## Introdução

No dia a dia, internautas navegam por diversos sites, tanto de comércio eletrônico como redes sociais e plataformas de serviços como Youtube e Spotify, e sequer percebem a quantidade de informações pessoais coletadas pelos sites e aplicações com os quais interagem. Quando digitam um comentário em resposta a um post, quando acessam (*logam)* com suas contas (e-mail e senha), quando realizam uma compra ou mesmo quando escolhem a opção de cor de fundo de uma tela, todos esses dados são coletados, armazenados e comparados com dados já registrados no banco de dados do servidor. Porém, muitos desconhecem como esses bancos armazenam informações cruciais de sua vida, como produtos adquiridos, senhas e números de cartões de crédito. Desde a década de 90 a internet entrou em um crescente estado de popularização e, com o aumento do número de usuários, o número de sites e serviços disponíveis também aumentou, necessitando também que o desenvolvimento das aplicações Web acompanhe esse crescimento. Atualmente existe uma grande demanda dessas aplicações e elas estão cada vez mais complexas e o prazo de entrega exigido em seu desenvolvimento é cada vez mais curto. Isso faz com que programadores foquem nas funcionalidades principais e negligenciem aspectos de segurança dessas aplicações. Com isso, o número de ataques que exploram vulnerabilidades de segurança vem aumentando, e, então, estudos sobre a segurança de códigos se tornaram mais frequentes e necessários.

## Material Elaborado

No início da nossa pesquisa científica coletamos dados e informações sobre segurança da informação, analisamos estudos sobre vulnerabilidades em aplicações Web, verificamos possíveis métodos de análise e prevenção de ataques a essas aplicações e pesquisamos diferentes tipos de ataques existentes. Para isso, recorremos a artigos científicos em bases virtuais que são verificadas e

reconhecidas no meio acadêmico, tais como IEEE Xplore digital library[1], ACM Digital Library[2] e Science Direct[3].

A partir desse estudo destacamos as vulnerabilidades de SQL Injection e XSS, que fazem parte do grupo de 10 vulnerabilidades mais críticas encontradas aplicações Web pela OWASP[4].

Em uma vulnerabilidade de SQL Injection, o usuário mal-intencionado consegue acesso ao banco de dados do sistema por meio da inserção de cláusulas SQL em campos de entrada do usuário, podendo realizar consultas, alterações ou até excluir dados. Já em uma vulnerabilidade de XSS, o invasor explora a inserção de scripts maliciosos em campos de busca, podendo modificar elementos de interface do site ou, em casos mais graves, sequestrar sessões de usuário com permissão de administrador.

Também, ao longo da pesquisa, desenvolvemos:

- **Mapa mental:**

Desenvolvemos um mapa mental, ou seja, um diagrama apresentando conceitos, ferramentas e exemplos relacionados à vulnerabilidade do tipo SQL Injection. O mapa foi baseado em artigos científicos e, desse modo, organizamos e classificamos as várias informações pesquisadas com o objetivo de facilitar o entendimento dessa vulnerabilidade de segurança e fornecer diretrizes para auxiliar programadores a desenvolverem aplicações sem esse tipo de vulnerabilidade.

- **Demonstração SQL Injection:**

Com base em nossa pesquisa sobre SQL Injection, desenvolvemos uma aplicação Web que simula um sistema de cadastro e login de usuários. Essa aplicação é, propositalmente, vulnerável para permitir verificarmos como ocorre um ataque de SQL Injection. Também implementamos um outro caso de uso, onde aplicamos técnicas seguras de codificação para que a aplicação seja protegida contra a vulnerabilidade de SQL Injection. O objetivo dessas aplicações é demonstrar o problema da falta de segurança e possíveis soluções para esse problema, auxiliando programadores a construírem aplicações mais seguras.

[1] https://ieeexplore.ieee.org/Xplore/home.jsp
[2] https://dl.acm.org
[3] https://www.sciencedirect.com
[4] https://owasp.org/www-project-top-ten/

- **Demonstração XSS:**

Outra vulnerabilidade abordada no nosso projeto foi do tipo Cross-site scripting (XSS). Também desenvolvemos uma aplicação vulnerável, com campos de entrada do usuário, buscando testar formas possíveis de ataques por meio da inserção de scripts maliciosos, onde podemos validar diferentes casos de ocorrência da falha XSS. Similarmente ao estudo de caso sobre SQL Injection, nós também desenvolvemos uma aplicação segura, implementando técnicas de sanitização de dados para proteger a aplicação contra ataques do tipo XSS.

- **Ferramenta automatizada de testes:**

Dada a complexidade de uso das ferramentas de análise de vulnerabilidades disponíveis no mercado ou na comunidade acadêmica, desenvolvemos um protótipo para identificar aplicações vulneráveis a ataques SQL Injection. O objetivo é que a ferramenta seja simples e fácil de utilizar para fins de estudo. Ela foi desenvolvida na linguagem de programação PHP, que é utilizada para criar funcionalidades *back-end* de aplicações Web, como conexões com o banco de dados e manipulação de entradas do usuário. A ferramenta simula um ataque de SQL Injection na página que o usuário deseja verificar a segurança e verifica se a página é vulnerável ou se possui proteções contra ataques do tipo SQL Injection.

- **Site informativo:**

Além de ter como objetivo o aprendizado e primeiro contato com o meio acadêmico, nossa pesquisa teve como finalidade a divulgação de informações úteis para desenvolvedores que desejam aprimorar seus conhecimentos sobre segurança e para auxiliar na conscientização da população sobre a importância da segurança de suas informações pessoais ao utilizar serviços e aplicações Web. Por esse motivo, após concluir as demonstrações de vulnerabilidades e a ferramenta de testes, partimos para o desenvolvimento de um website que tem como intuito concentrar todas as informações e todo o conhecimento que adquirimos ao longo dos doze meses de estudo, além de diversos dados sobre ataques reais, que comprovam a importância que o estudo desse ramo tem.

## Conclusão

Qualquer tipo de vulnerabilidade de segurança, seja do tipo *SQL Injection*, *Cross-site Scripting* ou outros, representa um grande risco de prejuízos tanto para a empresa que detém o *software* vulnerável quanto para o seu cliente e usuário.

É por esse motivo que defendemos a necessidade de conscientizar a população em geral sobre a importância da segurança de suas informações pessoais e preparar os desenvolvedores, que são as pessoas que constroem as aplicações, para combaterem essas vulnerabilidades, utilizando técnicas de programação segura. Por isso, decidimos nos empenhar em estudar sobre vulnerabilidades conhecidas no meio de aplicações Web e divulgá-las, bem como maneiras de preveni-las ou mitigá-las, para demonstrar a importância da segurança de um sistema.