

# Quality Test Template toward Multi-user Access Control of Internet-Based System

Nan Nie

School of Computer and Communication Engineering, Zhengzhou University of Light Industry  
Zhengzhou, China

Email: [oraclenienan@gmail.com](mailto:oraclenienan@gmail.com)

Suzhi Zhang

School of Computer and Communication Engineering, Zhengzhou University of Light Industry  
Zhengzhou, China

Email: [zhsuzhi@zzuli.edu.cn](mailto:zhsuzhi@zzuli.edu.cn)

**Abstract**—Aiming at three kinds of Internet-based system quality problems, which is performance, liability and security, the paper proposes a kind of test template during multi-user login and resource access control, which includes test requirement, login script, role-resource correlating and mutation test technique. Some Internet-based systems are tested and diagnosed by automation test technique of test template. At last, system quality can be verified and improved through the realization mechanism of test template.

**Index Terms**—Internet-based system, access control, test script, template, liability, software quality

## I. INTRODUCTION

With the rapid development of Internet, there are some large Internet-based systems, which have a deep effect on our society, for example, Orkut (<http://www.orkut.com>) and Xing (<http://xing.com>) of Web-based social network [1]. Network Game (<http://qqgame.qq.com/> and <http://www.ourgame.com/>) uses online interaction technique of virtual user group and game servers. Security, performance, especially liability plays a central role in determining the overall quality and usability of these Internet-based systems.

Firstly, reliability means a system keeps operating despite internal changes (such as part failure). Software reliability is defined as the probability of failure-free software operation for a specified period of time in a specified environment [2-3]. Secondly, with the rapid growth of large Internet-based system, performance has a deep effect on working of the system. So, developing test strategies of comprehensive liability and performance is essential for Internet-based system. For instance, ref. [4] presents a comprehensive review of recent research in the field of model-based performance prediction at software development time in order to assess the maturity of the field and point out promising research directions. Ref. [5] provides a new approach for the role of modeling in the performance testing of e-commerce applications, which is based upon performance signatures and software

rejuvenation. Furthermore, mechanisms that increase fault-tolerance (reliability) and performance are able to reduce security but are illogical if reliability and performance is part of security. For example, the Internet is constantly under attack as witnessed in recent reports. The overall severity of vulnerabilities increased, with high and critical severity vulnerabilities up 15.3 percent and medium severity vulnerabilities up 67.5 percent [6]. Many vulnerabilities are related with system performance and liability.

Access control is one of the most fundamental and widely used security mechanisms. The importance of access control is growing rapidly where computers are ever-more interconnected in a world, ref.[7] use model check technique to verify access control systems. When access control systems are built, access control policies will influence liability and performance of the system. So developing test strategies of comprehensive access control is essentially useful for improvement of Internet-based system quality.

There have been a few works published to describe approaches to resolve one of above problems. This paper proposes a kind of test models during users' login and resource access control toward multi-subsystems, which try to find some quality problems of software related with performance, liability and security of system. Even it can be used to validate the efficiency of private matter protection in the legal sense. It is the main goal of this paper to define test template and generate test suites for testing login and access.

## II. TEST TEMPLATE INITIATION

### A. Test Requirement

According to an investigation report of IAG in 2008, if a company focuses on business' requiring assignments and executing them consistently, 73% of projects should turn out successful with only the rare project ending up a failure. Software requirements derive from user requirements, which in turn derive from business requirements. In large-scale and complex systems, performance, security and liability of software

---

Manuscript received April 6, 2010;

requirements are the most important. So test requirement are primary part of test template.

We can integrate requirements across the application lifecycle by using some requirement tools, for example, Collaboration Server of IBM RequisitePro. Testing requirements is an important factor of Software requirement, ref.[8] proposes a formal method called Viewcharts for early software reliability assessment, which is based on software behavioral requirements. We firstly achieve the documents of requirement analysis of system, to exchange ideas with developer and customer, and to provide a piece of test outline, which includes quality indexes and data of test requirement of function, performance and security. The test and evaluation of the quality indexes will determine the quality of Internet-based system.

