Research on penetration test of the SQL injection based on the formalization model

CHEN Ping[1, a]

[1] College of Command Information Systems,PLA University of Science and Technology,Nanjing, 210007,China

[a]email: chenpin0361@sina.com

Keywords: SQL injection; penetration test; vulnerability; test case

Abstract. To solve the problem of generating adequate test cases to reduce omissive report of the SQL injection vulnerability in penetration testing, this paper proposes a model-driven penetration test case generation method, which can describe the regularity of current SQL injection attacks. The experiment shows that the test cases generated by the proposed method can more effectively find the SQL injection vulnerability hidden behind the inadequate defense mechanism, and can reduce the omissive report of SQL injection.

Introduction

SQL injection vulnerabilities are a great threat to the security of web applications[1], it is necessary to accurately find the SQL injection vulnerabilities in web applications and timely repair them. At present, there are two main ways for Web application security vulnerability testing: source analysis (white box testing) and penetration testing (black box)[2]. The penetration test is to find out the software security vulnerabilities in the way of simulating attack. This method is more convenient, low complexity and wide adaptability relative to the method of source analysis, so the research on the penetration testing is increasingly concerned. The penetration testing of SQL injection vulnerability in web application is one of the research hotspots.

The current research about improving the accuracy of SQL injection penetration test focus on enhancing the ability of Web application information collection and vulnerability analysis[3][4],and the theoretical study on how to optimization the test cases to improve the accuracy of SQL injection penetration test is not enough.The related research in this field has not given appropriate theory method to guide the generation of multi-species, multi-form of SQL injection penetration test cases to find more fully SQL injection vulnerabilities, which will cause the omissive report of SQL injection vulnerabilities and reduce the test accuracy.

This paper studies how to improve the accuracy of penetration test from the perspective of optimizing the test cases in penetration test of SQL injection vulnerabilities.

SQL injection attack modeling

Penetration testing is a security test through simulating attacks, which actually test whether the web application is dangerous or not in front of the attack. So understanding the regularity of attack, and application of the attack regularity is the fundamentality of penetration test to determine the existence of web application security vulnerabilities. Model is helpful for people to understand the inherent regularity of complex behavior, so establishing model is important to guide the implementation of penetration testing.

According to the characteristics of SQL injection attack, the attack tree[5] is used for modeling currently, and the research shows that the model has some shortcomings, such as attack behavior and results are expressed by nodes, which will cause confusion; tree branches and nodes are often redundancy and duplication,etc.

Security goal model(SGM)[6] is a new model used to describe vulnerabilities, security features, attack or security software development. As an improvement to the attack tree model, this paper

applied the new SGM to build a model for the SQL injection attack. The SGM describe things from the view of the security related behavior target, so this paper builds model for SQL injection attack based on the attacker's direct attack target , as shown in Figure 1.

In Figure 1, the total target of the SQL injection attack is described by the root node in the model, and then the target attack is divided into four kinds: steal the system information, bypass authentication, tamper with the database, the direct implementation of the attack, the upper nodes in the model further describe the realization of these four seed target until the top nodes which describe the modes of attack injection and finding the injection points.

