EXECUTING SECURITY SCANNING IN SECURE SOFTWARE PROCESS IMPLEMENTATION WITHIN ORGANIZATION

Nor Izyani Daud, Chong Seak Sea, Galoh Rashidah Haron
MIMOS Berhad
Technology Park Malaysia, Bukit Jalil, 57000 Kuala Lumpur, Malaysia
{izyani.daud, cssea, rashidah}@mimos.my

ABSTRACT
This paper attempts to share about an approach on executing security scanning in the secure software process life cycle within organization. It will describe about an overview of the cybercrime issues in Information Technology industry, specifically in Malaysia. It will also discuss about the tool that is used to execute this activity. Then, it will focus on the process flow of the implementation of security scanning process in the organization. Challenges and problems during the implementation also will be discussed in this paper. The authors also will discuss about the trends of vulnerability detected in some of the projects in the organization. The improvement of the security scanning process will be elaborated in detail in order to ensure effectiveness and efficiency of security scanning activity.

KEY WORDS
Security scanning, security risk analysis, security software process

1. Introduction
Cybercrime has become a very critical issue all over the organization in the world. Research from Malaysia Computer Emergency Response Team (MyCERT)[1] shows that report incident regarding the cybercrime increased drastically from year 2008 until 2011. The number of incident reported in the year 2011 has increased more than 88 percent from the number of incident reported in the year 2010. Based on the article, dated on 29 March 2012 [2], Malaysia has lost 2.75 billion ringgit (US$897.6M) over the past 5 years to cybercrime.

There are many concerns come from the security expert regarding cybercrime issue. They started to find a solution on how to secure their company information. As one of the prevention action, most of the company started to implement security scanning activity as a part of their software life cycle.

2. Security Scanning
2.1 Overview
Security scanning activity was adapted to MIMOS Secure Software Process [3] in September 2010. In the secure software process life cycle, security scanning will be executed during the last phase of a software life cycle; Deployment Phase [3].

The purpose of having security scanning activity is to perform vulnerability analysis to the application that the organization develops. It is one of the prevention actions in order to make sure that the entire organization public server is safe from the outside attackers. It is also to ensure that all the servers that we have are updates with the latest patch. The current risk in the organization can be estimated by performing security scanning activity [4]. Once we implement this activity, we can identify the vulnerability and detect the weakness of the current application that we developed. From the result, we can identify that the software that we develop is secure to be deployed to public.

2.2 Tools
The organization has been using Nessus and Acunetix as security scanning tools. For Nessus, we used Nessus version 5.0.0, Professional Edition while for Acunetix, we used Acunetix Web Vulnerability Scanner version 7.0.

Nessus is the comprehensive vulnerability tools. According to the survey done by sectools.org [5], Nessus has become one of the most popular scanning tools. Most of the organization and people has been using it since year 2000 until today. Nessus is still the best UNIX vulnerability scanner available and among the best to be run in Windows [6]. With constantly updates of more than 11,000 plug-ins, it includes remote and local authenticated security checks, a client/server architecture with GTL graphical interface and an embedded scripting language for writing own plug-ins[6].

Acunetix is the most popular and effective web vulnerability scanner tools. It covers most of the vulnerability type. The examples are SQL injection and cross site scripting. US Air Force, Skype, NASA and American Express [7] have been using Acunetix for their vulnerability scanner tool.

2.3 Implementation in the organization
In our current implementation, the scanning tools will be used by security personnel and project team members.
Security personnel will use the tool to perform the scanning for the projects and the organization public servers (Refer to section 2.31 and 2.32). For the project team members, they will use the tool to perform the scanning within their own projects. This is an optional step, but it is advisable for them to scan their own project first before submit a formal request to security personnel to perform the scanning. It is to reduce the number of vulnerability detected during the formal scanning from security personnel.

To implement this activity, security personnel conducted awareness program to the organization members about security scanning process. The purpose of the awareness program is to give an idea about the overall process of security scanning activity and tools that will be used for the scanning. Security scanning activity is performed compulsory in all web application projects that need to be deployed to public.

Before the awareness program starts, security personnel send out an email to all manager levels in the organization to assign their project team members to attend the awareness program. This is to make sure that everyone in the organization is aware about the implementation.