### B. Test Script

In order to reduce system test cost and promote system test quality and efficiency, we usually need to do with the help of automatic tools. Browsing the Internet, one can find an immense variety of tools supporting different activities performed during the test process, such as Smartbit of Spirent, HP MI Quick Test Pro(QTP), HP LoadRunner and IBM App Scan etc. We not only use Avalanche of Spirent and trace plus tool to assure well-working of the network, but also use LoadRunner to test the work of Web-system. LoadRunner can be used to test liability and performance of our system. QTP, which can be used to create function test script, is a graphical interface record-playback automation tool. Both of them can use test data by a lot of methods, such as only once, unique and random etc.

Firstly, we need to define test script according to the type, protocol and scale of system. The instructions in a test program define the actions and pass/fail criteria. For example, if the action is "to enter a valid account number," the expected result is that the data are accepted. Entering an invalid number should yield a particular error message.

The test tool cannot provide test case towards a kind of application system. If we can create the general test script framework model for the application, then translate it into application script to run on a practical system. A test script can realize required preconditions, the execution steps, set-up and clear-down facilities. Additionally, test scripts can implement generation produces for test data and expected outcomes. Test flow of system login and access is shown as Fig.1. It illustrates the test produce from the test of network environment to the generation of test script. At last, test results are modified and analyzed to improve system quality.

There is a lot of system performance constraints existing in system client and server, such as (1) network bandwidth and login time; (2) browsers function compatibility and password checking delay; (3) connect type (session restrict); (4) login abnormal prompt process, for example the detection of buffer overflow by a fitness function that takes into account static and dynamic analysis technique [9]; (5) maximum and minimum value

of system user; (6) test tools bug etc. Test script can identify system all test objects, such as system platform information including box, menu and buttons. So we can achieve these objects parameters, and build access tables between users and resources. This is the first step of building test template. However, the crucial step is to build complete and efficient data correlation for test template.

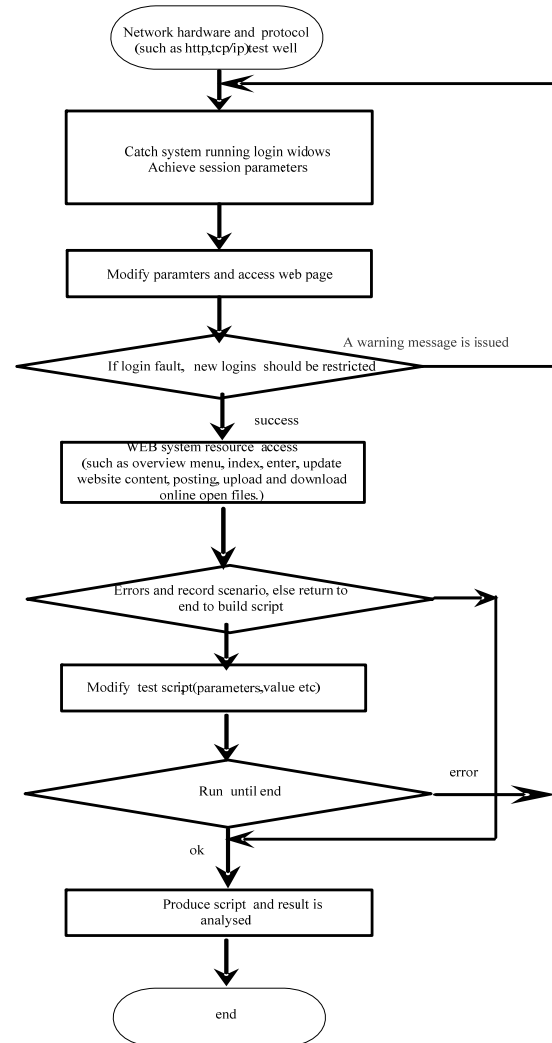


Figure 1. system login and access Test flow

## III. DATA CORRELATION AND MUTATION TEST OF SYSTEM

### A. Data Driver

With the advent of XML as an information ex-change standard, almost every corner of the software industry is inspired to share the once-incompatible information to the other XML enabled applications<sup>[10]</sup>. The choice of XML as the description mechanism makes the test tool suited to do test toward Internet-based system. You can view or modify the XML code implemented as a guest request, also test your XML pages by inserting the XML code as a