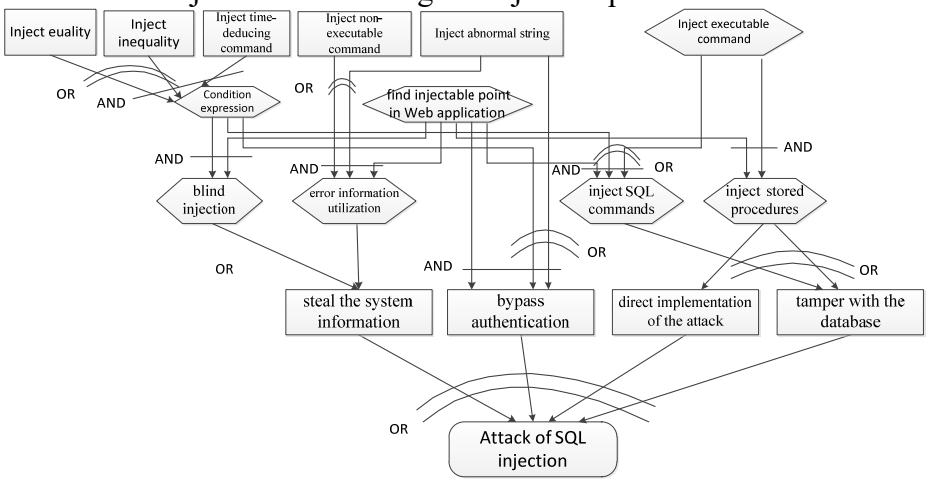

Fig.1. the SGM diagram of SQL injection attack

According to the description of SQL injection attack in Figure 1, the direct attack modes of SQL injection can be summarized as 5 kinds: blind injection, error information utilization, bypass authentication, execute SQL commands, and execute stored procedures.

Each attack pattern can be described by three tuple <OBJ, INP, OUT>, OBJ is the target of attack, INP is the input, OUT is the response of the Web application when security vulnerabilities exist.

Need to note is that some Web application system has taken the black list defense mechanism based on keyword filtering to avoid the SQL injection vulnerabilities, the attacker can use the method of camouflage with encoding SQL injection code to bypass this simple defense mechanism, which applied to all attack targets. Thus, this attack method is not reflected in the SGM diagram, but when generation the attack cases, which should be considered as the input of each attack.

Generation of SQL injection penetration test case

The formal description of SQL injection attack input: operator Δi is defined to represent some kind of attack input set, such as ΔSQL stands for the total set of SQL injection attacks input. Table l gives the definition of the set Δ, table 2 gives the definition of operators. The operator is used to combine the set Δ to form complex expression in order to describe the attack input set and its law to achieve some kind of attack target.

Table 1 the definiation of set Δ

set	definition	set	definition
Δ_{DS}	Abnormal string set	Δ_{CON}	Condition expression set
Δ_{LG}	Abnormal string set of disturbing authentication	Δ_{IE}	Equality set
Δ_{DC}	Non-executable command set	Δ_{NE}	Inequality set
Δ_{TI}	Time-relat command set	Δ_{COMS}	Executable command set
Δ_{SQLC}	Executable SQLcommand	Δ_{SPC}	Executable stored procedure set

Table 2 the definiation of operator

operator	definition	operator	definition
•	Δ1•Δ2 means use Δ1 to process the injection parameters of Δ2	AND	Constructing conditional formula with AND predicate，used for Δ_{CON}
OR	Constructing conditional formula with OR predicate, used for Δ_{CON}	DISG	Camouflage the injection
||	Δ1||Δ2means the Δ1 or the 2 can all achive the goal	&&	Δl &&Δ2means Δl and Δ2 should be used together to achive a goal

According table 1, table 2, the five attack modes are described in table 3, the formal expression of attack input can overcome the unlimity and irregular of the method of random enumeration use case , on the other hand, the classification of various SQL injection attack mode can guide the generation of different attack types of cases.

Table 3 attack modes of SQL injection

Object of attack	Attck input
error information utilization	$\Delta_{DS}\|\|\Delta_{DC}\|\|\text{DISG}(\Delta_{DS}\|\|\Delta_{DC})$
blind injection	$\Delta_{TI}\bullet\Delta_{CON}\|\|\text{AND}(\Delta_{IE})\&\&\ \text{AND}(\Delta_{NE})\|\|\text{DISG}(\Delta_{TI}\bullet\Delta_{CON}\|\|\text{AND}(\Delta_{IE})\&\&\ \text{AND}(\Delta_{NE}))$
execute SQL commands	$\Delta_{SQLC}\|\|\text{OR}(\Delta_{IE})\ \|\|\text{DISG}(\Delta_{SQLC}\|\|\text{OR}(\Delta_{IE}))$
execute stored procedures	$\Delta_{SP}\|\|\ \text{DISG}(\Delta_{SP})$
bypass authentication	$\text{OR}(\Delta_{IE})\|\|\Delta_{LG}\|\|\text{DISG}(\text{OR}(\Delta_{IE})\|\|\Delta_{LG})$

Experimental results and comparison

This research concerns the problem of penetration test case, that is, the ability of the use case set to find the SQL security vulnerabilities behind the black list defense mechanism.

