Coordination and organization of security software process for power information application environment

Wang Qiang
Central China Branch of State Grid Corporatio of China
e-mail:882148@qq.com

Abstract. As an important part of software engineering, the software process decides the success or failure of software product. The design and development feature of security software process is discussed, so is the necessity and the present significance of using such process. Coordinating the function software, the process for security software and its testing are deeply discussed. The process includes requirement analysis, design, coding, debug and testing, submission and maintenance. In each process, the paper proposed the subprocesses to support software security. As an example, the paper introduces the above process into the power information platform.

1. Introduction
According to statistic from the USA, many web business looks at security assurance control as most important characteristic of web development[1]. Some business established department to build safe software. Even some software business, such as qihoo, majors in safe software when it established. Future trend on software and IT adoption shows that for a business to be a pacesetter and becoming successful, business has to establish its security and privacy policy ahead of their peers[2]. Through these statistical analysis reports, it shows that security issue in software development environment is very important.

There are various types of model recommended in the software reliability practices including the estimation model by Littlewood-Verall Model[3], s-shape reliability model[4], Duane’s Model[5], and the Jelinski/Moranda reliability growth model. These are the few models based on the statistical Modeling and Estimation of Reliability Functions for Software[8].

Because a lot of users have increased their dependency on web to operate daily business process[6]. Having web security, it is able to address the secure issues of web which include the functionality, usability, capability, and reliability[7]. Without software security, there are high probability software bugs, which will impair high visibility programs, thus lead to business loss[8].

2. Security Engineering Cooperation in Power Information Application Environment
Similar to any other software development environment, development of the Power Information application has to ensure its security. In order to achieve the reliability of this application, the software developer has to be involved in reliability assessment and issues as early as the application is developed. It is has to start with good planning.

The security of the power information application environment ensures that safety engineering is an integral part of the whole project. It is impossible to succeed in isolation because of safety engineering, which includes open communication with all related projects[9].

1) Define the collaborative goals and relationships between the security team and other
working groups, define the agenda, objectives, and action topics.

2) To ensure the coordination of safety engineering and other related projects to ensure the standardization of communication plans, conference reports, messages, memos, texts, decisions and recommendations.

3) Promote the implementation of safety engineering and other related projects to determine the effective ways to resolve conflicts among different engineering entities.

4) Use standard validation mechanisms to work with other security related projects.

3. The Power Information Development Process Considering Coordination and Organization

For a normal web development, usually we first decide its functions, the web switching and the data flow. Considering the security for the web, it is necessary to find out the software security. As a security engineering, it is also necessary to design the security testing module to test the web security which will defend the attack when it runs in the suitable environment. So the coordination for the power information development becomes important and significant.

3.1 Requirement Analysis Coordination and Organization

Table 1. Requirement analysis

| Items          | Function requirement analysis | Security requirement analysis | Security testing requirement analysis |
|----------------|-------------------------------|-------------------------------|---------------------------------------|
| Participants   | Customers; Project leaders; Function analysis group; | Customers; Project leaders; Security analysis group; Function analysis group; | Project leaders; Security analysis group; Security testing analysis group; |
| Entrance Conditions | The Project is authorized. | The function analysis is proposed. | The security requirement analysis is proposed. |
| Exit Conditions   | The function requirement document is signed by the customer. | The security requirement document is signed by the customer and the function analysis group. | The security testing requirement document is signed by the customer and the security analysis group. |
| Input          | Project task document; | Project function requirement document. | Project security testing requirement document. |
| Output         | Requirement document for function analysis. | Requirement document for security analysis. | Security testing document for security testing analysis. |

The aim of requirement analysis is to describe the software product from the analysis of customers, coproducing the documentation for the software requirement analysis. In secure web software development, requirement analysis not only includes the requirement analysis includes user software requirement analysis, but includes security software requirement analysis. Security analysis is very important for such development, which is often ignored by the software analyst even the programmers. The final result will lead to wasting much time for the exclusion of fault. Following the security requirement analysis, it is necessary to go on the security testing requirement analysis, which is done based on the security requirement analysis.

The dependency of each requirement should be known and the coordination of these requirements should be done based on the dependency. As is shown in table 1, the requirement analysis comparison items include participants, entrance conditions, exit conditions, input and output. From each item, the comparison among the function analysis, the security analysis and the security testing analysis can be clearly described.

The participants should include the system analyst not only the function requirement analysts, but also the security requirement analysts and the security testing requirement analysts. The first step is that the function requirement analysts obtain the demand from the customers. Because of the major knowledge limitation of the customers, the security requirement analysis should be proposed by the
security analysts, who can help the customer to propose his security requirement. And then the security testing requirement analysis should be proposed by security testing analysis.