guest request. A very common method for using data driver is instructing virtual user (Vuser) to take values for parameters from an external file or a data table. The most popular data driver files include data file, excel and XML. For instance, management console of Exchange Server 2007 uses XML as data drive of test framework.

*B. Methods for Setting Test Data*

The test script is compiled together with software test drivers necessary to enable the execution of the test program by the test data generator. The test driver is used to assure system functions and parameters correction. Each step is unchanged only with data different. Methods for setting test data are listed as follows. (1)error assumption towards test data correction and data migration; (2) boundary analysis towards interface data overflow and numerical precision; (3) parameter type mutation(double and real exchange); (4)data process : test parameters of interface access can be divided into ① only once, ②unique, ③loop, ④random, ⑤ iteration etc. Setting methods of test data should be single or shared by the test script with same parameter errors.

We defined usage scenarios by assigning realistic values to the parameters that most directly affected performance. For most data types, you can customize a format for the parameter by selecting an existing format or specifying a new one. You should have a try to match the parameter format with the recorded values. If the format of the parameter differs from the format of the original recorded value, the script may run incorrectly.

*C. Role - Resource Correlating Model*

The key of efficiency of test driver data is to build correlation table between user roles and resources (such as cookie value, URL (you access the associated web page), web form, blogs, etc). Firstly, we should set up the correlation between system roles and their system resources; then set up their intersection correlation according to rights attributes (update, read, write, etc). If there are different roles, we can combine role rights; If there are sub systems, we can correlate sub system roles with resource. According to system requirement, the user-resource correlation model definition is given as follows.

**Definition 1** The Access Control Template Model is defined as ACTM ( S, R, Σ , A ) :

- 1) S is a a data set,  $S \in (admin, user, guest, \dots)$  , S can be defined as a role set;
- 2) R is system resource assemblage, which includes web manage system, bbs, post bar, virtual users groups and other internet resources;
- 3) Σ is the cartesian product of role resource access. It can also define the cartesian product of subsystem  $\sum n$  (n is the number of subsystem).

So a binary correlation function can be built between users and subsystem. From the way user x uses resources y, we define access test ordered pairs as follows:

ordered pairs  $\langle x,y \rangle$  ,  $x \in A$  (  $A = \{ x_1, x_2, x_3, \dots$  user set};  $y \in B$  (  $B = \{ B_1, B_2, B_3, \dots$  subsystem set ( $B_x$  include subsystem resources } ).

All possible ordered pairs formed from two sets are:  $A \times B = \{ A_1 \times B_1, A_2 \times B_2, A_n \times B_n, X$  is resource access control test policy. It defines the restriction to access attribute resource of different roles (object). So resource (subject) can be tested whether be accessed.  $x \in \{WRITE|READ|APPEND| PERMIT|DENY | \dots\}$ ;  
 $A_1 \times B_{2-n}, A_2 \times B_{1,3-n}, \dots A_n \times B_{1-(n-1)}$   $X \in \{ READ \}$

There exist some social problems concerning legal protection of users' privacy, which exert great impacts on our system implementation. OECD principles and privacy laws, such as COPPA, GLBA, and HIPAA [11], not only motivate the need to support system implementation of Internet-based system, but also limit some of application of the systems. Some access definitions influenced by these regulations are given as follows.

Regulation 1 challenge system (for example, system should consider a users including managers can not be allowed to overlook other's private information).