During the program, each of the team members who attend the training needs to sign “Declaration of Authorized Use” form. It is a form that acknowledges a user to use Nessus and Acunetix to perform security scanning activity for their project. It is also to ensure them to be responsible if they use the tools for their own personal purposes. Security personnel will be responsible to submit the form to legal department for record purposes. Once they sign the form, security personnel will create user id for them to access the tools.

The security scanning activity can be separated into two levels. The first level is to be implemented to the entire project in the organization. The second level is to be implemented to the entire of the organization server that can be accessed by public.

2.3.1 Implementation within a project

All projects in the organization were forced to implement the security scanning activity. This activity will involve security personnel and project teams. Figure 1 describes the process of security scanning activity for the project.

![Figure 1: Security Scanning Process Implementation in project](image)

The step below describes security scanning process in the organization.

1. Project team needs to submit System Deployment and Security Risk Analysis document to ISL team for security personnel to review.
2. Security personnel will review the document. It contains all the information needed to perform the scanning. The detailed explanation about the document will be described in section 3.
3. Security personnel will verify either the document is complete or not. If the document is not complete, project team members need to do rework on it based on the needs and requirement.
4. Once the document is complete, security personnel will perform the scanning activity. In this activity, the advanced scanning policy will be use. The detailed explanation about the policy will be described in section 4.1.
5. Once the scanning process is complete, security personnel will generate a report based on the scanning result.
6. From the report, security personnel will identify is there any critical or high vulnerability detected.
7. If the application does not contain any critical or high vulnerability, security personnel will give a recommendation to deploy the application.
8. If the application has the vulnerability, project team needs to fix it. Once the fix is completed, the security personnel need to re-scan again the particular machine or server to ensure that the vulnerability has been fixed.

Figure 2 is an example of the vulnerability detected during the scanning activity.
2.3.2 Implementation within the organization public server

For the second level, security scanning activity needs to be implemented to the entire organization public server. This activity will involve security personnel, project team, Infrastructure team and Internal Audit team. In this activity, security personnel will conduct this activity quarterly. Figure 3 describes the process of security scanning activity for the organization public server.

Once scanning process is complete, security personnel will get the scanning result from the tool.
2. Security personnel will compile the scanning result.
3. Security personnel will summarize the result and prepare the scanning report.
4. If there is vulnerability detected, security personnel need to submit the report to the Infrastructure team.
5. Infrastructure team need to communicate the information to the server owner to fix the vulnerability. Once it completes security personnel needs to perform the scanning again to the particular server only. This is to ensure that there is no more vulnerability found after the fix is completed.
6. Security personnel will submit the report to the Internal Audit team.

3. System deployment and security risk document

As part of the process, each of the projects in the organization will prepare System Deployment Security Risk Analysis document. The purpose of the document is to gather security information regarding the project. The document will also become a reference for project team member regarding their system specification and information. The contents of the document are:

1. Deployment Architecture
   This section shall describe the system specification for all the server or machine involves in the application. The domain name, IP address, CPU size, RAM size, storage size and the operating system needs to be documented clearly in this section.

2. System Logical Diagram
   This section describes the logical diagram for the system. The logical diagram should contain all the connectivity of each machine or server mentioned in Deployment Architecture section which includes the IP address and port number. This is important for security personnel to know how the machine for the application connected to each other. Figure 4 shows the example of System Logic Diagram.
3. System Network and Physical Diagram
This section describes system network and physical diagram for the application. The diagram should contain all connectivity, including firewall, machines, switch and server for each of the machine identifies in the previous section. Figure 5 shows the example of System Logical Network and Physical Diagram.

4. Business Impact and Continuity Plan
This section shall describe business impact of the system in the event of system failure, partial failure or full failure. It should contain the business continuity plan for the application. For example; the backup plan if the application or the database were attacked.

5. Logical Access
This section shall describe all aspects of logical access to the system either by the end user or administrative personnel. The information that should be contained in this section is port number, application, authentication method (e.g., username-password) and account storage (e.g., MySQL, DB2).

6. Physical Access
At this section, it shall describe the administrative practice in controlling physical access to the system hardware. It includes controlling of the access right and usage of the authentication mechanism to enter the system hardware location, for example, using the biometric or access card reader.

7. Security Administration
This section covers administrative practices for resource access, system access and security system control. For example, the admin password rotation, who keeps the admin password or how many persons knows the password.

8. Finding
This section shall describe finding from the scanning result. It needs to be prepared by the security personnel once the security scanning is completed. Security personnel will compare the report from the scanning activity with the information that the project owner provides in the documentation.

