**Segurança na web – conceitos, práticas recomendadas, ferramentas e tendências**

**Introdução:**

Um dos temas mais importantes quando estamos pensando na criação de um site, ou até mesmo entrando em um como usuários, é a segurança. As importâncias de um sistema web seguro vai muito além da proteção de dados sensíveis, pois sabemos que a internet não é um lugar seguro. Trazendo a confiança dos usuários, cuidando das reputações das organizações e garantindo uma estabilidade do ambiente digital.

Tendo o objetivo de proteger sites de acessos, uso, destruição ou interrupção não autorizados, a segurança requer reforços em todo o site: dentro da própria aplicação web, no servidor, nas politicas de criação e renovação de senhas. Contando com os conceitos de autenticação, autorização, confidencialidade, integridade e disponibilidade.



Frameworks são ótimos aliados na segurança, por possuírem em sua grande parte, mecanismos de defesas fortes e bem programados contra ataques mais comuns.

**Ataques mais comuns e como se proteger:**

As aplicações webs podem sofrer vários tipos de ataques, dependendo de quem está atacando, quem está sendo atacado, os objetos dos ataques e das falhas na segurança do aplicativo. Entre os ataques mais comuns, podemos destacar:

* Injeção de código (Cross-Site Scripting) – XSS é um modo de ataque é achar as falhas dos sites e injetando (por isso o nome) códigos ou scripts maliciosos, visando o acesso de bancos de dados. Quando o site, com o código malicioso, é acessado, quem o atacou consegue, por meios de cookies disparados, acessar informações como se fosse um usuário.

Os processos para assegurar disso não acontecer, são a parametrização das consultas do SQL e o uso de sanitização de entrada, fazendo a limpeza das entradas que foram feitas.

* Injeção de SQL (SQLi) – seria algo parecido com o XSS, porém visando especificando as vulnerabilidades dos bancos de dados, obtendo acessos não autorizados para poder excluir, incluir ou alterar qualquer informação desse banco.

Para evitar esses tipos de ataques, devesse garantir que os dados dos usuários que passam para a consulta SQL não tenha a liberdade de alterar a natureza da consulta.

* [Falsificação de solicitações entre sites (CSRF)](https://www.cloudflare.com/learning/security/threats/cross-site-request-forgery/) – seria mais um ataque especificamente ao usuário, e não a aplicação em si, onde é induzido a vitima a fazer uma solicitação que utiliza a autorização e a autenticação. Tirando essas informações, é possível acessar os sites como se fosse o usuário e podendo alterar, excluir ou extrair dados. Normalmente visado em usuários com permissões e privilégios.

Práticas para evitar isso seriam os usos de domínios padrão, onde um usuário saiba se está sempre acessando a pagina correta, mesmo entrando em vários menus no site e também a utilização de token anti CSRF.



**Conclusão:**

Com o aumento do uso da internet, vemos também o aumento dos perigos dentro dela. E nos, como programadores, devemos sempre estar atentos, adotando melhores práticas de segurança, entendo os princípios básicos da segurança da informação e estudando cada vez mais em como deixar o ambiente digital mais protegido para os usuários.

**Bibliografia:**

‌FABRICIO GALDINO. **Implementando segurança na programação Web**. Disponível em: <https://www.devmedia.com.br/implementando-seguranca-na-programacao-web/34375>. Acesso em: 15 ago. 2024.

MARCELA. **Segurança da informação para desenvolvedores: protegendo seu código e seus usuários**. Disponível em: <https://awari.com.br/seguranca-da-informacao-para-desenvolvedores-protegendo-seu-codigo-e-seus-usuarios/>. Acesso em: 15 ago. 2024.

**Segurança Web: 5 práticas para um desenvolvimento seguro**. Disponível em: <https://www.devdofuturo.com/seguranca-web/#Fundamentos\_da\_seguranca\_web>. Acesso em: 15 ago. 2024.

‌**Segurança em aplicação web - Aprendendo desenvolvimento web | MDN**. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Learn/Server-side/First\_steps/Website\_security>.

‌