3.2 Summary Design Management
The aim of summary design is to map the content from requirement document to a unified flow in logic. As a customer software developer, it is necessary to form a software system including software functions, security and its security testing. At the same time, the software security designer may propose his thinking, which will enrich to the software design, for the project. For some minor software, it is not usual to do summary design.

| Items                  | Function summary design | Security summary design | Security testing summary design |
|------------------------|-------------------------|-------------------------|---------------------------------|
| Participants           | Project leaders;        | Project leaders;        | Project leaders;                |
|                        | Software function design| Software function design| Software security design        |
|                        | group; Function analysis| group; Security design  | group; Security testing design  |
|                        | group; Function analysis| group; Security design  | group; Security testing design  |
| Entrance Conditions    | Software function       | Software security       | Software security testing       |
|                        | requirement document    | requirement analysis is | requirement analysis is         |
|                        | qualified;              | qualified;              | qualified;                      |
|                        |                         | Software function       | Software security               |
|                        |                         | summary design document | summary design document qualified;|
|                        |                         | qualified;              |                                 |
| Exit Conditions        | Summary function design | Summary security design | Summary security testing        |
|                        | document qualified;     | document qualified;     | document qualified;             |
|                        |                         |                         |                                 |
| Input                  | Software function       | Software security       | Software security testing       |
|                        | requirement document;   | requirement document;   | requirement document;           |
|                        |                         | Software function       | Software security               |
|                        |                         | summary design document | summary design document;        |
|                        |                         | qualified;              |                                 |
| Output                 | Software function       | Software security       | Software security testing       |
|                        | summary design document | summary design document | summary design document;        |

Deeply analyzing the dependency, the coordination and the organization is shown in table 2. Comparing the entrance condition item among summary designs, the exit conditions of security summary design should consider the function summary design. And the input items among these designs also considers the function summary. It is concluded that security summary design depends on the function summary design.

3.3 Detail Design Management
Detail design is to divide the functions into modules or functions. And it provides the flow corresponding chart. When it comes to the detail design, it is necessary to describe the modules or functions in detail. It is also necessary to make out a plan for the security unit. For security testing, it is necessary to make out a plan for the testing unit which can be a tool, a software downloaded, or a program to be coded. However, function detail design should be made out so that it will guarantee that the security detail design, which is according to the function detail, works well. And security testing detail design should also be made out so that it will guarantee that the security. It is important to list all the security testing modules clearly so that you can guarantee the security of the software design.
The dependency of each detail design should be known and the coordination of these designs should be done based on the dependency. The detailed designed is summarized in table 3.

Table 3. Detail Design

| Items       | Function detail design | Security detail design | Security testing detail design |
|-------------|------------------------|------------------------|--------------------------------|
| Participants| Project leaders;       | Project leaders;       | Project leaders;               |
|             | Software function      | Software function      | Software security design       |
|             | design group;          | design group;          | group;                         |
|             | Security design group; | Security design group; | Security testing design group; |
| Entrance    | Software function      | Software security      | Software security testing      |
| Conditions  | summary design         | summary design         | summary design document is     |
|             | document qualified;    | design document is     | qualified;                     |
|             |                        | qualified;             |                                |
| Exit        | Detail function        | Detail security design | Detail security testing         |
| Conditions  | design document        | design document qualified; | design document qualified;     |
|             | qualified;             |                         |                                |
| Input       | Software function      | Software security      | Software security testing       |
|             | summary design         | summary design         | summary design document        |
|             | document;              | document;              | is qualified;                  |
|             |                        |                        |                                |
| Output      | Software function      | Software security      | Software security testing       |
|             | detail design          | detail design          | detail design document         |
|             | document;              | document;              | is qualified;                  |

3.4 Coding Management
Having finished detail design, the project enters into coding stage. During this stage, it is not necessary to divide into function coding and security coding. Otherwise, the integrity will be increased. However, it is also necessary to code the security testing program, which simulates the attack. Moreover, some attack testing software can be gained from Internet or other platform.

There is no dependency between the software coding and the security testing coding. So the coordination should not be considered. The coding is summarized in table 4.

3.5 Debug And Testing Management
Having finished coding, the project will enter into debug and testing stage. In the period of software module testing, the upper layer software modules depend on the support of the lower layer software modules.

The function testing can be done without finishing the attack simulation. So it can be tested earlier. Sometimes the security testing can be done without the attack tools or the attack simulations, it also can be test earlier and accelerate the process of the project.

As is shown in table 5, the debug and testing comparison items include participants, entrance conditions, exit conditions, input and output.

Table 4 Software Coding and Security Testing Coding

| Items     | Software Coding                  | Security testing coding                  |
|-----------|----------------------------------|------------------------------------------|
| Participants| Project leaders;                 | Project leaders;                         |
|           | Function design group;           | Function design group;                   |
|           | Security design group;           | Security design group;                   |
|           | Programmer;group;                | Security testing design group;           |
| Entrance Conditions | Programmer group; Software security detail design document is qualified; Software function detail design document is qualified; |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Exit Conditions     | Software interface qualified; Software based on the detail design flow; Code accords with the criterion, software compiled correctly; |
|                     | Software attack program module qualified; Software attack testing tool is obtained; Code accords with the criterion, software compiled correctly; |
| Input               | Software security detail design document; Software function detail design document; |
| Output              | Qualified software and code; Code checkup report; |
|                     | Qualified testing program; Code checkup report; |

Table 5 Debug and Testing

| Items | Function testing | Security testing |
|-------|------------------|------------------|
| Participants | Project leaders; Function design group; Coding group; Function testing group; | Project leaders; Security design group; Coding group; Security testing group; |
| Entrance Conditions | The software code is qualified; | The security testing programs or tools are qualified; |
| Exit Conditions | The function works well. | The security testing is qualified; |
| Input | Web module; | Web module |
| Output | The Web works well. System wrong report. | The web works securely. System security report; |

3.6 Submission and Maintenance Management
Having finished the last stage, you are going to submit the system and to prepare maintenance. If the customer asks for new demands, we may have to modify the code. Some document should be made to submit to the customers.

