Risk Evaluation Model Approach

Vinod Sapkal
Department Of Information Technology,
PVPPCOE Sion Mumbai,

Abstract : One of them is "web mashup" which allows users to create new web applications by integrating data and services from other various web applications and data sources.. At the same time, some new technologies and social challenges reveal themselves sequentially. This paper researches on mashup application from a perspective of security, discusses how to assess the risks of mashup application, and puts forward a risk evaluation model and some possible metrics for the actual assessment.

IndexTerms—Group mashup; risk assessment; evaluation model

I. INTRODUCTION

The term "mashup" has been originated from the music field where it means that producing/composing a new song by mixing lyrics and background music from several different source songs [1]. A number of new techniques for the creation of web applications have been resulted into the rapid growth of Web 2.0; one of these techniques is meshing up of required content/services from several independent sources for the purpose to create a new content/service [3]. Web Mashups started as a result of security violation of hole in Google Maps web application by a hacker named Paul Rademacher, but instead of taking him to the law, Google offered him a job, documented his efforts and announced it as an API [4]. A web mashup can be defined as a situational web application or web page that is created by pulling out,
Parsing, and aggregating required relevant content/service from diverse and different publicly available web applications to satisfy user needs and tasks through displaying it to them on their computer screens in a completely different and new way Skilled web users, who develop their own applications online, so-called mashup applications, are a reality of today's Web. Mashups are simple web applications (most of the times even consisting of only one page) that result from the integration of content, presentation, and application functionality stemming from disparate web sources. [3] Like any other data integration domain, mashup development is replete with technical challenges that need to be addressed, especially as mashup applications become more feature and functionality-rich.

II. DECONSTRUCTING MASHUP APPLICATION

Mashup Application

A mashup application includes several mash-pages. Each mash-page is divided into several rectangular regions, where the components used to display. The Web pages, which contains a number of components called "Part-Page" by Microsoft. The part-page, which contains the external components called mash-page. The site contains mash-page called mash website. To facilitate data retrieval, the content providers often expose their content through Web-protocols such as REST, Web Services, and RSS/Atom. Mashups that extract content from sites like Wikipedia, TV Guide, and virtually all government and public domain Web sites do so by a technique known as screen scraping. In this context, screen scraping connotes the process by which a tool attempts to extract information from the content provider by attempting to parse the provider's Web pages, which are originally intended for human consumption.
concerned with the movement patterns of group of animals. The details are given as input data.

Mashup Site

A Web site publishes data about APIs (links to the URLs for API download, descriptions, comments, etc.) and their use within mashups. This is where the mashup is hosted. Interestingly enough, just because this is where the mashup logic resides, it is not necessarily where it is executed. On one hand, mashups can be implemented similarly to traditional Web applications using Server-side dynamic content generation technologies like Java Servlets, CGI, PHP or ASP. The mashup site can use data through three ways like API, Web feeds, or screen scraping.

III. REFERENCE RISK EVALUATION MODEL FOR MASHUP APPLICATION

A. Quality Assessment for Mashup Components

Computing, the term composite application expresses a perspective of software engineering that defines an application built by combining multiple existing functions into a new application. The technical concept can be compared to mashups. However, composite applications use business sources (e.g., existing modules or even Web services) of information, while mashups usually rely on web-based, and often free, sources.

B. Behavioral Model for Mashup Application

By definition, an information system based web is composed of components, which use to collect, process, memorize, develop and deploy information. A web information system consists primarily of functional components described in terms of their behaviors and interfaces and component-component interconnections. It is necessary to describe the behavior model before the security risk model is constructed. In "Fig. 3", we illustrate the behavioral model, which can be divided into three layers. At the application layer, a mashup component can provide several APIs developed by using different technologies. [5]In order to implement the goal, a series of applications will be executed. At the task layer, a series of tasks will be implemented. To finish the task, components and component-component interconnections are used. Considering the aforementioned factors, three assumptions are been put forward.