Regulation 2 trust levels and their measures should be set up according to legal restrictions and users' responsibility laid down by systems ( for example, subsystems should consider allowing others to overlook their own information, and trust can be tested in the sense of depth and scope to the degree laid down by the system.)  
 Regulation 3 all of these rules can become effective, such as legal regulations, rules laid down by access limitation of systems, for instance, time limitation, physical limitation, and so on.

Test model framework is shown in Fig.2. Taking web links test for examples, some tools including Web Link Validator, Xenu and so on, can only use default users to overall all links. We can define one kind of roles to access the links of a functions module. On the basis, we can determine whether they have right to access each other.

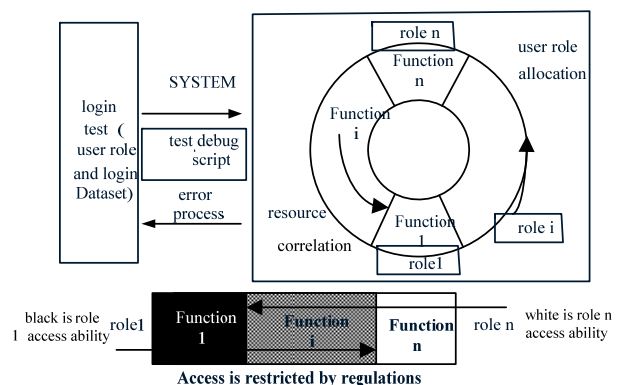


Figure 2. Test framework of system login access

*D. Access Control Combination Mutation Test*

To better manage access control, systems often explicitly specify access control policies by using policy languages such as P3P, EPAL, XACML and Ponder. In this paper, we propose a kind of mutation test method towards access control policy languages. Mutation

analysis is a testing technique, which is first designed to evaluate a test dataset. The original idea of mutation testing is proposed by DeMillo, Lipton and Sayward<sup>[13]</sup>. A mutant is the program modified by the injection of a single fault. In practice, faults are modeled as a set of mutation operators where each operator represents a class of software faults. To create a mutant of test script, it is sufficient to apply its associated mutation operator to the original script. We have provided a kind of three level mutation test of XACML towards component<sup>[12]</sup>, and we can also use regulation combination mutation algorithm and depend on safety policy management standard. Mutation Test case of multi-users and their combination (multi-roles access control (RBAC)) can be realized.

The rule combination of mutation logic of access control can be shown as follows:

**Definition 2** Whether transfer trust logic mutation can change object logic of access control is determined by rule combing algorithm of access control language. For example, XACML.

Now we begin to explain it: according to XACML Index of mutation operators<sup>[14]</sup>, we can use rule logic mutation as proposition, rule combination=condition1 × condition 2 × ... × condition n (× is rule combination logic), because rule mutation combination result=> policy target=> policy set target,

∴ Many Rule Combining Algorithms (RCA) exist, ((there are only three kinds of algorithms: (1) Deny-overrides, (2) Permit-overrides, (3) First-applicable) ∈ Change Rule Combining Algorithm (CRC) :

Algorithm 3: change of any first rule logic determines the result logic;

Algorithm 1: if rule mutation “||” leads to “Permit”, then decision policy will be “Permit”; If rule mutation “&&” leads to “Deny”, then decision policy will be “Deny”;

Algorithm 2: if rule mutation “&&” leads to “Permit”, then decision policy will be “Permit”; if rule mutation “||” leads to “Deny”, then decision policy will be “Deny”;

∴ Whether rule logic mutation can change object, is determined by actual rule combing algorithm. (rule combing mutation leads to error called “Indeterminate”; and we don’t consider the situation that rule can not be used to lead to “NotApplicable”). Definition 2 can be suitable for single user and single assignment (RBAC). If one user’s mutation object is able to consider a policy target, multi-user’s combination mutation object can be considered the policysset target, thus the definition suits for multi-user and multi-role combinatorial access multi-subsystems mutation test.

