL Workbench versão 8.0.18 para 64 bits.

Figura 6 - Página inicial do site



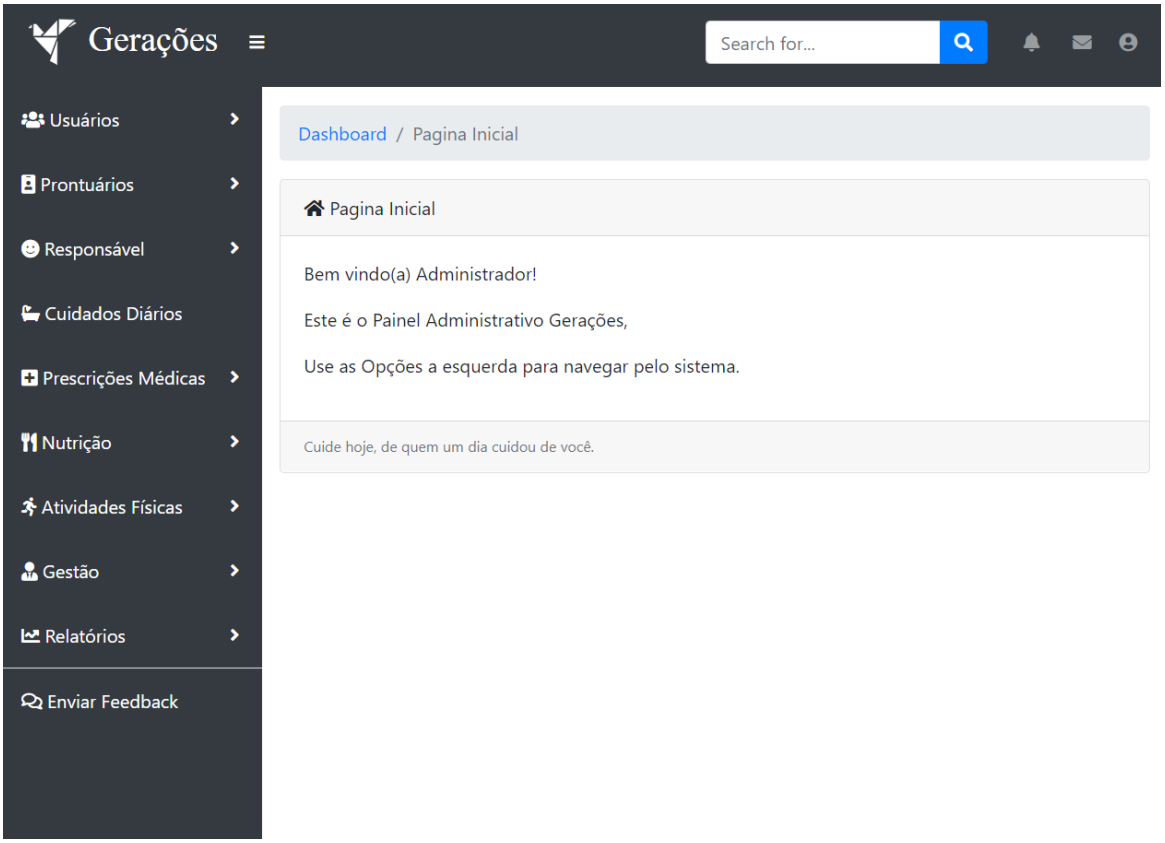
Na figura 6 torna-se visível a página inicial do site intitulado como Gerações com suas funções disponíveis, como por exemplo a parte de login.

Figura 7 - Acesso ao login com o user do administrador do Projeto

Acima na figura 7 é apresentado a parte de login com o user do administrador, filtrado o acesso, com campos de email e senha, o sistema disponibiliza o caminho para um novo

cadastro, opção de recuperação de senha e checagem de identificação, filtrando se o usuário é um robô ou não.

