



**3º Informática Noturno**

**Feito por : Caio Souza Santos  
 Grupo: Gerenciamento de Finanças Pessoais**

**O que é Sql Injection?**

SQL Injection é um conhecido método utilizado em ataques a banco de dados através de formulários que contenham campos de entradas de dados do tipo texto.

Este método utiliza da vulnerabilidade do código implementado na aplicação para alterar a string de conexão ao banco de dados, comprometendo toda troca de informações entre “Aplicação – Base de Dados”.

O impacto causado pela alteração poderá ser apenas de um acesso indevido a uma área restrita, sendo imperceptível aos olhos de um desenvolvedor inexperiente, como também poderá permitir a exclusão de uma tabela, comprometendo toda a aplicação, dentre outros.

Aplicações diariamente estão sendo disponibilizadas na Web, sendo, a maioria dessas aplicações, dinâmicas, em que o acesso aos dados se torna necessário; a falta de informação da parte do desenvolvedor da aplicação a torna um alvo fácil de ataques. Exemplo de uma string simples de conexão via SQL entre a aplicação e base de dados.

**SELECT** id, senha **FROM** usuarios **WHERE** id = ‘Bruno’ **and** senha = ‘1234’;

A string definida na Ilustração acima retorna a confirmação se o usuário Bruno possui a senha ‘1234’. Caso esta informação seja verdadeira, irá localizar na tabela as informações e retornar a coleção de dados.

Porém a distração do desenvolvedor no tratamento da entrada de texto permitiu a digitação de caracteres especiais, permitindo assim que o usuário entrasse com id de ‘Bru’no’. Com esta entrada, a aplicação não irá conseguir interpretar o comando requisitado, retornando com um erro.

Assim como a entrada foi inválida, esta poderia ser válida, e a execução causar um transtorno enorme como exemplo

**SELECT** id, senha **FROM** usuarios **WHERE** id = ‘’ **or** 2 = 2 **and** senha = ‘1234’;

A aplicação permitiu que a entrada de texto pudesse conter o caractere ‘Aspas’, vulnerabilidade explorada pelo atacante para ter acesso às informações contidas na base, permitindo assim ações como inserir, editar, apagar etc.

O SQL Injection é uma das práticas mais utilizadas na Internet. Apesar de sua defesa ser bem simples, poucos desenvolvedores se atentam a isto. Segundo Fisher(2010), em junho deste ano iniciou-se o ataque mais recente, tendo como alvo servidores web que utilizam o software Microsoft IIS. A estimativa de páginas infectadas ultrapassa a marca de 100.000. Este ataque direcionava os visitantes para uma segunda página, onde era instalado um malware. De acordo com McMillan(2010), o ataque não explorar qualquer vulnerabilidade no IIS, mas é um ataque contra aplicativos Web de terceiros.

**Como prevenir**

O grande vilão responsável pela imensa maioria das vulnerabilidades de injeção de SQL é, sem sombra de dúvidas a concatenação de strings para montagem de comandos SQL. No entanto, é perfeitamente viável a construção de consultas dinâmicas sem a concatenação direta dos parâmetros à string principal. A API JDBC, por exemplo permite a utilização de prepared statements para execução de acessos ao SGBD.

Um prepared statement é uma instrução SQL pré-processada pela API do SGBD, que recebe os parâmetros separadamente da string de consulta principal. Na string principal, os parâmetros constam como marcadores que serão, posteriormente, substituídos pelo SGBD pelos seus respectivos valores durante a execução da consulta.

**SELECT** nome **FROM** funcionario **WHERE** departamento = **?**

Parâmetro 1: nomeDepartamento

Neste exemplo, o valor da variável nomeDepartamento será utilizado exclusivamente como critério de comparação com a coluna departamento. Caso seja recebido como valor de nomeDepartamento a string “ '' OR 1 = 1”, o resultado será a busca de um registro cujo valor de departamento seja a string “ '' OR 1 = 1”, garantindo assim o sentido original da consulta.

Evidentemente, é claro, as APIs e mecanismos dos SGBDs que fornecem funcionalidades como os prepared statements citados acima necessitam, internamente, serem seguras contra a injeção de SQL. A implementação insegura das camadas de acesso a dados pode tornar uma aplicação que as utilize completamente insegura, mesmo que esta tenha sido construída sob os mais criteriosos preceitos de programação segura.

**Ferramentas Automatizadas de Injeção SQL**

Existem ferramentas que fazem esse trabalho, e são bastante usadas tanto para testes de penetração como para invasões de Black Hat(Hacker Mal-Intencionado), procuram vulnerabilidades e injeta o comando sql de forma automatizada, Algumas ferramentas para esta finalidade sao:

1. **SqlMap**

Sqlmap é uma ferramenta open source para penetration test que automatiza o processo de detecção e exploiting de vulnerabilidades a Sql Injection.

É atualmente a ferramenta mais utilizada para esta finalidade.

O SqlMap pode rodar nativamente no Linux, no Windows você precisa dos complementos para poder executar programas em python.

1. **Havij**

Foi descontinuado no final de 2014. Em janeiro de 2015 o site da ITSecTeam foi fechado, encerrando todas as atividades.

  Havij é daquelas ferramentas "One click ownage", visual limpo, programa eficiente, rápido e totalmente intuitivo, isso é, você cola o endereço da página clica em "analyze" e está feito, não oferece tantos recursos e "poder" do sql map, mas é a ferramenta com GUI mais utilizadas para isto.