Additionally, the deductive method for rules is also able to apply to rule combination. For example, access right level can be ordered by quality restriction. When access broadband is extend out of requirement value, such as access time or access data flow (video or audio).

Sun xacml and xacml.net is the up-to-date access control realization technique. At present, sun has put forward XACML library (sun.xacml) and rw-XACML

based on Linux; On the other hand, Microsoft also put forward XACML library (XACML.NET). We have do test cases of different users under different operation systems. We will follow the development of XACML, and we will try to test and verify our regulations toward rule combination for considering performance and liability.

#### IV. GENERATION, ANALYSIS AND MANAGEMENT OF TEST CASE

##### A. WEB Systems Used to Test

This work is based on experience gathered while testing the performance, liability and security of different WEB software systems. One is a login system built by AJAX, it includes register and user management function. Its login encrypted password is show as follows.

验证码:  

Another is CNAS (China National Accreditation Service for Conformity Assessment) T0407 test cases. Its messages poster function is given as Fig.3.

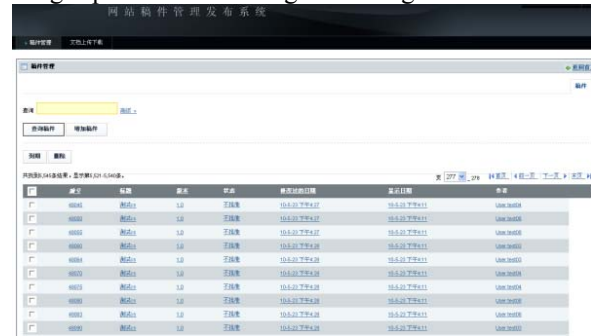


Figure 3. T0407 messages poster test

Test results show that the file uploaded and downloaded by 50 users, who are logging and posting messages at the same time. During this course, site server and database server work well.

Fig.4 is the picture of a role management of government E-Monitoring system. Different identifications are used to access different government departments. We outline our approach to present a case describing our experience of our access control and performance test for this system.



Figure 4. Subsystems access right management

##### B Test Analysis of Login and Access Control

During the process of login and access resources, system performance and security, which is a very important issue for users, can help us answer the

following more far-reaching questions relating to such issues as scalability(the ability of the system to handle significantly heavier workloads than are currently required),cost and bottlenecks<sup>[15]</sup>. With the details shown in Table 1.

TABLE I. TEST AND DIAGNOSIS TABLE

Test metrics	T/N or F /N cases	question	resolution
login	50/50	Fail message Interface fault injection (super-long string, heterogeneous inputs,boundary value)	Test report can set; defense of buffer overflow
Upload download	1000/153 25/20868 Faild/ stop 24/76	Disk operation errors; cocurrent process don't releasing memory	Online sessions (peak performance control) and OS package update
Post messages mails	310/670/ 818/ 1000	Memory Leak; Mail server no answer	Remain to be solved
On-line open Multi-media files	34/100	Network broadband is limit	hardware upgrading
Access links/Sql select	All web pages/50 0 ;	Web resoures lost;CPU percent too high	Set up resource config files ;using procedure and XML parser

P means test value each time

At the same time, Diagnostics Add-in installed on the controller of LoadRunner can be used to analyse the liability of system.

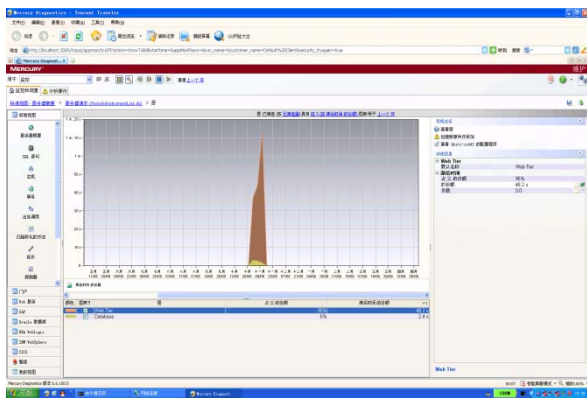


Figure 5. The diagnostics of web Tier and database

The Fig.5 is that diagnostics probe makes a diagnosis on the WEB system database. It shows Web Tier level has a very low efficiency, which costs system resource 95%. It is because of the low speed of DOM parsing XML documents, which is used to present the data of bean calling.