Figura 8 - Página de acesso restrito ao login

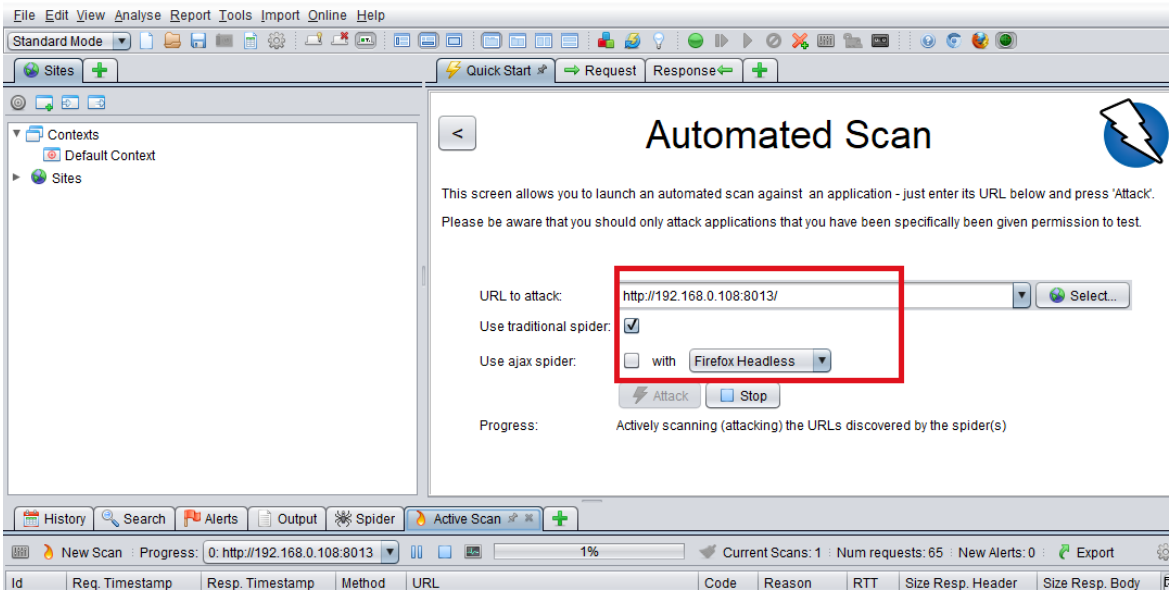


Assim, com o acesso concedido é exibido as divisões que o projeto dispõe cada uma com suas funcionalidades. Dispondo de campos específicos com os seus diferentes requisitos visível na figura 8.

2.2.4 Execução de testes

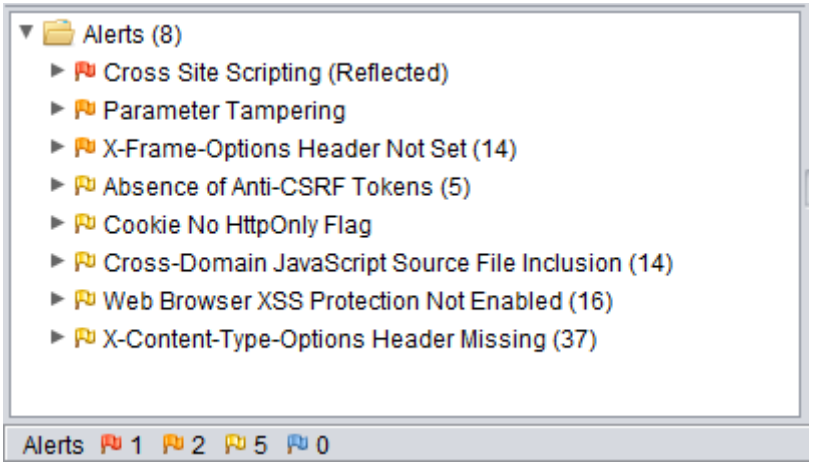
Para realização do escaneamento do Gerações, foi informado no scanner a URL escolhida do projeto. A página referente aos usuários foi escaneada pelo OWASP ZAP e o resultado foi registrado. Para a varredura do sistema o scanner foi configurado do modo padrão para executar no “Standard Mode” o que significa que as metodologias especificadas são verificadas ativamente conforme são descobertas. A porcentagem de escaneamento foi acompanhada pela barra de carregamento ativa até que a mesma estivesse em 100% indicando o fim da varredura.

