IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 19, Issue 2, Ver. I (Mar.-Apr. 2017), PP 48-52
www.iosrjournals.org

Rediscover Web Application Testing

Rajiv Chopra
Assistant Professor (CSE/IT) (Dept. of Computer Science/IT,GTBIT, GGSIPU DELHI)

Abstract: *Web applications are very complex in terms of its design and its code. The dynamic aspects of web applications makes website testing quite challenging. A web application consists of web page as its basic unit. This makes it a page-based web application. This paper focuses on these types of web applications. Based on the web application's navigational graph we extend the MM-path based testing to test page-based web applications. Different test paths are thus obtained.*

Keywords: *Extended MM-path graph, n-tier, Page-based testing Web page,*

I. Introduction

Web applications are developed rapidly that affects current businesses. As the number of web applications are increasing, their functions and architecture becomes more complex [3]. Presently, the web applications follow an n-tier architecture with n=3. By n=3, we mean that there are three-tiers (or layers) as client-side (layer-1) with web browsers like Mozilla, Firefox, IE7, Chrome and server-side (layer/tier-2) with web-server software like IIS, PWS, NT. They have back-end like Oracle, MS-SQL SERVER, PARADOX, SYBASE, installed at the back-end. The third layer happens to be the business logic layer (or third tier) where all computations related to the business are done. This forms complete 3-tier web architecture. Internet is the best example for this. Although this layering simplifies the task of a web engineer but at the same time it complicates the process of website testing.

The basic unit of any web application is a web page (or simply a page). And users navigate through these websites, page by page. This web graph showing navigations can be drawn either manually or more easily with the help of CASE TOOLS like VeriWeb. This paper will focus on page-based web applications and will apply traditional MM-path based approach to test web applications. Also it will extend the existing MM-path based approach for page-based website testing. Several test cases and hence test suites may be obtained from this approach.MM-path graph to PP-path graph approach is better as MM-path graph has a better ability to test GUIs and also has a complete testing ability.

II. Related Work

In [1], Chopra Rajiv et. al develop P^3R^2 model to test static as well as dynamic websites. But no integration tests are performed. In [3], Mc Call et. al relate testing and security with each other. However, they do not consider web page under integration testing. In [8], Jingxian Gu, Lei Xu, Baowen Xu, Hongji Yang, propose an extended MM-path approach to component-based Web Application Testing. MM here refers to module-to-module. But it does not take into consideration the basic unit of any website i.e. a web page. In [9], Alejandra Garrido and Gustavo Rossi perform refactoring of Web Applications to increase its usability. But it does not take into account the integration testing part of the websites and web applications. As functional (black box) and structural (white box) testing is preferable at unit-level of testing, similarly integration and system testing is needed [10]. Co-functioning testing (i.e. interactions among these units) is needed. MM-paths are a hybrid between Black box and white box testing. It is black box as it represents actions with inputs and outputs. White box as it focuses on how they are identified i.e. MM-path graph. The point of reference is that the cross-check of black box and white box approaches should be consolidated into the constructs for path-based integration testing.

Path-based integration testing works equally well—any process model may be followed. Path-based integration testing is closely coupled with the actual system behaviour instead of structural motivations of decomposition and call-graph based integration. Although more effort is needed to develop and identify MM-paths yet it eliminates the need of stub and driver development. The set of MM-paths should cover all source-to-sink paths in the set of units. When loops are present, condensation graphs' will reduce result in directed acyclic graphs. This resolves the problem of path-explosion too. *MM-paths in traditional Software* – We use "message" and Module-Execution-Paths (MEPs) instead of full modules [10]. *MM-paths in Object-Oriented Software-* Here, "methods" may have several internal execution paths. An MM-path starts with a method and ends when it reaches a method that does not issue any messages of its own. This is the point of message quiescence.
This paper focuses on extended MM-paths on web pages of websites and web applications.