To assure service efficiency of reliability, fault recovery, such as database rolling back, may be used<sup>[16]</sup>. In my opinion, all kinds of the system functions are like the human's health. So we deal with the problem that should not only search errors from itself, but also find other things related with it. Taking memory capacity as example, hardware memory may not be big enough.

However we found that parallel users still occupy memory after they logged out. It is the real reason that leads to memory problems.

C Script Mutation and Test API calling

Test script for the login is run in sequence, and there is no branch. It is required to add the logic process so as to respond to error conditions after mutating script is built. When errors take place, the decision logic comes as follows.

- 1) Going to other test condition.
- 2) Invoking script that try to clean up error condition.
- 3) Dividing script and adding transferring parameters so as to realize module management of test template.

Error process can add verification points to follow, achieve, and verify data (for example, breaking point can be added to check the login system rather than use extra-code to verify the login).

On the basis of above techniques, most popular test tools all provide open APIs. API functions allow you to gather information about Vusers. For example, you can use Vuser functions to measure server performance, control server load, add debugging code, or retrieve runtime information about the Vusers' participating in the scenario. QTP VBS calls test script , connects out database so that the dll of login extra-code(picture or text) can developed by APIs. SOAP is an XML-based messaging protocol used in the exchange of Web services over a network. Diagnostics API can captures SOAP faults and classifies them into different types, based on their SOAP fault codes.

D Configuration Management of Test Template

Software test produces a lot of repeated test reports of requirement, performance, security and function. Thus all documents need to be managed by software configuration management (SCM) technique.

We use SCM tool, such as SVN (subversion), to realize management of the test documents at Google SVN Server. It is shown in Fig.6.

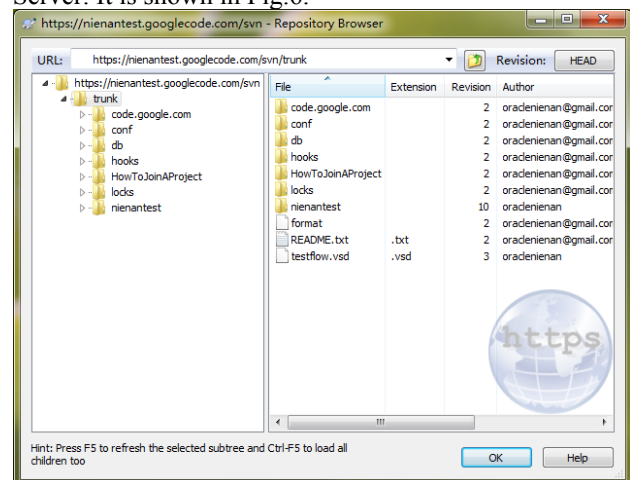


Figure 6. The repository of SVN Google server

The configuration relationship of these test documents can be represented by XML technique. For instance,

Test configuration template: : = **【test requirement】**<sup>m</sup> + **【test script】**<sup>n</sup> + **【test tool】**<sup>k</sup>

performance requirement: : = **【quality index】**<sup>m</sup> + **【function list】**<sup>n</sup> (m,n,k is the integer value)

