Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 6, June 2015, pg.741 – 749

RESEARCH ARTICLE

Extended Security Techniques on Web Applications

Yashanjot Singh, Parminder Kaur
Guru Nanak Dev University, India
yashanbrar@gmail.com, parminder.dcse@gndu.ac.in

Abstract: Sql injection are one of the topmost threats for application written for the Web. In sql injection attacker gains an unauthorized access to the DB and some malicious codes are injected into it. This paper deals with web security as well as security techniques. To better counter these attack various techniques for detection and and prevention of SQL injection attack are identified in this paper also some predefined methods of detection and prevention are discussed .Finally we have come up with a new ideology i:e OTP(one time password) that will further enhance the security of web application from Sql injection.

***Keywords: SQL Injection, Sql query, Pattern matching, Web security*.**

1. INTRODUCTION

Computer security (also known as cyber security or IT security) is information security as applied to computing devices such as computers and smart phones and also to computer networks such as private and public networks . Internet security is a branch of computer security specifically related to the web security . Its focus is to establish rules and measures to use against attacks over the Internet. Web applications like e-commerce, online banking, enterprise collaboration and supply chain management sites, concludes that at least 92% of Web applications are vulnerable to some form of attack[1].Security techniques on the web applications need to be well defined so that they can deal with all kind of security threats. OWASP 2013 (online web application security project) has declared top vulnerabilities among which SQL Injection ranks the highest.[2].

2. WHAT IS SQL INJECTION

Sql injection is a technique in which attacker injects an input query in order to change the structure of the query and gain the access of the Database through unauthorized way. SQL Injection allows an attacker to create, read, update, alter, or delete data stored in the back-end database. In its most common form, SQL Injection allows attackers to access sensitive information such as social security numbers, credit card number or other financial data. The liability happens when user inputs one of either incorrectly string literal (‘, --, =), escape characters embedded in SQL statements. Attacker get a leads towards the hacking phase through the error message as the error message displayed by the web server depicts the type of database structure that has been used.

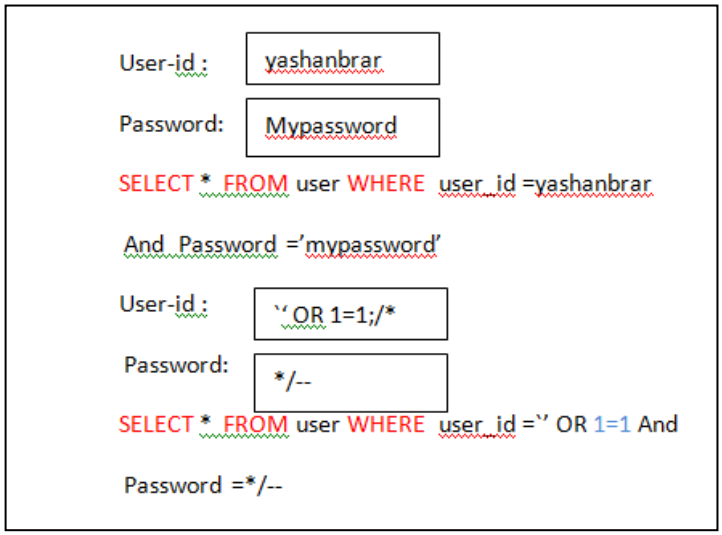
Eg of sql query

Select * From user WHERE username =admin AND password = 123

Eg of SQLinjection

Select * From user WHERE username = 'admin'--AND password = '

In the infected query system is going to to log in as the user=admin if user of this name exist in the system, because the query after (--) is ignored.



3. METHODOLOGIES USED FOR SQL INJECTION ATTACKS

- **Tautologies**: This type of attack injects SQL tokens to the conditional query statement to be evaluated always true. [12]
- **llegal/Logically Incorrect Queries**:When a query is rejected , an error message is returned from the database including useful debugging information. This error messages help attacker to find vulnerable parameters in the application and consequently database of the application[13].
- **Union Query**: By this technique, attackers join injected query to the safe query by the word UNION and then can get data about other tables from the application[14]
- **Piggy-backed Queries**: In this type of attack, intruders exploit database by the query delimiter, such as ";", to append extra query to the original query.[5]
- **Inference**: By this type of attack, intruders change the behavior of a database or application. There are two well known attack techniques that are based on inference: blind injection and timing attacks[12]
- **White space manipulation attack** : Blank spaces are used by the attackers to intrude into the system For example, the SQL-I pattern ' or 'a' <> 'b can be re-written as 'or 'a'<>'b'[15]
- **String Concatenation Attack**: SQL has an option to concatenate separate strings or characters to form complete strings. This is accomplished using + or "double pipe" (||)[15]

