SQL-Injection

Eduardo Pereira
Fernanda da Silva Gomes
Ricardo dos Casaes Belo
Matheus Westhelle
Vilmar Dorneles Aprato Neto

O que é ?

SQL-injection é a inserção de código SQL usando um campo de input mal protegido, com a finalidade de fazer com que o programa execute ações fora de seu funcionamento normal.

É usado para dar acesso a informações ou funções restritas a um usuário comum, geralmente com a intenção de acessar ou modificar informação confidencial ou inserir informações falsas em um banco de dados.

Exemplo

Usando o seguinte BD como exemplo:

Cliente

#Cliente	Nome	Password
1	Eduardo	123
2	Vilmar	abc
3	Matheus	lol

Agencia

#Agencia	Cidade
1	POA
2	SP
3	RJ

CC

#Agencia	#Cliente	Saldo
1	1	50
2	1	75
3	3	200

Com um campo de input onde se entra o id do cliente e sua senha.

Exemplo

Se esse sistema hipotético for completamente desprotegido contra esse tipo de ataque então a query SQL usada para acesso ao DB é algo como:

Select nome

From Cliente

Where id = '***$id***' And password **= '*$password*'**

A onde $nome e $password são as variáveis contendo o input do usuário em alguma janela de login.

Exemplo

Um exemplo simples de input malicioso seria no campo password seria ***password' Or '1=1***, que nesse caso o parse do SQL interpreta-lá como:

Select nome

From Cliente

Where id = 'id' And password **= '*password' or '1=1***'

O comando ***password'*** completa a busca original, que pode até ser falsa para todos os passwords da BD, já que ela está ligada por um "or" a "***1=1***", que por sua vez retorna "True" para qualquer linha que exista na tabela cliente.

O resultado dessa query acaba sendo uma lista com todos os nomes dos clientes registrados no BD.

Exemplo

Várias alterações indesejadas podem ser feitas através desse mesmo método, como por exemplo, a criação de um cliente falso:

Select #id, nome, etc
From Usuario
Where #id = “id” And password = ‘***x’; Insert into cliente (‘#id, nome,password’);***
Values (‘4’,’Fernanda’,’321’);--’;

Ou deletar uma tabela inteira:

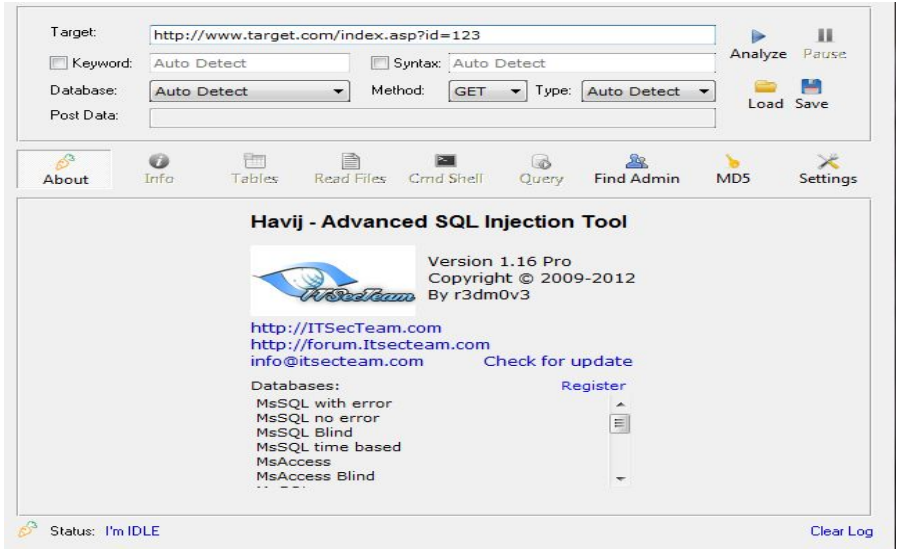
Select #id, nome, password
From cliente

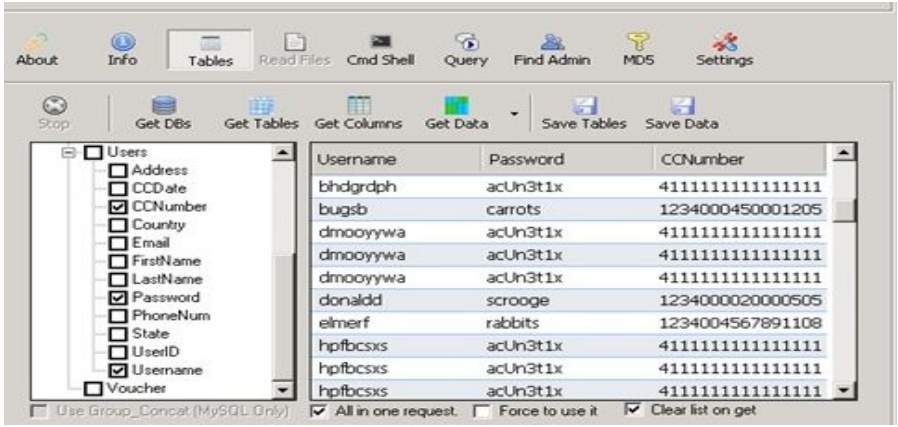
Where #id = “id” And password = ‘***x’; Drop Table Usuario;--***’;

Nestes exemplos o símbolo de comentário do SQL, “**--**”, é usado para comunicar ao parser que o ultimo “;” deve ser ignorado, efetivamente inserindo um novo comando no meio do programa.

Exemplo

Enquanto ataques de SQL injection manuais ainda acontecem, na “vida real” é mais comum o uso de programas que automaticamente buscam, detectam e usam aberturas em um determinado site:





Importancia

Mesmo sendo um tipo de ataque de fácil detecção e proteção, administradores descuidados e programas mal intencionados continuam a deixar aberturas para SQL injection.

Alvos mais comuns são sites de venda online, que são atacados geralmente com o objetivo de obter a informação dos cartão de créditos de seus usuários.

Em um estudo de 2014 da empresa de segurança digital Imperva, um site ficou sobre quase constante ataque durante tota a duração do mesmo (176 dos 180 dias).

Como evitar

Existem varias maneiras de se proteger:

Verificar o input e recusar sua entrada, se não for de um certo padrão esperado, um caractere como “=” em um campo “nome”, por exemplo.

Usar “Prepared staments”, um tipo de variável temporária que só aceita informação de um tipo especifico, ao invés de embarcar o input do usuario dentro do codigo SQL.

Usar algum tipo de função de tradução no nível de interface, que detectam certos caracteres especiais na String de input e adicionam um “\” após eles, bloqueando algum possível comando SQL injetado nela.

Fontes:

https://www.imperva.com/docs/HII_Web_Application_Attack_Report_Ed4.pdf

https://en.wikipedia.org/wiki/SQL_injection

http://www.unixwiz.net/techtips/sql-injection.html

http://www.acunetix.com/websitesecurity/sql-injection/