The template can be defined by XML technique, such as schema file (scmlabel.xsd).

```
<?xml version="1.0" encoding="UTF-8" tandalone="yes"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
<!--Some elements are created by xmlespy and some values
are simplified -->

<xs:element name="author">
<xs:simpleType> <xs:restriction base="xs:string">
<xs:enumeration value="zhangshan"/><xs:enumeration
value="lisi"/>
</xs:restriction></xs:simpleType></xs:element>

<xs:element name="date"><xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="2010-10-1"/>
</xs:restriction></xs:simpleType></xs:element>
<xs:element name="scmlabelid">
<xs:simpleType> <xs:restriction base="xs:string">
<xs:enumeration value="c:\requirement and feedback
01.doc"/>
<xs:enumeration value="c:\requirement and feedback
02.doc"/>
</xs:restriction></xs:simpleType></xs:element>

<xs:element name="exchangedocument">
<xs:complexType> <xs:sequence>
<xs:element ref="testdocumentNo"
maxOccurs="unbounded"/>
</xs:sequence> </xs:complexType>
</xs:element>
<xs:element name="testdocumentNo">
<xs:complexType> <xs:sequence>
<xs:element ref="scmlabelid"/><xs:element
ref="author"/><xs:element ref="date"/>
</xs:sequence>
</xs:complexType></xs:element>
</xs:schema>
```

Then we can create xml document by using schema file.

```
<?xml version="1.0" encoding="UTF-8"?>
<exchangedocument
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="C:\Documents and
Settings\inn\桌面\xml测试模板\scmdocument.xsd">
<testdocumentNo>
<documentid>"C:\Documents and
Settings\inn\桌面\xml测试模板\requirement and feedback
01.doc"</documentid>
<author> zhangshan </author>
<date>2010-10-1</date>
</testdocumentNo>
<testdocumentNo>
<documentid>"C:\Documents and
Settings\inn\桌面\xml测试模板\requirement and feedback
01.doc"</documentid>
<author> lisi </author>
<date>2010-10-1</date>
</testdocumentNo>
```

These documents can be managed by repository browser. Especially, all non-binary system files can be browsed by calling the software tools, which is useful for (1) resource share; (2) remote control; (3) information integration.

## V. CONCLUSION AND FUTURE WORKS

It is a systematic engineering how to evaluate Internet-based system software quality objectively. This article attempts to provide a test template for the problem of performance, security, and liability in Internet-based system. It builds the test data driver, which is the individual XML data. The approach makes coupling degree between test-scripts. Thus, the difficulty of creating and maintaining test-scripts is diminished.

According to some automatic test tools and its related technique, we have described our experience of doing liability, security and performance testing, such as CNAS T0407 test cases. It can test software security and performance towards system login and access control by accessing system resources. The system can become more liable after overall test and diagnostics. The results show that software quality can be improved after the comprehensive and justice evaluation result is counted up.

In my opinion, our next work will be: How to use application program to invoke test tools interface to test system problems; How to test the degree of match between law rules and access control regulations.

## ACKNOWLEDGMENT

The authors wish to thank Zhansheng Feng, Yufeng Guo and Yaojun Feng. This work was supported by the Innovation Fund for Small Technology-based Firms of China ( 08C26244202133 ) ; the Science and Technology Plan of Henan Province (No.112102210322) ; the Natural Science Foundation of Henan Province (No.2010A520039) ; Zheng Zhou University of Light Industry Ph.D Foundation (200903).

## REFERENCES

- [1]. Barbara Carminati, Elena Ferrari, "Privacy-Aware Collaborative Access Control in Web-Based Social Networks," ACM Transactions on Information & System Security, 13(1):pp.6-38, 2009.
- [2]. Hooshmand Alipour, Ayaz Isazadeh, "Software Reliability Prediction Based on a Formal Requirements Specification," In: Lecture Notes in Computer Science - CSICC 2008 - International Conference, Berlin: Springer, 2008, pp. 816-820.
- [3]. ANSI/IEEE, Standard Glossary of Software Engineering Terminology, STD-729-1991, 1991.
- [4]. Balsamo, S., Di Marco, A., Inverardi, P., "Model-based performance prediction in software development A survey," IEEE Trans Software Engineering 30(5), pp.295-310, 2004.
- [5]. Avritzer, Alberto1; Weyuker, Elaine J., "The role of modeling in the performance testing of e-commerce applications," IEEE Transactions on Software Engineering, v 30, n 12, pp.1072-1083, December 2004.