9. Recommendation
At this section, security personnel need to give their recommendation for the application based from security scanning result. The recommendation will be a guideline for the project owner to proceed with their project deployment. The example of the recommendation should be the number of vulnerability detected, number of extra open port detected and number of port that not able to be detected.

4. Challenges during the implementation

4.1 Determine best configuration for the security scanning policy
At the beginning of the implementation of the security scanning activity, security personnel need to setup the environment; hardware and software involved. As for Nessus, security personnel have difficulty to determine the best security scanning policy for the organization needs. Security personnel need to do a lot of research from the book and online forum, example [8], [9]. Security personnel need to understand detail about Nessus before they can configure the scanning policy. The scanning policy needed to be detailed to ensure that it is covers the important item during the scanning; for example type of the port scanner to be scanned. The
example of type of the port scanner are TCP Scan, UDP scan, SYN scan, SNMP scan, Netstat SSH scan and Netstat WMI scan. This is the very challenging part for security personnel. If the policy is not done properly, the scanning result is not comprehensive.

Security personnel will revise the security scanning policy as and when necessary. It is based on the update of the latest vulnerability patch detected. The policy also needs to be agreed by the Head of Security Department before it been adapted to the current practise.

The example of the security scanning policy configuration is Global Variable Setting Preference. In this setting, security personnel enabled “thorough test (slow)” configuration. This configuration means that during the scanning, it will trigger the “thorough_tests” keyword within the Nessus plugin script files (.nasl) [12]. Hardware and software including the security scanning policy need to be ready before the roll out of the security scanning process in the organization.

Figure 6 shows the screen capture for Global Variable Setting Preference.

![Figure 6: Example of Global Variable Setting Preference](image)

### 4.2 Lack of knowledge from the project team member and delay project timeline

Project team members play important role in order to execute security scanning during the secure software life cycle. Without them, security personnel will not get any information regarding the project, and therefore, scanning unable to be executed.

The most common problem with the project team members is lack of knowledge on how to fill up the System Deployment and Security Risk Analysis template. The example is the system logical diagram. Most of the time, system logical diagram only contains a server diagram with the connectivity. They do not include specific IP address and port number for each of the server. For security personnel, this is important because we need to specify and identify how the server is connecting with one another. This problem is the common problem that also was discussed in the previous paper [3].

There are also in some cases where the project team members do not know how to read the report from the scanning result. This is because they are not familiar with the scanning tool. To solve this problem, sometimes security personnel need to explain and guide them on how to read and find out information from the report.

Another challenge is projects team do not know how to fix their machine or server if there is any vulnerability detected during the scanning. With the lack of knowledge and experience, sometimes they also need helps from the other team to fix the vulnerability for them.

One of the vulnerability examples is JBoss Enterprise Application Platform. Based from Nessus recommendation, project team has removed the entire http-method element in the security-constraint section. But when security personnel perform the scanning again, Nessus detected the same vulnerability again. Therefore, project team had to read back all the information in Nessus report in detailed and find out that there is an error regarding the JMX Console which they did not realized it previously when they read it for the first time. Once they removed the JMX Console, the problem solved.

End up; all the issues that have been highlighted above will delay the project team timeline.

### 4.3 Understanding security scanning process

It is very difficult to educate the organization members especially project team about the important of scanning activity before the deployment of the application or system to public. For them, the purpose of scanning activity is to detect vulnerability and also open port for the application.

In some cases, there is some project team suggested to use normal or simple scanning policy instead of using advance scanning policy. The reason is to get the fast result of the scanning.

During the scanning, security personnel will use advanced scanning policy. It will require more time to complete the scanning. Most of the project team always referred to the security personnel to estimate the time when will be the scanning process completed. It is impossible because it depends on the policy setting that we use and also the number of server and open port involved for the scanning. Table 1 is the example of time taken for the latest projects that we perform security scanning.

<table>
<thead>
<tr>
<th>Project</th>
<th>Number of Server</th>
<th>Number of Port</th>
<th>Total time for scanning(hour)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>5</td>
<td>11</td>
<td>13</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
<td>11</td>
<td>6</td>
</tr>
<tr>
<td>C</td>
<td>16</td>
<td>32</td>
<td>26</td>
</tr>
<tr>
<td>D</td>
<td>30</td>
<td>65</td>
<td>30</td>
</tr>
<tr>
<td>E</td>
<td>125</td>
<td>80</td>
<td>35</td>
</tr>
</tbody>
</table>

The data in the above table was taken in July 2012. It can be concluded that the more server and port that you have, it will require more time to complete the scanning. Besides that, the organization network traffic also affects
the performance of the scanning activity. If the network traffic is high, the process of the scanning activity will be slow. In our current practise, security personnel will always perform the scanning at the end of the day, so that it will not affect the organization network traffic.