III. Proposed Work

When a web user (on client-side) sends a http request to the web server, the web server returns back the page as per users request. Web pages and its components are related. Some basic terminology is defined for PP-based path testing as follows:-

a) **Page Fragment:** It refers to one web page in a navigational graph (definition-1).
b) **Source Node:** It refers to the first/start page from which the navigation starts (definition-2).
c) **Sink Node:** It refers to the last/exit/end page at which the navigation ends (definition-3).
d) **Page Execution Path (PEP):** It is the sequence of web pages that begins with a source node (a web page) and ends with a sink node, with no intervening sink nodes (definition-4).
e) **Message:** A message in a web page refers to the mechanism by which one page transfers data to another page (definition-5).
f) **Page-to-Page Path (PP-Path):** A PP-path is an interleaved sequence of Page-Execution-Paths (PEP) and messages (definition-6).
g) **Page-to-Page Path Graph (PP-Path Graph):** Given a web graph, its PP-path graph is a directed graph in which nodes (or pages) are page execution paths and edges correspond to messages and returns from one page to another. So, the page graphs will now have multiple sources and sink nodes (pages). The proposed work describes sequences of page execution paths. We can find PP-paths in a web flow graph in which nodes are page-execution paths and edges are messages.

Consider a hypothetical website as shown in figure-1.

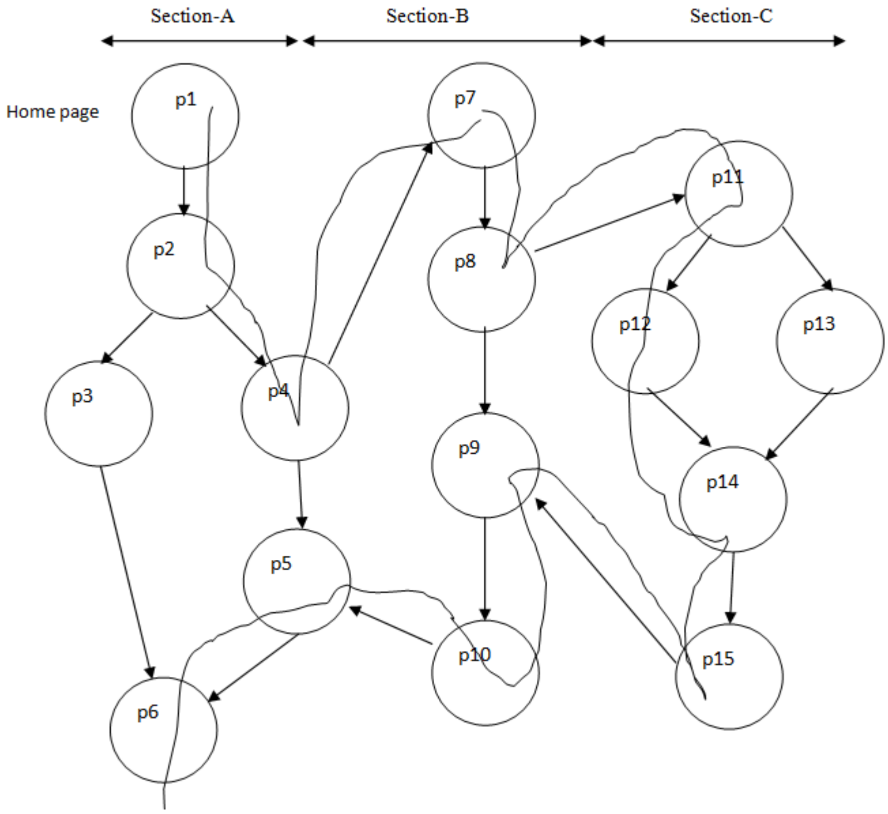

Figure 1: A WAG for hypothetical Website

Here, PP-paths begin and end in the main/home/login page only.
In section-A, node-1 (i.e. p1) and node-5 (i.e. p5) are source nodes and node-4 (i.e. p4) and node-6 (i.e.p6) are sink nodes.
Similarly, in section-B, node-7 (i.e. p7) and node-10 (i.e. p10) are sink nodes. Section-C has single source node, p11 and a single sink node, p15. That is,

