




O que é:



Uma injeção de SQL, às vezes abreviada como SQLi, é um tipo de vulnerabilidade em que um invasor usa uma parte do código SQL (Structured Query Language) para manipular um banco de dados, injeta ou infecta um aplicativo da web com código malicioso e obter acesso a informações potencialmente valiosas, destruição de dados sensíveis ou outros comportamentos manipuladores. Esse é um dos tipos de ataque mais comuns e perigosos porque pode ser usado contra qualquer aplicação de Web ou site que utilize um banco de dados SQL.

Como é executado:



SQL Injection (SQL)

SQL é um comando usado para enviar consultas a um banco de dados, especialmente para acessar, recuperar, salvar ou excluir dados do banco de dados.

Injeção de código

Injeção um ataque de Durante de código, um invasor garante que está familiarizado com a linguagem de programação ou o código do aplicativo de sua rede.

Injeção de Comando

Os aplicativos da Web às vezes são configurados para chamar comandos do sistema em seus servidores da Web em operação.

A falha em restringir ou validar a entrada do usuário pode levar a um ataque de injeção.

Injeção CCS

Uma injeção CCS ocorre quando um invasor detectar e explorar lacunas no processamento de ChangeCipherSpec em algumas versões do OpenSSL.

Injeção de cabeçalho de host




Os servidores que hospedam muitos sites precisam de um cabeçalho de host. Quando uma solicitação HTTP é feita, é o valor do cabeçalho do host que determina qual aplicativo da web responde a ela.

Um cibercriminoso pode manipular o cabeçalho do host para iniciar uma redefinição de senha. Em alguns casos, injetar o cabeçalho do host pode causar envenenamento do cache da web.

Como pode ser evitado:

Usar instruções preparadas (com consultas parametrizadas)



Este método de sanitizar entradas de banco de dados envolve forçar os desenvolvedores a primeiro definir todo o código SQL e, em seguida, passar apenas parâmetros para a consulta SQL; Isso permite que o banco de dados faça a distinção entre os dados que estão sendo inseridos e o código que está sendo executado, independentemente do tipo de dados fornecidos no campo de entrada.

"Escapar" todas as entradas fornecidas pelo usuário

Ao escrever a SQL, caracteres ou palavras específicos têm um significado particular. Por exemplo, o caractere "*" significa "qualquer" e a palavra "OR" é uma condicional. "Escapar" um caractere é a maneira de dizer ao banco de dados para não analisá-lo como um comando ou condicional, mas sim tratá-lo como entrada literal.

Usar procedimentos armazenados

■ Embora não seja uma estratégia de segurança robusta por si só, os procedimentos armazenados podem ajudar a limitar o risco associado à injeção de SQL. Os procedimentos armazenados também podem verificar o tipo de parâmetros de entrada, evitando que sejam inseridos dados que violem o tipo que o campo foi projetado para receber. Nos casos em que as consultas estáticas são insuficientes, os procedimentos armazenados normalmente devem ser evitados.

Reforçar o menor privilégio

É importante reduzir a exposição à injeção de SQL limitando as permissões ao escopo mais restrito necessário para executar a consulta relevante . Em sua forma mais óbvia, isso significa que uma conta administrativa não deve, em hipótese nenhuma, executar comandos de SQL como resultado de uma chamada de API de uma solicitação não autorizada. Embora os procedimentos armazenados sejam melhor utilizados para consultas estáticas, a aplicação do menor privilégio pode ajudar a reduzir os riscos de consultas SQL dinâmicas.

Quais propriedades são afetadas:

Um ataque de injeção de SQL bem-sucedido pode acarretar sérias consequências a uma empresa. Isso ocorre porque um ataque de injeção de SQL pode:

- **Expor dados confidenciais.** Os invasores podem recuperar dados, o que gera o risco de exposição de dados confidenciais armazenados no servidor SQL.
- **Comprometer a integridade dos dados.** Os invasores podem alterar ou excluir informações de seu sistema.
- **Comprometer a privacidade dos usuários.** Dependendo dos dados armazenados no servidor SQL, um ataque pode expor informações confidenciais dos usuários, como endereços, números de telefone e dados de cartões de crédito.

- **Conceder a um invasor acesso de administrador a seu sistema.** Se um usuário de banco de dados tiver privilégios administrativos, um invasor poderá obter acesso ao sistema usando código mal-intencionado.
- **Dar a um invasor acesso geral a seu sistema.** Se você usar comandos SQL fracos para verificar nomes de usuário e senhas, um invasor poderá obter acesso a seu sistema sem saber as credenciais de um usuário. Depois disso, o invasor poderá causar danos acessando e manipulando informações confidenciais.

Exemplos:



Em julho de 2012, o site da Yahoo foi violado por um ataque de injeção de SQL que vazou os dados de mais de 450 mil usuários, incluindo senhas, e-mails e nomes de usuário.

Em abril de 2014, o site da revista Forbes foi comprometido por um ataque de injeção de código que infectou os leitores com malware, explorando uma vulnerabilidade no Adobe Flash Player.

Em maio de 2015, o grupo hacker Lizard Squad realizou um ataque de injeção de SQL contra o site da Malaysia Airlines, redirecionando os visitantes para uma página com uma mensagem de apoio ao Estado Islâmico.

Em dezembro de 2020, a empresa SolarWinds foi alvo de um ataque de injeção de código que afetou milhares de organizações, incluindo agências governamentais e empresas privadas dos Estados Unidos