Assumption 1: There is only connection between different tasks. This paper first studied this model, then follow that can be extended to.

Assumption 2: We assume that the state of controlled component has known. The next activities have relation to the previous are no relation to the other past events.

Assumption 3: The risk between component and interconnection are independent. When one component has a problem, other components and interconnections are not affected. [6]

C. Risk Evaluation Model for Mashup Application

Based behavioral model, a three-layer risk assessment framework was established. Evaluation results are divided into four levels, including information-class, alert-class, urgent-class and desperate-class. [7] Each task contains two kinds of risks: (1) the risks of component itself, (2) the risk of the component-component interconnection. According to the above ideas, each task signs the overall risk assessment value as R(N), where R(N) consists of two parts: (1) the risk of component itself sign as E(N), (2) risk evaluation of the connecting path sign as P(N) when component connects forward. Thus, the risk of a task X can be defined as:
risk factor while risk happened on the connection between Ci and Cj. Given task X, the markov chain of information system includes safety state n and one terminated status. Task X have k+1 transient states, the figure of components include dis k, and therefore, the probability of transfer matrix between components is k + 1. The transfer matrix of task X signs as R.

$$R = \begin{pmatrix} R_{00} & \cdots & R_{0k} \\ \vdots & \ddots & \vdots \\ R_{k0} & \cdots & R_{kk} \end{pmatrix}$$

Transition probability matrix P can be defined as:

$$P = \left[p_{ij} \right]$$

According to reasoning and operation, the risk evaluation system can be established under different state, the risk for mashup application from different task is cumulative.

IV. CONCLUSION

Mashups are certainly an exciting new genre of Web applications. However, the mashup application genre is still in its infancy, with hobbyist developers who produce many mashups in their spare time. These developers might not be cognizant of (or concerned with) issues such as security. Additionally, content providers are only beginning to see the value in providing APIs for machine-based content access, and many do not consider them a core business focus. This combination can yield poor software quality. It will be interesting to understand how to assess the risk of a mashup application and how to develop high-security applications, in order to facilitate mature software development processes. In this paper, we research on mashup application development from a security perspective, that is, from the perspective of the mashup composer and componentcomponent interconnection, and we discusses how to assess the risks of mashup application, and puts forward a risk evaluation model and some possible

REFERENCES

[1] Shukat Ali, Shah Khusro and Azhar Rauf, A Cryptography-Based Approach to Web Mashup Security, IEEE, 2011.
[2] Saman Zarandioon, Danfeng (Daphne) and Yao Vinod Ganapathy, OMOS: A Framework for Secure Communication in Mashup Applications, 2008 Annual Computer Security Applications Conference
[3] Chen Yanchun, Wang Xingpeng, A Security Risk Evaluation Model for Mashup Application, 2009 International Conference on Information Management, Innovation Management and Industrial Engineering
[4] Gerald Bader Amin Anjomshoaa and A Min Tjoa, Privacy Aspects of Mashup Architecture, IEEE International Conference on Social Computing / IEEE International Conference on Privacy, Security, Risk and Trust, 2010
[5] http://en.wikipedia.org/wiki/Mashup_%28web_application_hybrid%29
[6] Duane Merrill, "Mashups: The new breed of Web app," retrieved from http://www.ibm.com/developerworks/xml/library/xmashups. html, updated 16 Oct. 2006.
[7] Ravi Ganesan, Erhan J. Kartaltepe, and Todd Wolff, "Mashups, Risks and MashSSLTM," retrieved from https://www.safemashups.com/home/public_html/education.html.
[8] Florian Daniel, Maristella Matera, "TurningWeb Applications into Mashup Components: Issues, Models, and Solutions," Proc. International Conference (ICWE 09), Jun. 2009, pp. 45-60. Chain," Journal of the China railway society , 29 (2), Apr. 2007, pp. 50-53