At the same time, security testing modules should be submitted to the customers who will test the security in the future.

4. Implementations
According to the process discussed above, when we come to implement the power information, we have to do the following work:

4.1 Requirement Analysis
In this period, the analyst should know the network security threats when the Power Information software runs in the network. The following threats are to be pointed out without ask the customers because of their limitation of major knowledge: such as Hit the Base Attack, Brush Credit, Denial of Service, Homepage Tampering, Middleware attack, Web Application Firewall, Event Monitoring and so on. Correspondingly, the corresponding testing modules are required.

4.2 Software Detail Design
In this stage, the software goes straightly to detail design for Power Information.
Firstly, the designer should design the security operation from the security requirement. Secondly, when you have the Power Information function design finished, the designers should firstly learn the modules, the chart or the program flow in detail. Then the designers dig out the dangerous factor with the detail design of the Power Information. Such as the recharge flow without mobile message reminding, which leads to some new threats, for power. For example, when you designed a DB visiting with a SQL structure language, you should consider the SQL injection attack. As a security testing designer, it is necessary to design such threat testing case, preparing a testing tool to test that threat.

4.3 Realization
Having finished the design, the Power Information enters into its realization stage, which includes coding, debug and testing.

All these steps are to realize the Power Information. According the designed document, the coder should code, debug and test the modules proposed in the document. The security testing tools are mostly downloaded from the website or bought from the software corporation:

A) Malware Analysis Toolkit. The website, virustotal.com and jotti.org, can mark the malicious file or URL. B) Metasploit. The software can attack an check out the loophole. C) Nmap. The software can scan the ports of TCP or UDP. It can support Linux and Windows. D) Wireshark. The software is used to analyse the network traffic and the protocol. E) Acunetix. The software can be used to test the attack of web testing, SQL injection attack. F) Burp Suite. Burp Suite is a web application testing tools. G) Cain and Abel. The software can simulate a hacker. It can Cracked the password, pollute DNS, resolve the address. H) TrueCrypt. The software is used to encrypt the message. I) VMWare. The software can simulate different operating system runs at a computer.

4.4 Submission and Maintenance
Having realized the Power Information and finished testing the information, it is time to submit it to the customer and maintain the software for several years.

The submitted content include: software testing toolkit or coded program, the documents of the above process.

5. Conclusions
This paper discusses the design and development feature of security software process. It also introduces the present significance of such adopting process. As a software engineering, security engineering is different from the normal software engineering. The coordination of the function software and the security is discussed in each process, including process includes requirement analysis, design, coding, debug and testing, submission and maintenance. All the paper is based on the Power Information. The implementation of Power Information is an example to introduce the above process.

References
[1] Executive Brief. (2012). 2011 Software Development Trends. Available: http://c.softserveinc.com/newsletter/Software-Development-Trends-Survey-Results-2011.pdf [2] D. Hauenstein, L. Fonda, J. Wynn, and A. Casey, ”Fast track to the future: The 2012 IBM Tech Trends Report,” IBM, New York, USA2012. [3] B. Littlewood, ”The Littlewood-Verrall model for software reliability compared with some rivals,” Journal of Systems and Software, vol. 1, pp. 251-258, 1979. [4] S. Yamada, M. Ohba, and S. Osaki, ”s-Shaped Software Reliability Growth Models and Their Applications ” IEEE Transactions on Reliability, vol. R-33, pp. 289 - 292 October 1984 1984.
[5] J. T. Duane, "Learning Curve Approach to Reliability Monitoring," IEEE Transactions on Aerospace, vol. 2, p. 563, April 1964.

[6] B. Littlewood and L. Strigini, "Advantages and Disadvantages of Software Reliability," in Proceedings of the Conference on the Future of Software Engineering, Limerick Ireland, 2000, pp. 175-188.

[7] M. R. Lyu, "Software Reliability Engineering: A Roadmap," in Future of Software Engineering Minneapolis, USA, 2007, pp. 153-170.

[8] M. R. Lyu, “Handbook of software reliability engineering”. USA:IEEE Computer Society Press, 1996.

[9] Oian Gang, Da Oingli. “Management of info-security engineering based on SSE-CMM model” JOURNAL OF SOUTHEAST UNIVERSITY （Natural Science Edition）. 2002 Vol32(No.1) Jan:1-5, (in Chinese).