4. LITERATURE REVIEW

- (M.muthuprasanna,ke wei 2006) They have proposed a TDM(Transparent defense mechanism) in which they have combined static application code analysis and run time validation to detect the occurrence of such attacks this technique help in elimination the need to modify the source code of application script and additionaly allows seamless integration with current legacy systems . In first phase i:e static ananlysis stage they used program analysis technique to represent SQL query as FSA (finite state automata) and in second stage they check dynamically generated SQL query with static data structure and flag them .[16]

- (C.pinzon A.herroro,E.corchado ,2010) They proposed AIIDA-SQL (An Adaptive Intelligent Intrusion Detector Agent for Detecting SQL Injection Attacks) The AIIDA-SQL agent incorporates a Case-Based Reasoning (CBR) engine which is equipped with learning and adaptation capabilities for the classification of SQL queries and detection of malicious user requests. An advanced algorithm is incorporated in reasoning cycle stages and it help to classify the received SQL query in reliable way a projection neural technique is incorporated, which notably eases the revision stage carried out by human experts in the case of suspicious queries .CBR helps in solving the new problem by consulting the case memory to find similar case which has been resolved in the past .[17]

- (W.Halfond A.orso ,2005) They propose a tool AMNESIA (Analysis and Monitoring for Neutralizing SQL Injection attacks) .It consists of a static and a dynamic phase. During the static phase models for the different types of queries which an application can legally generate at each point of access to the database are built. During the dynamic phase queries are intercepted before they are sent to the database and are checked against the statically built models. If the queries violate the model then a SQL Injection Attack is detected and further queries are prevented from accessing the database. Our proposed approach does not consist of a static and dynamic phase. SQL Injection attacks are detected based on the behavior of the application with the help of runtime monitors developed by using our proposed framework.[18].

- (R.Dharam and S.G.Shiva,) They proposed Runtime Monitors for Tautology based SQL Injection Attacks which uses two pre-deployment testing techniques i.e. basis path and data flow testing techniques to identify legal execution paths of the software. Runtime monitors are then developed and integrated to observe the behavior of the software for identified execution paths such that their violation will help to detect and prevent tautology based SQL Injection Attacks. In this ,the source code contains certain critical variables that interact with the external world by accepting user inputs, build queries and process them by accessing the internal database and then monitor the behavior of application during its execution with respect to the indentified critical variable to detect and prevent tautology based SQLIAs.[19].

- (M.amutha ,M.kartileyan.K.marimuthu,2013) They proposed a technique for preventing SQL injection attacks Aho-corasick (pattern matching algorithm) in which a detection and prevention technique for preventing SQL Injection Attack (SQLIA) using Aho–Corasick pattern matching algorithm. In this paper, we proposed an overview of the architecture. In the initial stage evaluation, we consider some sample of standard attack patterns and it shows that the proposed algorithm is works well against the SQL Injection Attack. Aho corasick algorithm contains two phases static phase and dynamic phase . in static phase list of known anomaly patterns are maintained and user generated SQL wuery is are checked by applying pattern matching algorithm .in Dynamic phase if any new anomaly is generated them alarm is raised and anomaly pattern is generated which is further added to static pattern list . [20]

- (T,monatro, N.Abdul Aziz 2013) They proposed a study that presents a way to prevent and detect intrusion through the deployment of reverse proxy with an intrusion and prevention mechanism built in against web attacks especially SQLIA. With the flexibility offered in server logging process, we obtain and analyze preferred data to visualize the type of attack based on logs information. Their graph visualization development monitors

three web security aspects, i.e. the top traffic blocked attempted by IP address, number of regular expression rules violated and detect the rules of intrusion detection in this technjique a reverse proxy with an intrusion and prevention mechanism built in against web attacks like SQLIA.[21]

