Questão 2: Teoria de Criptografia

A criptografia simétrica e a assimétrica são dois métodos fundamentais de proteção de dados, cada um com suas particularidades e usos específicos. A criptografia simétrica utiliza a mesma chave para cifrar e decifrar informações, sendo rápida e eficiente para grandes volumes de dados, como na criptografia de arquivos ou na proteção de comunicações em redes privadas (VPNs. No entanto, sua principal limitação é a necessidade de compartilhar a chave de forma segura, o que pode ser vulnerável a interceptações.

Já a criptografia assimétrica usa um par de chaves: uma pública (para cifrar) e uma privada (para decifrar). Ela resolve o problema da troca segura de chaves, sendo ideal para comunicações iniciais em sistemas como HTTPS, para assinaturas digitais (como em certificados de autenticação). No entanto, é mais lenta e não é prática para criptografar grandes quantidades de dados diretamente.

Em resumo, a criptografia simétrica é melhor para proteger dados em trânsito ou armazenados quando a troca de chaves é segura, enquanto a assimétrica é essencial para estabelecer comunicações seguras, autenticar identidades e permitir trocas de chaves sem risco.

Questão 3: Mitigação de Ataques

1. Injeção SQL (SQL Injection)
Ocorre quando um invasor insere comandos SQL maliciosos em campos de entrada, como formulários, permitindo acessar ou manipular o banco de dados. Para evitar isso, use consultas parametrizadas para separar dados de comandos SQL.

2. Cross-Site Scripting (XSS)
Acontece quando um atacante injeta scripts maliciosos em páginas web, que são executados no navegador das vítimas. Para prevenir, escape os dados antes de exibi-los no HTML. Utilize Content Security Policy (CSP) para restringir fontes de scripts confiáveis e valide/sanitize entradas de usuário.

3. Quebra de Autenticação
Ocorre quando falhas em sistemas de login permitem que invasores roubem credenciais ou assumam contas. Mitigue isso usando autenticação multifator (MFA) para adicionar uma camada extra de segurança. Armazene senhas com algoritmos de hash fortes (como bcrypt ou Argon2) e implemente proteção contra força bruta (limite de tentativas de login e CAPTCHAs).