Table 1: Source and Sink Nodes of Graph

Section	Source-Node	Sink-Node
A	p1, p5	p4, p6
B	p7, p9	p8, p10
C	p11	p15

So, 7 PEP exists. And these 7 paths are as follows:-
PEP (A, 1) = <p1, p2, p3, p6>
PEP (A, 2) = <p1, p2, p4>
PEP (A, 3) = <p5, p6>
Similarly, other paths are as follows:-
PEP (B, 1) = <p7, p8>
PEP (B, 2) = <p9, p10>
PEP (C, 1) = <p11, p12, p14, p15>
PEP (C, 2) = <p11, p13, p14, p15>
These are the seven PEPs.
Thus, its page-graph based, PP-Path graph is as follows:-

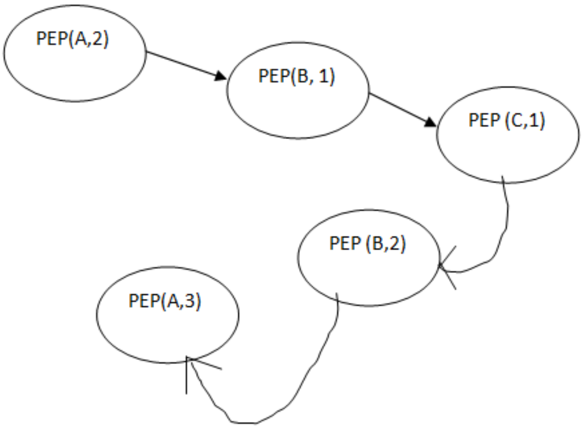

Figure 2: PP-path Graph for graph of figure-1

Thus, these pages in a website or a web application must be tested thoroughly.
Some basic extended MM-path based, PP-based definitions as follows:-

a) **Source Node/ Page:** The source node of extended PP-path is at the beginning of the page in the web server. The first executable page of a server is a source node. A source node (page) also appears from the node from which the control transfers to other pages.
b) **Destination Node/Page:** The destination node of extended PP-path is at the end of the page in the web server. The last executable page of a server is a destination node. A destination node (page) also appears from the node from which the control transfers to other server pages.
c) **Executable Paths:** They are the sequence of web pages from source node (page) to the destination node (page).
d) **Messages:** In extended, PP-path approach, it refers to the redirection and page call/returns.
e) **Extended PP-paths:** It is the sequence that execution paths and messages appear alternately.
Thus, seven extended PP-paths that are found for the hypothetical graph of figure-1,can now be used for testing of website and web application under test.

IV. Conclusions

In this paper, we have applied the traditional PP-path based integration testing for page-to-page based path testing. Then, extended PP-path approach is used to find the test paths in a web navigational graph.

Based on the literature survey and recent cyber losses, in this paper fifteen (15) laws of cyber security conservation are stated as follows:-

Law-1: Whatever may be the phase of SDLC (Software Development Life Cycle), Security Engineering must be applied throughout the life cycle of website development.
Law-2: Security flaws propagate (amplify) if they are not identified there itself.
Law-3: Security must be measurable (estimates will be a good start).
Law-4: Security is NOT dependent on Lines of Codes (LOC) of the software. Even small software may not be secure.
Law-5: Number of software developed or purchased by an organization and their security measures are directly proportional.
Law-6: Website or web application complexity, web testing and web security are all inter-related. Lesser is the complexity of the website code, easier is to test it and it is more secure.
Law-7: It is impossible to achieve 0% risk or 100% security.

Law-8: Threats to website security are converging but the whole of cyber threat space is diverging.
Law-9: Rate of digital theft growth appears exponential.
Law-10: Effective security depends on a strong partnership between Senior Management, the corporate security team and the employees using the systems.
Law-11: Information security begins with you.
Law-12: Infections spread because people don't take precautions to protect themselves. Cyber infections spread the same way. Don't let it happen to you.
Law-13: You could spend a fortune purchasing technology and services and your network infrastructure could still remain vulnerable to old-fashioned social engineering manipulations.
Law-14: Be proactive about security. The data you protect may be your own.
Law-15: The security of the system is only as strong as its weakest link.