- (M.H. alattar, S.P medhane) They have proposed the Detection Model of SQL Injection Vulnerabilities and SQL Injection Mitigation Framework. These approaches are based on SQL Injection grammar to identify the SQL Injection vulnerabilities during software development and SQL Injection Attack on web-based applications. They tend to show a way to extract the SQL DOM mechanically from Associate in existing info schema, demonstrate its relevance to unravel the issues, and valuate its performance. They tend to project a SQL Injection Detection Model and SQL Injection Mitigation Framework to mitigate the SQL Injection Attacks (SQLIAs). Once mistreatment this potential resolution throughout software system development and once development, then we tend to might say that our net applications area unit secured from SQL Injection Attacks[22].

- (S.Panda, Ramani) They have discussed predefined methods and hybrid encryption method is applied in the database to avoid attack on login phase. This applied hybrid encryption method is a combination of Advanced Encryption Standard (AES) and Rabin cryptosystem. These two level encryption methods are applied to a system where faculty's information are kept and the designing of this system are done by using PHP and MYSQL The reason behind the use of two layer of encryption is that it will be more secured. SQL query is generated and encrypted by Rabin‟s cryptosystem because even if hackers hack the information and decode the AES encryption part, it will still be more difficult for them to know about the encrypted query..[23] .

- (S.Srivastava, R.R Tripathi,2012) They presented a new technique for prevention of SQL injection attack for web application.SQL injection attack can be easily prevented by applying more secure scheme in login phase. To address this problem they proposed a technique with highly secure login scheme which uses hash code with salt. In this approach they added one extra column is required in user account table to store Final hash value. This value is created at the time of new user registration and stored in user account table together with user name and password. At the time of login Final hash-code is calculated using stored procedure at run time and authentic user is identified by exact matching of username, password and final hashcode.[24]

- (M.kumar , L.indu,2014) They proposed the techniques for detection and prevention of SQL injection attack. There are no any known full proof defenses available against such type of attacks. They also come up with some predefined method of detection and the some modern techniques of preventions are discussed. Countermeasures of SQL injections area also discussed by them . Various detection techniques such as - code based detection techniques, code based detection techniques, Taint-based vulnerability detection. Similarly various prevention techniques such as Defensive coding , Manual defensive coding practice, Data type validation , White list filtration [25]

- (M.khari,A.karar,2013) They proposed a survey on Intrusion detection and prevention systems. Their work is carried out for details in intrusion detection and SQL based attacks. The result will help for database and IDS work together. Their paper contains work since 2002 to 2011 with some drawbacks and advantages suggested. Including SQL injection we have discussed about some XSS attacks and mimicry attacks. describes various approaches used by authors to prevent SQL injection attack using various methods like intrusion detection, black box testing etc.[26]

- (Varian Luong,2010) He introduced two ways for detection of Sql injection namely (i) signature based (ii) anomaly based . In signature based technique the information obtained from the Html application is compared

with the SQLinjection patterns , if the resultant information is matched then particular user access is denied .Whereas in anomaly based the behavior of the client computer is under supervision i:e number of times login attempt has been made , number of successful logins etc. This approach requires no user interaction, and no modification of either the backend database or the source code of the web application itself.[27]

- (R.Rai, J.jhadav 2013) They proposed a technique that uses a concept of filter called ―Smart Filter, that avoid the SQL injections with static matching and dynamic signature based intrusion detection mechanism with MS SQL database web application smart filter actually works in between the web application & database server. Therefore, before sending SQL queries to the database, the smart filter will analyze the query to check the vulnerability. If found any, it reported else it forwards the query to database server. Apart from the checking the SQL query by smart filter, it also reports the new vulnerabilities found in SQL queries. [28]

- (F.S Rietta,2006) In this paper examines the threat from SQL injection attacks, the reasons traditional database access control is not sufficient to stop them, and some of the techniques used to detect them. Moreover, it proposes a model for an anomalous SQL detector which observes the database traffic from the perspective of the database server itself. The proposed anomaly model can be used in conjunction with the existing methods to give the database server a way to mitigate the SQL injection risk that is a major application security problem. An application layer intrusion detection system should take the form of a proxy server and employ an anomaly detection model based on specific characteristics of SQL and the transaction history for a particular user and application[29].

