Aluno: Eduardo Gregório Wagner. CURSO: TURNO: Noturno TURMA: 1
DISCIPLINA: ***Desenvolvimento Web/Mobile III***
PROFESSOR: Adrian Ferreira Ramos DATA: **10/05/2024**

2° Bimestre - Prática 1

1 - Confidencialidade: A confidencialidade se refere à garantia de que apenas usuários autorizados possam ter acesso a um software ou site. Formas de prevenir invasões, por exemplo: o uso de criptografia, controle de acesso e autenticação. Tem como principal objetivo proteger as informações sensíveis do usuário contra o acesso por pessoas não autorizadas.

2 - Integridade: É a garantia de que os dados sejam mantidos completos e inalterados, seja por você ou por softwares de terceiros. Isso envolve o armazenamento correto dos dados para evitar qualquer tipo de adulteração ou fraude. A integridade assegura que os dados sejam confiáveis e precisos durante todo o seu ciclo de vida.

3 - Disponibilidade: Deixar disponível o sistema operacional para uso quando for necessário.

A - SQL Injection (SQLi): Sql Injection tem como principal objetivo roubar informações de um banco de dados, pode ser por meio de url, inputs, cookies ou requisições HTTP.

B - Cross-Site Scripting (XSS): É um tipo de ataque que pode permitir que um invasor insira scripts maliciosos em páginas da web às quais o usuário tenha acesso. Esses scripts podem realizar ações que não são do interesse do usuário, como roubar suas informações pessoais. Esses ataques são direcionados principalmente a solicitações que resultam em alterações de estado no site ou no software.

C - Cross-Site Request Forgery (CSRF): É um ataque que força o usuário final a executar ações na qual ele não deseja. Esses ataques visam especialmente solicitações de mudança de estado.

D - Man-in-the-Middle (MITM): É um ataque em que o invasor tenta interceptar a comunicação entre dois hosts e com isso acaba roubando informações.

E - Broken Authentication: Acontece quando existem falhas no processo de autenticação. Se os software ou sites não limitarem a quantidade de vezes em que o usuário possa tentar acessar sua conta e não bloquear essa mesma pessoa pode acabar tendo um ataque Brute Force.

F - Security Misconfiguration: É quando você não configurou direito um aplicativo por exemplo, e com isso pode haver vulnerabilidades no sistema.

4 - Princípios de Segurança em TypeScript:

A - Validação de entrada de dados: Garante que todas os dados sejam válidos e também evita de ocorrer ataques como SQL Injection e Cross-Site Scripting (XSS)

B - Prevenção de vazamentos de informações: Garante que informações pessoais como senha, email, cartão cpf entre outros tipos de dados pessoais não sejam vazados. Além disso, utilizam o uso de ferramentas de prevenção de bibliotecas de criptografia confiáveis.

C - Controle de acesso e autenticação seguros: Acaba utilizando uma autenticação mais segura através de Token (JWT) e até mesmo autenticação de dois fatores (2FA).

D - Uso seguro de APIs e serviços externos: Com uma API segura garante que os dados recebidos de uma Api externa sejam validados e com isso acaba evitando ataques como de SQL Injection e Cross-Site Scripting (XSS)

E - Gerenciamento de dependências seguro: Envolve em manter suas dependências sempre atualizadas para garantir que ela fique na versão sempre mais recente.

F - Auditoria de código e revisões de seguranças. Envolve em realizar algumas auditorias do seu código para buscar corrigir bugs e ou até mesmo vulnerabilidades dentro do seu código. É também bem importante utilizar ferramentas de análise estática de código como scanners de vulnerabilidades.

G - Tratamento seguro de exceções: Evitar, por exemplo, expor informações como mensagens de erro e logs. E implementar um tratamento de execução adequado para lidar com erros