THE EXPERT'S VOICE IN SQL SERVER INJECTION AVOIDANCE

SQL SERVER INJECTION AVOIDANCE

Venkatesan Prabu Jayakantham

Microsoft Most Valuable Professional

ABOUT THE AUTHOR

). Aligned with KaaShiv InfoTech's mission,he

-
-
-
-
-
-
-
-
-
-
-
-

`

ACKNOWLEDGEMENT

DISCLAIMER

Intro about SQL Injection:

[Content taken from my company documentations]

The expansion of the Internet has made web applications become a part of everyday life. As a result the numbers of incidents which exploit web application vulnerabilities are increasing. A large percentage of these incidents are SQL Injection attacks which are a serious security threat to databases with potentially sensitive information. Therefore, much research has been done to detect and prevent these attacks and it resulted in a decline of SQL Injection attacks. However, there are still methods to bypass them and these methods are too complex to implement in real web applications.

A SQL Injection attack takes place when an attacker tries to gain access to a database by supplying special input to the web-site, which in turn sends the innocent input as an SQL-query to the Database Management System. The growing use of web-applications for business purposes has given motivation to attackers to explore the possibilities and exploit these type of attacks.

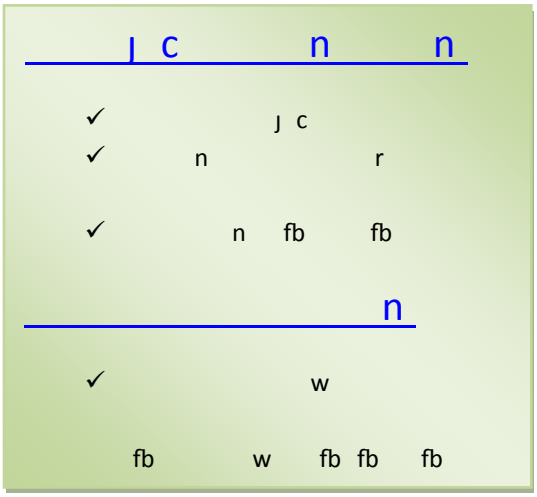
Am using a user defined stored procedure which will take care of validating the input values provided by the users and avoid the harmful SQL Server injection.

[Code]

```
CREATE FUNCTION dbo.SQLInjectionCheck_UserDefinedFun
(@VenkatString varchar(max))

RETURNS BIT
AS
BEGIN
DECLARE @Suspect_ValBit bit

SET @VenkatString = ' ' + @VenkatString
IF (PATINDEX('% xp_%' , @VenkatString ) <> 0 OR
PATINDEX('% sp_%' , @VenkatString ) <> 0 OR
PATINDEX('% DROP %' , @VenkatString ) <> 0 OR
PATINDEX('% GO %' , @VenkatString ) <> 0 OR
PATINDEX('% INSERT %' , @VenkatString ) <> 0 OR
PATINDEX('% UPDATE %' , @VenkatString ) <> 0 OR
PATINDEX('% DBCC %' , @VenkatString ) <> 0 OR
PATINDEX('% SHUTDOWN %' , @VenkatString )<> 0 OR
PATINDEX('% ALTER %' , @VenkatString )<> 0 OR
PATINDEX('% CREATE %' , @VenkatString ) <> 0OR
PATINDEX('%;%' , @VenkatString )<> 0 OR
PATINDEX('% EXECUTE %' , @VenkatString )<> 0 OR
PATINDEX('% BREAK %' , @VenkatString )<> 0 OR
PATINDEX('% BEGIN %' , @VenkatString )<> 0 OR
PATINDEX('% CHECKPOINT %' , @VenkatString )<> 0 OR
PATINDEX('% BREAK %' , @VenkatString )<> 0 OR
PATINDEX('% COMMIT %' , @VenkatString )<> 0 OR
PATINDEX('% TRANSACTION %' , @VenkatString )<> 0 OR
PATINDEX('% CURSOR %' , @VenkatString )<> 0 OR
PATINDEX('% GRANT %' , @VenkatString )<> 0 OR
PATINDEX('% DENY %' , @VenkatString )<> 0 OR
PATINDEX('% ESCAPE %' , @VenkatString )<> 0 OR
PATINDEX('% WHILE %' , @VenkatString )<> 0 OR
PATINDEX('% OPENDATASOURCE %' , @VenkatString )<> 0 OR
PATINDEX('% OPENQUERY %' , @VenkatString )<> 0 OR
PATINDEX('% OPENROWSET %' , @VenkatString )<> 0 OR
PATINDEX('% EXEC %' , @VenkatString )<> 0)
BEGIN
SELECT @Suspect_ValBit = 1
END
ELSE
BEGIN
SELECT @Suspect_ValBit = 0
END
RETURN (@Suspect_ValBit)
```



```
END
GO

 ---------------------
SELECT dbo.SQLInjectionCheck_UserDefinedFun
('SELECT * FROM HumanResources.Department')

---------------------- The result is "0"----------------------

SELECT dbo.SQLInjectionCheck_UserDefinedFun
(';SHUTDOWN')

---------------------- The result is "1"----------------------

SELECT dbo.SQLInjectionCheck_UserDefinedFun
('DROP HumanResources.Department')

---------------------- The result is "1"----------------------
```

[/Code]