- (M.gandhi ,J.baria 2013) They discussed about Advance SQL Injection (ASQLIA) first of all it identifies which type of attacks according to that prevention measures are suggested .Some New features are added to it Web Crawling ,Web Services and Advance SQL Injection (ASQLA)which will emphases more Security of Web Application technique presents the need for adding two additional columns in login table. These columns store hash values of username and password. When the user gets itself registered with a web application, it selects its username and password. At the same time, hash value of username and password is computed at the coding side and stored in the login table with username and password. When user logs in to the web application, hash value of username and password are matched at the backend and user is allowed to access the data. If SQL Injection attack. String is entered for logging into the database, its hash value does not match with the hash values stored in the table and hence attacker cannot access the database[30].

- (K.ahmed ,J.shekhar,K.P.Yadav ,2010) They have classified the SQL injection attacks in order to understand the attacks more carefully. This classification allows to understand different kind of vulnerabilities which lie under different classes of SQL attacks that help the developer to avoid the cause of occurrence of SQLinjection attacks. This classification of SQL injection attacks helps in reducing the possibility of vulnerability that can damage the database security [31]

Several ways to detect SQL vulnerabilities are:

4.1 Safeli : They provided a fixed analysis framework to look for Sql vulnerabilities .It contain two main advantages .Firstly, it contains a static white box static analysis which deals with byte coding and with strings and secondly it contains Hybrid constraint convergent which deals with string analysis technique which checks every string variable and number .

4.2 Haixia and zhihongs : They delivered an secure information testing style in which they look for several things firstly, detection of potential input string secondly , built up test cases that deal with information by executing a test case to a particular degree application.

4.3 Automated SQl-IDS : In this system they used a tainting mechanism to find out legitimate queries from the malicious queries it takes query as a combination of tokens and compare each with the trusted and untrusted data to make sure that all parts of query consist of only trusted data . This approach deals with dynamic detection of Sql injection attacks also, it identifies 'trusted' strings and only those are used to create the semantically relevant parts of Sql query such as keyword or operators .this approach uses dynamic tainting in which marks certain data in program runtime .[6]

4.4 Black Box Testing(Waves) : It is a black-box technique for testing Web Applications for SQL injection vulnerabilities. In this technique they used a web crawler tool to find any point in web application that can be used to provoke SQLinjection .After finding the points they form an attack that targets such points based on particular pattern and attack techniques. Then it monitors application reaction to the attack and make it enhance its attack methodology. [7]

4.5 Runtime monitoring: It presented a framework which can be used to handle tautology based SQL Injection Attacks using post-deployment monitoring technique. Their framework includes two pre-deployment testing techniques i.e. basis path and data flow testing techniques to identify legal execution paths of the software. Runtime monitors are then developed and incorporated to scrutinize the behavior of the software for identified execution paths such that their breach will help to detect and prevent tautology based SQL Injection Attacks.

5. PREVENTION OF SQLIA

5.1 Stored procedure : This technique deals with methodology that prevent SQL injection attacks in Stored procedures in the database layer although ,many researchers have proposed the methods to prevent SQL injection in the application layer. In this technique there are two phase security to the application, so that, if one phase is compromised, the second phase can still prevent the attack. The proposed solutions for preventing SQLIA provide security to either application layer or database layer but not to both. They have proposed a technique that provides security to both application layer as well as database layer via frontend phase and backend phase. Researchers have provided this two phase security because if security is compromised at one phase, the second phase can still provide security from attacks

5.2 PSIAIW : This technique deals with the hash values for username and password are used at runtime .On attempting relogin process username and password plus hash key values are checked first .Thus with the help of hash keys attacker doesn't know about the hash keys no matter even if he/she knows the username and password and hence the access to database if denied .Every time the DB is accessed the hash value of provided parameter is calculated and matched with the stored one . [8]

5.3 Defensive coding : The basic solution for eliminating these vulnerabilities is to apply suitable defensive coding practices (i)Input type checking: SQLIAs can be performed by injecting commands into either a string or numeric parameter. Even a simple check of such inputs can prevent many attacks. (ii) Positive pattern matching: In this input validation routines that identify good input as opposed to bad input. This approach is generally called positive validation, as opposed to negative validation, which searches input for forbidden patterns or SQL tokens.