From law-1, it is stated that security must be inserted in every phase of SDLC i.e. security during requirements analysis, security during design, security during coding, security during testing and security during maintenance.
From law-2, curb the security vulnerable points at the beginning of the project itself so that they do not amplify themselves during later SDLC phases.
From law-3, as Kelvin states that if you can measure what you say then it is better. Based on similar lines security of software must also be measureable.
From law-4, a C program may not be secure but a COBOL program may be safe even if C program is smaller than a COBOL program.
From law-5, buy software only which has some security measures too.
From law-6, it is possible to kill three birds with one stone only i.e. if web design is simple, then its complexity is low and if its complexity is low then it is easier to test it and hence better is its security. Thus, web design, web complexity, web testing and web security have symbiotic associations between them.
From law-7, it may not be possible to achieve 100% security in the code but it can be mitigated (reduced).
From law-8, researchers are trying hard to reduce security risks but the cyber space is diverging/ expanding every day.
From law-9, due to exponential rise in number of Internet users, the rate of digital data theft has also increased exponentially.
From law-10, security is everybody's responsibility in an organization. Only the CEO or MD of a company cannot retain web security within an organization.
From law-11, by giving secure and meaningful passwords, proper logins and logouts, keeping devices safe etc. can safe your system resources and increase web security.
From law-12, prevention is better than cure. So, prevent your organization rather than fixing security problems later on.
From law-13, don't tell your password to any one, your social security numbers (SSNs) or Master card numbers to any one as they may reuse them to steal your accounts.
From law-14, before any activity always focus on its security aspects too. Using keyboard hackers cause havoc, more than that is caused by bombs and bullets.
From law-15, by imposing strict security constraints, new vulnerabilities should not be introduced.
Our future work will focus on the following studies-

a) Generating test cases both manually and with automated tools.
b) Developing an efficient algorithm for it.
c) Developing a test tool for it.
d) Applying it to both static and dynamic websites as well as to static and dynamic web applications.
e) Performing both black box and white box testing using this approach on websites.
f) Extending this to DD-page graph based testing of websites.
g) Formulating new laws of cyber security conservation.

There is no single yardstick to measure web software security. But the guidelines and the laws of cyber security conservation listed in this paper are a way of improving web security and hence cyber security.

References

[1]. R. Chopra. Testing Web Applications: The State of Art and Future Trends.*(*UK: Cambridge International Science Publishing, 2016).
[2]. http://www.owasp.org/documentation/topten.html
[3]. Mc Cabe et al. Using Cyclomatic Path Analysis to Detect Security Vulnerabilities, 2011.
[4]. Hamzeh Al Shaar and Ramzi Haraty, Modeling and Automated Black Box Regression Testing of Web Applications, Journal of Theoretical and Applied Information Technology, 1182-1198.
[5]. Danny Roest, Ali Mesbah and Arie van Deursen, Regression Testing Ajax Applications: Coping with Dynamism, 2010, pp 127-136..
[6]. Rajiv Chopra, Sushila Madan, Analysis and Security Testing of Websites Using P^3R^2 Model, Cyber Times International Journal of Technology & Management (CTIJTM), Volume 5, No. 1, Oct. 2011-March 2012, 1-18.

[7]. Rajiv Chopra and Sushila Madan, Testing Websites using P^3R^2Model, International Journal of Computer Science Issues, IJCSI, Vol-9, Issue- 4, July 2012, 248-253.

[8]. Jingxian Gu, Lei Xu, Baowen Xu, Hongji Yang , An Extended MM-Path Graph Approach to Component-based Web Application Testing, 12th IEEE International Workshop, 2008, pp 144-150.

[9]. Alejandra Garrido, Gustavo Rossi, Refactoring for Usability in Web Applications, IEEE Computer Society, 2011, pp 60-67.

[10]. Jorgensen P., “Software Testing- A Craftsman Approach”, (CRC Press, 4th Edition 2016).