Figura 9 - Configurações do Standard Mode para o início da varredura



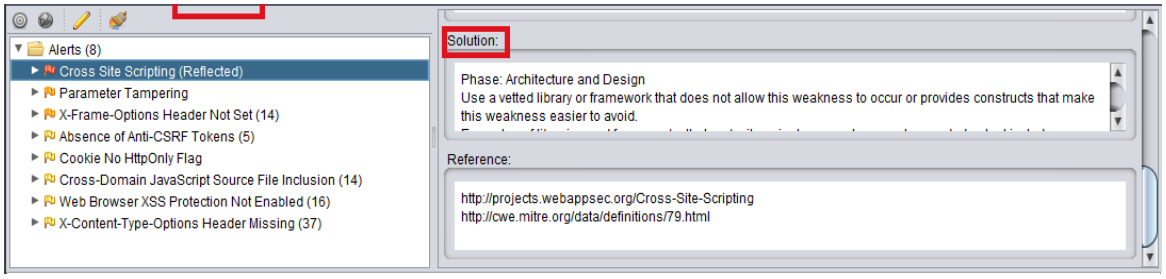
No momento do processo, a ferramenta gera um relatório das vulnerabilidades encontradas e atualiza em tempo real o número de vezes que ela foi encontrada. Sendo classificada por ordem de risco relacionado as vulnerabilidades encontradas, conforme a figura 10.

Figura 10 - Lista de vulnerabilidades encontradas durante a varredura



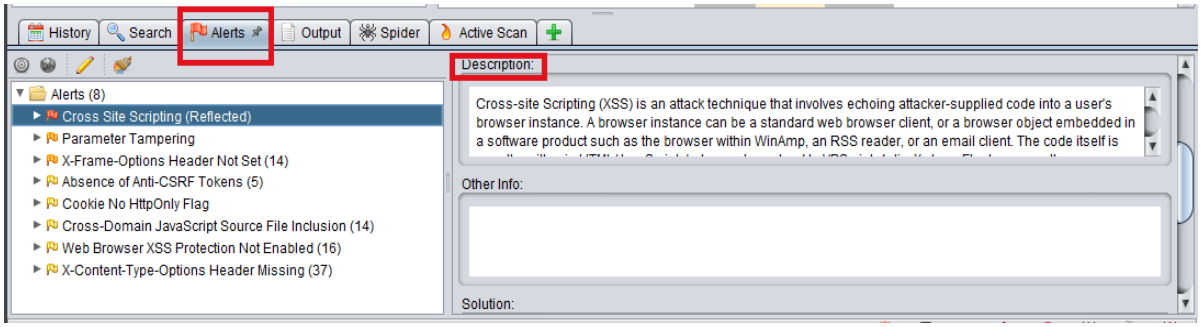
A partir da figura 10 é possível identificar que o scanner difere o nível da gravidade pela cor das bandeirinhas, dessa forma identifica-se que foi encontrada até o momento oito ameaças diferentes. Sendo uma de alto risco identificada como “Cross Site Scripting (Reflected)” representada pela bandeira de cor vermelha, duas são de nível médio que possuem a cor laranja e cinco das vulnerabilidades são de baixo risco e são classificados com a bandeira de cor amarela.

Figura 11 - Descrição do alerta de Cross Site Scripting



Além disso, ao selecionar uma das vulnerabilidades encontradas ao efetuar a varredura do sistema, o scanner, apresenta não só as definições das vulnerabilidades encontradas como as ações a serem tomadas para solucionar ou minimizar o risco, conforme a figura 11.

