

case snapshot

SQL Injection to the Portal Investigation and Fixing








SIRUSTI APPLICATION SERVICES

Business Challenge

- The client has an online portal with CMS functions to maintain their portal. They update live news frequently in CMS application and news will be published to public. This portal application was hacked. Hacker used SQL Injection to update the database tables with his site URL. Our client's portal was hacked almost every day periodically by the hacker using automated scripts and manually. Hacker injected his own script and website address into the SQL Database tables
- Because of these incidents, client's portal was down more frequently due to data restoration process and certain updated data was lost.
- The client chose Sirusti to investigate this incident and provide a solution for this.

Solution

Sirusti investigated the problem and found that the application architecture is allows SQL Injection and below was the statistics.

Total alerts found	1566
 High	55 
 Medium	0
 Low	771 
 Informational	740 

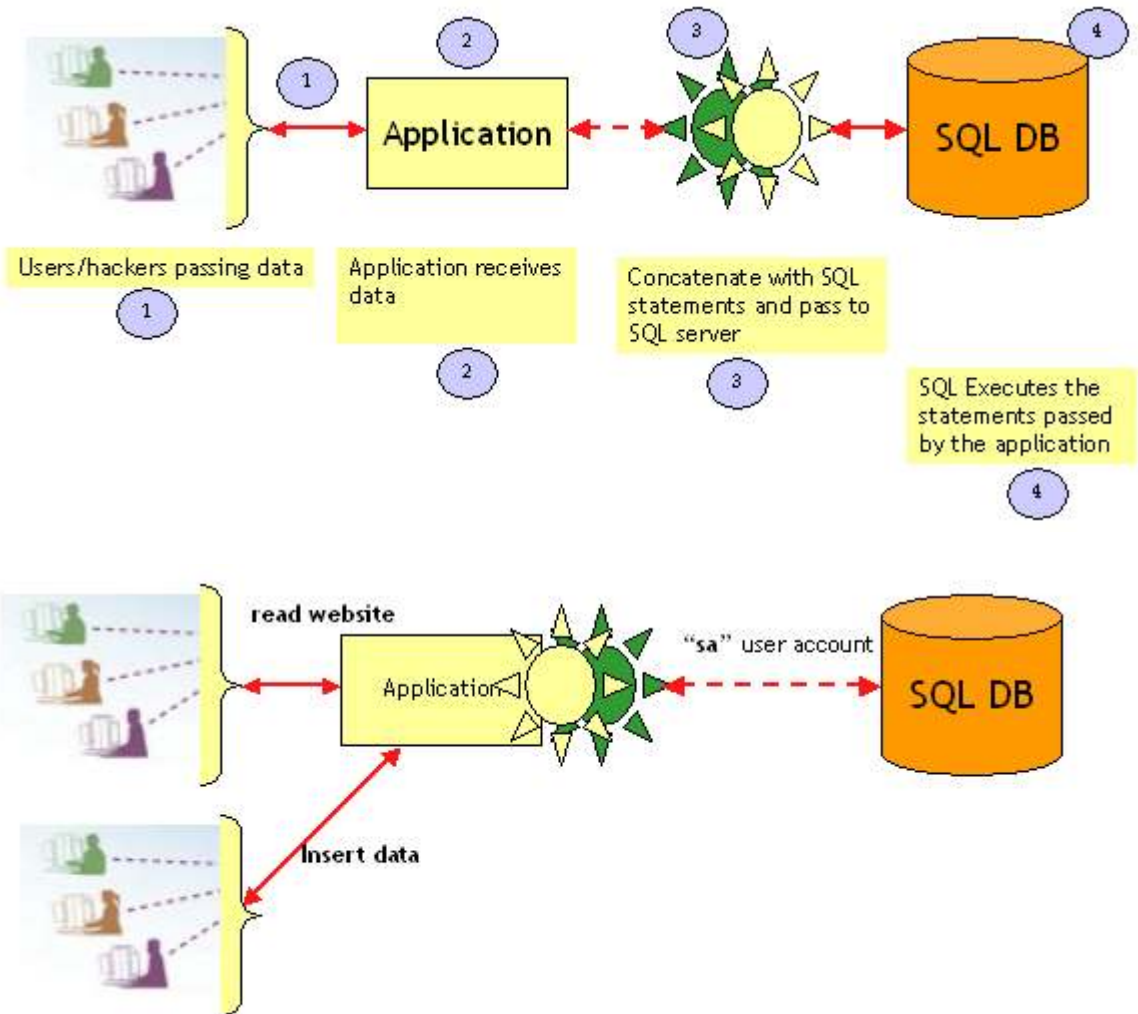
Total No. Of files updated: 277

SQL injection is a vulnerability that allows an attacker to alter backend SQL statements by manipulating the user input. An SQL injection occurs when web applications accept user input that is directly placed into a SQL statement and doesn't properly filter out dangerous characters.