- [6]. IBM, IBM Internet Security Systems X-Force® 2008 Trend & Risk Report, 2009.
- [7]. Zhang Nan, Mark D. Ryan, Dimitar P. Guelev, "Synthesising Verified Access Control Systems through Model Checking", Journal of Computer Security, IOS Press Amsterdam, The Netherlands, 2008, 16(1):pp.1-61.
- [8]. Alipour, H., Isazadeh, A., "Software Reliability Assessment Based on a Formal Requirements Specification". 2008 International Conference on Human System Interactions, Publisher:IEEE Computer Society, 2008: pp.311-316.
- [9]. C. Del Grosso, G. Antonioli, E. Merlo, P. Galinier, "Detecting buffer overflow via automatic test input data generation", Computers and Operational Research 35(10), pp.3125-3143, 2008.
- [10]. Xu, Wuzhi, Offutt, Jeff, Luo, Juan, "Testing Web Services by XML Perturbation", Source: Proceedings the 16th IEEE International Symposium on Software Reliability Engineering, IEEE Computer Society Washington, DC, USA, 2005, pp. 257-266.
- [11]. Qun Ni, E. BERTINO, "Privacy-aware role-based access control", ACM Transactions on Information and System Security (TISSEC), 13(3):pp.24-55, 2010.
- [12]. Nie Nan, Xia Qiming, Yao Junfeng, "Component Interface - Oriented XACML Mutation Testing Policies", Journal of University of Electronic Science and Technology of China. 2008,38(2):pp.288-292.
- [13]. Delamaro M E, Maldonado J C, Mat hur A P. "Integration Testing Using Interface Mutations", proceedings of International Symposium on Software Reliability Engineering (ISSRE' 96), : IEEE Computer Society Press, 1996, pp.112-121.
- [14]. E. Martin, Tao. Xie, "A fault model and mutation testing of access control policies" In Proc. 11th International Conference on World Wide Web, ACM New York, 2007, 667 - 676.
- [15]. T. Illes, A. Herrmann, B. Paech, J. Rückert., "Criteria for Software Testing Tool Evaluation - A Task Oriented View," Proceedings of the 3rd World Congress for Software Quality 2005, Vol. 2, pp. 213-222.
- [16]. Scully, J.K., "The Role of System Behavior in Diagnostic Performance," Northrop Grumman Corp., El Segundo, CA in: Systems Conference, 2008 2nd Annual IEEE Publication Date: 7-10 April 2008, pp. 1 - 7.



**Nie Nan** was born in P.R.China in 1973, earned M.S. degree in the field of computer application in 2001 from Huazhong University of Science and Technology, Wuhan city, P.R.China, earned Ph.D. in the field of computer software and theory in 2008 from Huazhong University of Science and Technology, Wuhan city, P.R.China.

At the present time, he takes part in many research projects in the area of software test supported by the Science and Technology Plan of Henan Province (No.112102210322); the Natural Science Foundation of Henan Province (2010A520039); Zheng Zhou University of Light Industry Ph.D Foundation (200903) respectively. His current research interests include software engineering.

Nie Nan is now a vice professor at Zhengzhou University of Light Industry. Recently he has published 14 research papers and 2 books in the area of computer science and technique.



**Zhang Suzhi** was born in P.R.China in 1965, earned M.S. degree in the field of computer application in 1995 from Southeast University, Nanjing city, P.R.China, earned Ph.D. in the field of computer application in 2004 from Huazhong University of Science and Technology, Wuhan city, P.R.China. His current research interests include database and intelligence computing.

Zhang Suzhi is now a professor at Zhengzhou University of Light Industry. Recently he has published more than 30 research papers and several books in the area of computer application and technique.

**Fujin. Hua** was born in P.R.China in 1980. He is working as a lecturer at Zhengzhou University of Light Industry. Recently he has published several research papers in the area of computer science and technique. His areas of interest include database and software engineering.