Figura 12 – Solução proposta do alerta de Cross Site Scripting



Dessa forma, é possível obter maiores informações relacionadas a cada uma das vulnerabilidades encontrada pelo scanner OWASP ZAP. Conforme a figura 11, o scanner forneceu detalhes do alerta de Cross Site Scripting, como a descrição da vulnerabilidade, outras informações, a solução para o problema, disposta na figura 12, e um link para que o usuário interessado possa verificar informações extras caso desejar.

2.2.5 Propostas de ações para correção

Analisando-se parte das observações representada como solução pela OWASP ZAP, verificou-se os capítulos impressos no Projeto Gerações, ambos possuindo um nível de gravidade e um direcionamento à ser melhorado.

O classificado como Cross Site Scripting, considerado o nível mais alto de identificação de vulnerabilidade, alerta mecanismos estruturados que imponham automaticamente a separação entre dados e código, se possível. Esses mecanismos podem fornecer as cotações, codificação e validação relevantes de forma automática, poupando o desenvolvedor de fornecer esse recurso em todos os pontos em que a saída é gerada.

No momento da implementação, recomenda-se que para cada página da web gerada, use e especifique uma codificação de caracteres como ISO-8859-1 ou UTF-8. Quando não

especificada, o navegador da web pode escolher uma codificação diferente, adivinhando qual está realmente sendo usada pela página da internet. Isso pode fazer com que o navegador trate determinadas sequências como especiais, expondo o cliente a ataques sutis do Cross Site Scripting. Para ajudar a mitigar esses ataques contra o cookie de sessão do usuário, defina o cookie de sessão como HttpOnly. Em navegadores que suportam o recurso HttpOnly (como versões mais recentes do Internet Explorer e Firefox), esse atributo pode impedir que o cookie de sessão do usuário seja acessível a scripts maliciosos do lado do cliente que usam document.cookie. Esta não é uma solução completa, pois o HttpOnly não é suportado por todos os navegadores. Mais importante, o XMLHTTPRequest e outras poderosas tecnologias de navegador fornecem acesso de leitura aos cabeçalhos HTTP, incluindo o cabeçalho Set-Cookie no qual o sinalizador HttpOnly está definido.

As partes encontradas e classificadas como X-Frame-Options Header Not Set, de nível médio, encaminha que os navegadores da Web mais modernos que oferecem suporte ao cabeçalho HTTP X-Frame-Optionsl, recomenda que esteja definido em todas as páginas da web retornadas pelo site o uso do sameorigin, caso contrário, deve-se utilizar do deny. Vale destacar, o valor do Allow-from, que permite que sites específicos enquadrem a página da Web em navegadores suportados.

Parameter Tampering, propõe como solução a identificação da causa do erro e que o mesmo seja corrigido. Sendo necessária uma busca na entrada do lado do cliente e a imposição de uma verificação rigorosa no lado do servidor, utilizando-se de um status de erro 500 genérica para falhas internas do servidor, apresentando a dificuldade de processamento do mesmo.

3 Conclusões e Recomendações

Atendendo às necessidades sociais do município de São João da Boa Vista, princípio fundamental do Projeto Gerações intitulado por 58 integrantes do Instituto Federal, cujo o objetivo é a elaboração de um sistema que controle o cotidiano dos idosos residentes das Instituições de Longa Permanência. Buscando desde o início melhorar a qualidade de vida dos idosos e facilitar nos registros das casas de repouso, que necessita de inúmeros cuidados.

Dessa forma, a atual pesquisa desenvolveu o papel de importância da segurança da informação, zelando pelos dados do ambiente institucional de idosos, contando com uma varredura do sistema. Projetou-se ao longo dessa obra, dividida em etapas para o preparo e alcance dos objetivos específicos, a exposição dos conceitos de termos da preservação de dados seguido por teste de vulnerabilidade.

O primeiro passo, voltou-se a exibição de significados, pautada em tópicos fragmentou-se em: distinção entre dado e informação, conceituação de um sistema de informação aliado aos seus componentes, abordagem da arquitetura de segurança OSI, o que se atribui ao modelo de cifra simétrica, os dez riscos mais críticos de segurança de aplicativos Web e os modelos existentes de criptografia.