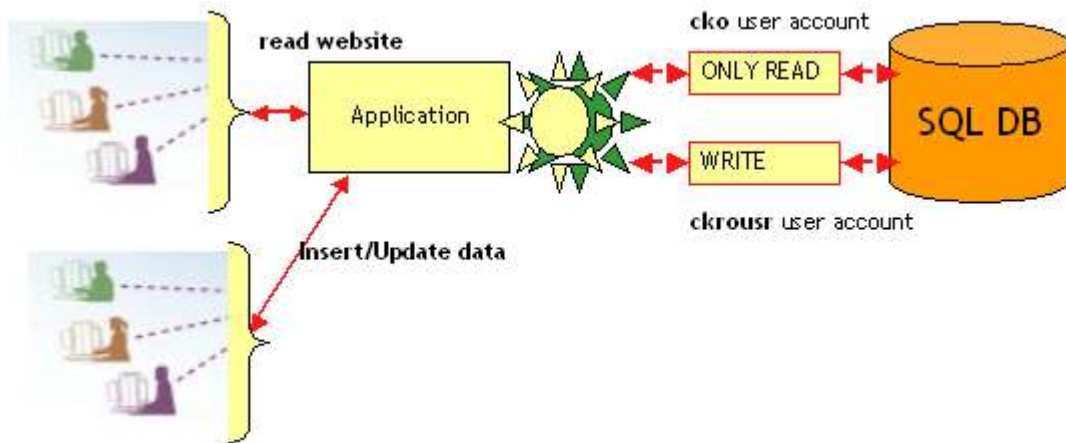
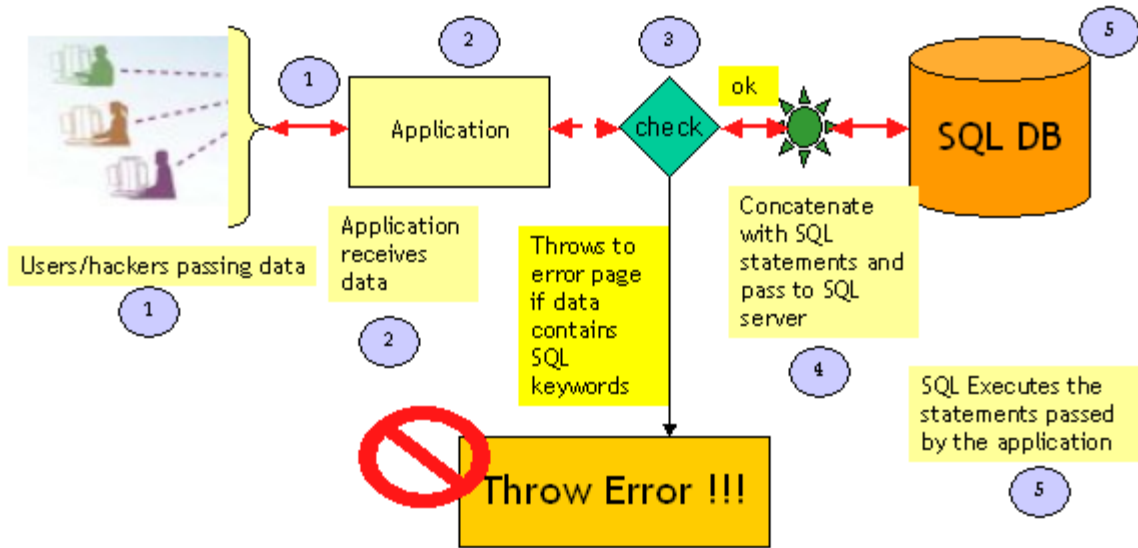
Ø First patch called "**quick fixes**" was applied into the source by using different SQL accounts based on the web page functionalities.

Ø The final deployment was to check the passing input parameter into the application and validate against SQL keywords before executing at database.

The below diagram shows the application structure before SQL injection fixes.



Sirusti restructured the database access without modifying current application functionalities. The below diagram shows the application structure after SQL injection fixes.



Software Specifications

- Windows 2003 Server
- IIS 6.0
- ASP.NET, VB.NET
- MS SQL Server 2005
- Web Vulnerability Scanning application

Exceed our Client's expectation

At Sirusti, we not only fix problems, but we provide additional services to exceed our client's expectation. In this project, our team suggested on our findings in the portal and advised to address those issues.

Below were our findings in the portal application.

1. User name & Passwords are not encrypted, it is stored in clear text. This may cause that any one access to the SQL server can retrieve all username & password and access the application without actual user's knowledge.
2. Application is NOT compiled. Also, Not able to compile and lot of errors are shown while compiling. Hence source code is being deployed in the server. It might cause the performance of the application.
3. Found lot of duplicated files. There are some databases connections in the project are missing.
4. Unwanted files are being hosted in the server.
5. No architecture followed in the application.
6. SQL Queries are in-built in the application. This will lead to SQL Injection easily and lead to mess of code. This should be organized by writing individual application control with stored procedures.
7. Business logic is not separated from User Interface.