5.4 Sql guard and Sql check : In these approaches, the model is articulated as a grammar that only accepts legal queries. In SQLGuard, the model is deduced at runtime by examining the structure of the query before and after the addition of user-input. In SQLCheck, the model is specified independently by the developer. Both approaches use a secret key to delimit user input during parsing by the runtime checker, so security of the approach is dependent on attackers not being able to discover the key. Additionally, the use of these two approaches requires the developer to either rewrite code to use a special intermediate library or manually insert special markers into the code where user input is added to a dynamically generated query.[9][10]

5.5 Combined Static and Dynamic Analysis(amnesia) : It is a model based technique that use static analysis to build models of the different types of queries an application can legally generate at each point of access to the database. Initially it intercepts all the queries before they are send to the DB and check each of query against

statistically built models .Queries that violate the models are recognized as the SQLIA and prevented from executing on the DB .[11]

5.6 Escaping If dynamic queries cannot be avoided, escaping all user-supplied parameters is the best option. Then the developer should identify the all input sources to define the parameter that need escaping, follow database-specific escaping procedures, and use standard defining libraries instead of the custom escaping methods.

5.7 Data type validation: After following the steps for the parameterized query and escaping the developer must properly validate the input data type. The developer must define the input data type is string or numeric or any other type and input data given by user is incorrect then it could easily reject.

5.8 White list filtering: Some of the special character which is normally used during injection .so the developer should characterize such special character as the black list filtering. The filtering approach is suitable for the well structured data. Such as email address, dates, etc. and developer should keep a list of legitimate data patterns and accept only matching input data

6. CONCLUSION AND FUTURE WORK

This Paper Deals with Web Security as well as security techniques .Today application software is characterized by its resistance power against intrusions. A security mechanism for application software against intruders is must. The work represented here introduced a technique by which the security against the application software security can be achieved. By IP Pattern tracking methodology the attacker is permanently put into Blacklist and thus avoiding any further intrusion attempts. While several techniques are available to mitigate the risk of SQL injection attacks, we propose that an additional measure of protection be added. With the addition of the OTP technology, our system would attain a high security environment as the online application threat is concerned.

REFERENCES

1. *[1]"S. W. Boyd and A. D. Keromytis, "SQLRand: Preventing SQL injection attacks," in Proc. of ACNS, 2004."*
2. *[2]"top ten most critical web application vulnerabilities", OWASP Foundation, http://www.owasp.org/documentation/topten.html,2013*
3. *[3] William G.J. Halfond, Jeremy Viegas, and Alessandro Orso (2006): A Classification of SQL Injection Attack*
4. *[4] San-Tsai Sun, Ting Han Wei, Stephen Liu, and Sheung Lau: Classification of SQL Injection Attacks. Electrical and Computer Engineering, University of British Columbia*
5. *[5] C. Anley. Advanced SQL Injection In SQL Server Applications. White paper, Next Generation Security Software Ltd., 2002.*
6. *[6] F. Valeur and D. Mutz and G. Vigna "A Learning-Based Approach to the Detection of SQL Attacks," In Proceedings of the Conference on Detection of Intrusions and Malware Vulnerability Assessment (DIMVA), July 2005.*
7. *[7] Y. Huang, S. Huang, T. Lin, and C. Tsai. Web Application Security Assessment by Fault Injection and Behavior Monitoring. In proceedings of the 11th International World Wide Web Conference(WWW 03), May 2003.*
8. *[8] By1Prasant Singh Yadav, 2 Dr pankajYadav, 3Dr. K.P.Yadav "A Modern Mechanism to Avoid SQL Injection Attacks in Web Applications",IJRREST: International Journal of Research Review in Engineering Science and Technology ,Volume-1 Issue-1, June 2012.*
9. *[9] S. W. Boyd and A. D. Keromytis. SQLrand: Preventing SQL Injection Attacks. In Proceedings of the 2nd Applied Cryptography and Network Security (ACNS) Conference, pages 292–302, June 2004.*