Aliado a isso, a segunda etapa desse estudo apoiou-se nos passos necessários para o desenvolvimento do mesmo. Prosseguindo com a apresentação do Projeto, levou-se em conta o tipo de linguagem de programação, o framework utilizado, o banco de dados e toda infraestrutura de segurança dos dados.

A fim de realizar os testes, ocorreu a escolha de uma ferramenta scanner para garantir que fosse feita a varredura do sistema. O OWASP ZAP, como instrumento trabalhou em cima do código do Gerações elaborado por desenvolvedores da equipe e notificou as partes à serem aperfeiçoada com o intuito de sustentar total sigilo das informações circundadas no sistema.

Para que não ocorresse falhas no processo, foi selecionado um espaço para testagem do sistema, rodando-o via web e conectado ao banco. Após isso, não apresentado erro algum seguiu-se para a parte de testes apresentando a URL do projeto a ser examinada.

Com o propósito de identificar falhas no sistema, detectou-se em nível de gravidade, modelo padrão da ferramenta, as vulnerabilidades presentes até o momento no Projeto, quantificada e dividida em números foram exibidas na execução dos testes.

Outrossim, concretizou-se um relatório com a exibição de todos os detalhes referentes a verificação de varreduras, dispondo de termos técnicos e com a finalidade de aprimorar o

sistema Gerações, no caso foram selecionadas as propostas de interesse e apresentadas nesse trabalho.

Percebendo-se a presença de falhas e atrasos durante o decorrer do projeto, algumas atividades mal executadas ou incompletas, como é o caso de alguns módulos no quesito das Iterações, tendo em vista recorrentes erros no Banco de Dados ou até mesmo por falta de responsabilidade do desenvolvedor levando-os a ultrapassar o prazo de entrega, recomenda-se para projetos futuros uma alteração no cronograma geral. Priorizando algumas atividades que levam mais tempo que outras, deixando um período maior para as mesmas que serão realizadas e contando com o adiantamento das tarefas base.

Recomenda-se estar em status como, adiantado no trabalho e em caso de conclusão antecipada, torna-se válido o início da próxima. Em conseguinte, é interessante ressaltar a interação entre os integrantes para que todos estejam a par de todo o desenvolvimento dentro do módulo, as dificuldades e dúvidas - caso exista - e qual o andamento do projeto, com o intuito de ter a comunicação presente e tornar possível que o grupo ajude de forma facilitada um ao outro quando necessitar.

Assim, dadas as etapas de desenvolvimento a partir de ações factuais, utilizando-se de materiais teóricos, adquiridos ao longo do curso Técnico Integrado em Informática, e baseado no projeto Gerações, inaugurado pelos alunos do último ano do ensino médio do mesmo ensino, atingiu-se o objetivo central da obra.

4 Referências Bibliográficas

[1] **Guia do Turismo Brasil**. São João da Boa Vista - SP. Disponível em: <https://www.guiadoturismobrasil.com/cidade/SP/186/sao-joao-da-boa-vista>. Acesso em: 14 de ago 2019.

[2] LAURA PRADO, ANA. As 40 melhores pequenas cidades para envelhecer, 2017. Disponível em:<https://exame.abril.com.br/brasil/as-40-melhores-pequenas-cidades-para-envelhecer/>. Acesso em: 14 de ago 2019.

[3] **Fala São João**. Reinaugurado o Centro de Integração do Idoso, 2018. Disponível em:<https://falasaojoao.com/reinaugurado-o-centro-de-integracao-do-idoso/>. Acesso em: 16 de ago 2019.

[4] **App Local Empresas**. Lar São José da Sociedade São Vicente de Paulo em São João da Boa Vista, SP. Disponível em:<https://applocal.com.br/empresa/lar-sao-jose-da-sociedade-sao-vicente-de-paulo/sao-joao-da-boa-vista/sp/8076570 >. Acesso em: 16 de ago 2019.

[5] Como chegar, 2017. Disponível em:<https://www.sbv.ifsp.edu.br/como-chegar>. Acesso em: 16 de ago 2019.