So we develop a web application as the target system, which has two levels of defense measures: Level 0 is corresponding to the ordinary non-security measure for SQL injection vulnerabilities；Level 1 has the black list filtering measure, but it is not sufficient (that is, only filter the key words in the list of possible attack input, and do not filter other attack inputs).

In order to verify the adequacy of the method of test case generation proposed in this paper, we have developed an automatic penetration testing system of Web application security vulnerabilities. The testing system includes the crawler module, the test case injection module and the vulnerability detection module. The crawler module is used to get all the pages of the Web application, and extract the URLs with the parameters and FORMs as the input points. The test case injection module inject the attack mode cases into the corresponding input point; the judge module determine whether the existence of loopholes according to the response of Web.

According to the description in second section of the paper, the instantiate model of the attack mode is set as shown in Table 4.

Table 4 the instantiate cases for the penetration test

INP	Instantiate of INP
Δ_{DS}	Select 5 random characters/string, such as @^*$#
Δ_{SQLC}	MV={create, alter, drop, select, update, insert, delete}, each verb in MV constructs a use case
Δ_{SPC}	MV={ Random select 3 stored procedures }, each stored procedure in MV constructs a use case
Δ_{TI}	MV={waitfor, benchmark, sleep}, each verb in MV constructs a use case
Δ_{IE}、Δ_{NE}	OP={<>,>,>,<=,>=, between, IN, like},each predicate in OP constructs a use case
DISG	DG={Unicode、ASCII }

According to table 4, generate about 103 test cases (including the use case of camouflage and non disguised form).

We select the well-known Web vulnerability testing tools Acunetix 6.5 and Rational AppScan IBM 7.7 for comparison,and call them tool A and tool B. Tool A and ToolB are the representatives of the general use of the random enumeration method in the current penetration testing research. the penetration testing tool develops in this paper called Mytool.

For undefended input points (1evel 0),three tools can test all SQL injection vulnerabilities,and in the situations of not sufficient defensive (1evel 1), tool A and B can not report some vulnerabilities. The method proposed in the paper can test out all SQL injection vulnerabilities hidden behind the unsufficient defense.

Summary

Improving the accuracy of testing is the fundamental purpose of the research in the field of penetration testing, this paper studies the problem of improving the accuracy of penetration test from the perspective of improving the use case.

The research shows that the penetration test case generating based on the SGM model of SQL injection attack can more effectivly find SQL injection vulnerabilities in the web application with non-sufficient defense , and improve the accuracy of the test.

References

[1]Antunes J，Neves N，Correia M，et a1．Vulnerability discovery with attack injection．IEEE Transactions on Software Engineering，2010，36(3)：357-369

[2]Bau J，Bursztein E，Gupta D，et a1．State of the art：automated black-box web application vulnerability testing．In：Proceedings of the 2010 IEEE Symposium on Security and Privacy，Oakland，USA，2010．332-345

[3]S.Kals，E.Krida，C.Kruegel，et a1. SecuBat:A Web Vulnerability Scanner.In：Proceedings of the 15th International Conference on World Wide Web．New York,USA：ACM，2006．247-256

[4]William G.J．Halfond，Shauvik Roy Choudhary，Alessandro Orso．Improving Penetration Testing Through Static and Dynamic Analysis．In：Proceedings of the Second IEEE International Conference on Software Testing，Verification and Validation．West Sussex．UK:John Wiley and Sons Ltd，2011.195-214

[5] Andreas L.Opdahl，Guttorm Sindre．Experimental comparison of attack trees and misuse cases for security threat identification．Information and Software Technology，2009，V01.51(5)：916-932

[6] David Byers，Nahid Shahmehri．Unified modeling of attacks，vulnerabilities and security activities．In：Proceedings of the 2010 ICSE Workshop on Software Engineering for Secure Systems，Cape Town，South africa：IEEE Press，2010.36-42