10. *[10] G. T. Buehrer, B. W. Weide, and P. A. G. Sivilotti. Using Parse Tree Validation to Prevent SQL Injection Attacks. In International Workshop on Software Engineering and Middleware (SEM), 2005*

11. *[11] W. G. Halfond and A. Orso. AMNESIA: Analysis and Monitoring for NEutralizing SQL-Injection Attacks. In Proceedings of the IEEE and ACM International Conference on Automated Software Engineering (ASE 2005), Long Beach, CA, USA, Nov 2005.*

12. *[12] Khaleel Ahmad*, 2Jayant Shekhar and 3K.P. Yadav Classification of SQL Injection Attacks Vsrd technical and non technical society VSRD-TNTJ, Vol. I (4), 2010, 235-242*

13. *[13] By AtefehTajpour ,Suhaimi Ibrahim, Mohammad SharifiWeb Application Security by SQL Injection DetectionTools.IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 2, No 3, March 2012*

14. *[14] Mihir Gandhi, JwalantBaria : SQL INJECTION Attacks in Web Application. International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-6, January 2013*

15. *[15] Varian Luong : Intrusion Detection And Prevention System: SQLInjection Attacks,2010.*

16. *[16] M. Muthuprasanna, Ke Wei, Eliminating SQL injection attacks – A transparent defense mechanism Eighth IEEE International Symposium on Web Site Evolution (WSE'06) 0-7695-2696-9/06 $20.00 © 2006*

17. *[17] C.pinzon A.herroro,E.corchado, AIIDA-SQL (Adaptive Intelligent Intrusion Detector Agent) 2010, 10TH International conference on hybrid intelligent systems 978-1-4244-7365-6/10/ IEEE*

18. *[18] W. G. Halfond and A. Orso, "AMNESIA: Analysis and Monitoring for Neutralizing SQL-Injection Attacks", Proceedings of the IEEE and ACM International Conference on Automated Software Engineering (ASE 2005), Nov 2005.*

19. *[19] R.Dharam and S.G.Shiva, Monitors for Tautology based SQL Injection Attacks ,*

20. *[20] M.Amutha ,M.Kartikeyan.K.marimuthu, An efficient technique for preventing Sql injection attack using pattern matching algorithm , 2013 IEEE International Conference on Emerging Trends in Computing, Communication and Nanotechnology (ICECCN 2013)*

21. *[21] T,Monatro, Normaziah binti Abdul Aziz 2013, Log Visualization of Intrusion and Prevention Reverse Proxy Server Against Web Attacks, 2013 International Conference on Informatics and Creative Multimedia*

22. *[22] M.H. Alattar, S.P medhane, Efficient Solution for SQL Injection Attack Detection and Prevention, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-3, Issue-1, March 2013*

23. *[23] S.Panda, Ramani Protection of Web Application against Sql Injection Attacks International Journal of Modern Engineering Research IJMER Vol.3, Issue.1, Jan-Feb. 2013 pp-166-168.*

24. *[24] S.Srivastava, R.R Tripathi , Attacks Due to SQL Injection & Their Prevention Method for Web-Application, International Journal of Computer Science and Information Technologies, Vol. 3 (2) , 2012,3615-3618*

25. *[25] Manish Kumar ,L.indu Detection and Prevention of SQL Injection attack (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (1) , 2014, 374-377.*

26. *[26] M.khari,A.karar,2013 ,preventing sql injection attacks using intrusion detection and prevention system .*

27. *[27] V.lounge,2010, Intrusion detection and prevention system ,SJSU scholarworks http://scholarworks.sjsu.edu/etd_projects/16.*

28. *[28] R.Rai, J.jhadav,2013 Implementaion of smart filter to avoid sql injection with signature based intrusion detection*

29. *[29] F.S Rietta , Application Layer Intrusion Detection for SQL Injection, ACM SE'06 March 1012, 2006, Melbourne, Florida, USA Copyright 2006 ACM 1595933158/06/0004.*

30. *[30] M .gandhi ,J.baria, SQL Injection Attacks in Web Application , International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-6, January2013*

31. *[31] K.ahmed ,J.shekhar,K.P.Yadav Classification of SQL injection attacks , VSRD Technical & Non-Technical Journal Vol. I (4), 2010.*