[6] Cursos, 2018. Disponível em:<https://www.sbv.ifsp.edu.br/cursos>. Acesso em: 20 de ago 2019.

[7] Cursos técnicos, 2018. Disponível em: <https://www.sbv.ifsp.edu.br/cursos?id=113>. Acesso em: 20 de ago 2019.

[8] **Cert.br – Estatística**. Estatísticas dos Incidentes Reportados ao CERT.br, 2018. Disponível em: <https://www.cert.br/stats/incidentes/>. Acesso em:03 de set 2019.

[9] **G1**. 540 milhões de dados de usuários do Facebook ficam expostos em servidores da Amazon, 2019. Disponível em: <https://g1.globo.com/economia/tecnologia/noticia/2019/04/04/dados-de-540-milhoes-de-usuarios-do-facebook-ficam-expostos-em-servidor.ghtml>. Acesso em: 04 de set 2019.

[10] **Category: OWASP Top Ten Project**. OWASP Top 10 Most Critical Web Application Security Risks, 2019. Disponível em: <https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project>. Acesso em: 05 de set 2019.

[11] STAIR, R.; REYNOLDS, G. **Princípios de Sistemas de Informação:** Tradução da 9ª edição norte-americana. São Paulo: Cengage Learning, 2011

[12] O'BRIEN JAMES, A. **Sistemas de informação:** e as decisões gerenciais na era da internet. São Paulo: Saraiva, 2010

[13] **Canaltech**. O que é um RFC. Disponível em: < https://canaltech.com.br/internet/O-que-e-um-RFC/ > Acesso em: 25 de set 2019

[14] STALLINGS, William. **Criptografia e segurança de redes:** princípios e práticas. São Paulo: Pearson Prentice Hall, 2008

[15] **OWASP**. OWASP Top 10 – 2017, 2017. Disponível em: < https://www.owasp.org/images/7/72/OWASP_Top_10-2017_%28en%29.pdf.pdf >. Acesso em: 10 de out 2019.

[16] **Jaguaribe Tech.** Validação de formulários com HTML5, 2018. Disponível em: < https://medium.com/jaguaribetech/valida%C3%A7%C3%A3o-de-formul%C3%A1rios-com-html5-d1d1aa89bc77 >. Acesso em: 12 de out 2019.

[17] **Atitude Reflexiva.** Criptografia de Senha em Java, 2016. Disponível em: < https://atitudereflexiva.wordpress.com/tag/sha-256/ >. Acesso em: 14 de out 2019.

[18] **Brasil na Web.** Criptografia PHP – md5, sha1, base64, 2013. Disponível em: < https://www.brasilnaweb.com.br/blog/criptografia-no-php-md5-sha1-base64/ >. Acesso em: 29 de out 2019.

[19] **SegInfo.** Ataques de colisão SHA-1 agora são realmente práticos e criptografia se torna um perigo iminente, 2019. Disponível em: < https://seginfo.com.br/2019/05/20/ataques-de-colisao-sha-1-agora-sao-realmente-praticos-e-criptografia-se-torna-um-perigo-iminente/ >. Acesso em: 29 de out 2019.

[20] **Portal Web Designer.** Linguagens para Programação Web, 2019. Disponível em: < http://portalwebdesigner.com/programacao/ >. Acesso em: 30 de out 2019.

[21] **GETTING STARTED.** O que é um Framework?. Disponível em: < https://tableless.github.io/iniciantes/manual/js/o-que-framework.html >. Acesso em: 30 de out 2019.

[22] **Hostinger.** O que é mysql? Guia para iniciantes. Disponível em: < https://www.hostinger.com.br/tutoriais/o-que-e-mysql/ >. Acesso em: 30 de out 2019.

[23] **Linkedin.** Executando testes dinâmicos com ferramentas OWASP, 2018. Disponível em: < https://www.linkedin.com/pulse/executando-testes-din%C3%A2micos-com-ferramentas-owasp-bruno-dantas >. Acesso em: 31 de out 2019.