All the above information is the rule of thumb of security scanning process but most of the project team does not understand and aware about it.

5 Trends of vulnerability detected

We had conducted a study regarding the trend of vulnerabilities. Total of six web base application projects were identifies as a sample for the case study. This project was selected based from the highest number of critical and high vulnerability detected. Table 2 describes the vulnerability detected for all the projects.

Table 2: Trend of vulnerability

<table>
<thead>
<tr>
<th>Vulnerability Description</th>
<th>Vulnerability Type</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Windows Remote Desktop Protocol Server Man-in-the-Middle Weakness</td>
<td>Windows</td>
<td>4</td>
</tr>
<tr>
<td>Terminal Services Encryption Level is Medium or Low</td>
<td>Misc</td>
<td>2</td>
</tr>
<tr>
<td>SMB Signing Disabled</td>
<td>Misc</td>
<td>2</td>
</tr>
<tr>
<td>Microsoft Windows SMB NULL Session Authentication</td>
<td>Windows</td>
<td>1</td>
</tr>
<tr>
<td>CGI Generic SQL Injection Detection (potential, 2nd order, 2nd pass)</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>CGI Generic SQL Injection (HTTP Headers)</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>XAMPP ADOdb msqql_connect Remote Buffer Overflow</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>Apache HTTP Server Byte Range DoS</td>
<td>Web Servers</td>
<td>2</td>
</tr>
<tr>
<td>Apache 2.2 &lt; 2.2.15 Multiple Vulnerabilities</td>
<td>Web Servers</td>
<td>1</td>
</tr>
<tr>
<td>phpMyAdmin Installation Not Password Protected</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>PHP 5.3 &lt; 5.3.7 Multiple Vulnerabilities</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>PHP 5.3 &lt; 5.3.3 Multiple Vulnerabilities</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>PHP 5.3 &lt; 5.3.9 Multiple Vulnerabilities</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>PHP 5.3 &lt; 5.3.4 Multiple Vulnerabilities</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>PHP 5.3 &lt; 5.3.6 Multiple Vulnerabilities</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>Web Application SQL Backend Identification</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>Apache HTTP Server httpOnly Cookie Information Disclosure</td>
<td>Web Servers</td>
<td>1</td>
</tr>
<tr>
<td>phpMyAdmin error.php BBcode Tag XSS (PMASA-2010-9)</td>
<td>CGI abuses : XSS</td>
<td>1</td>
</tr>
<tr>
<td>Apache 2.2 &lt; 2.2.17 Multiple Vulnerabilities</td>
<td>Web Servers</td>
<td>1</td>
</tr>
<tr>
<td>Apache 2.2 &lt; 2.2.21 mod_proxy_ajp DoS</td>
<td>Web Servers</td>
<td>1</td>
</tr>
<tr>
<td>Apache 2.2 &lt; 2.2.16 Multiple Vulnerabilities</td>
<td>Web Servers</td>
<td>1</td>
</tr>
<tr>
<td>Apache 2.2 &lt; 2.2.18 APR apr_fnmatch DoS</td>
<td>Web Servers</td>
<td>1</td>
</tr>
<tr>
<td>Apache 2.2 &lt; 2.2.22 Multiple Vulnerabilities</td>
<td>Web Servers</td>
<td>1</td>
</tr>
<tr>
<td>PHP expose_php Information Disclosure</td>
<td>Web Servers</td>
<td>1</td>
</tr>
<tr>
<td>Apache mod_info /server-info Information Disclosure</td>
<td>Web Servers</td>
<td>1</td>
</tr>
<tr>
<td>Adobe Flex SDK Cross-Site Scripting (APS11-25)</td>
<td>CGI abuses : XSS</td>
<td>1</td>
</tr>
<tr>
<td>Backup Files Disclosure</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>HTTP TRACE / TRACK Methods Allowed</td>
<td>Web Servers</td>
<td>1</td>
</tr>
<tr>
<td>PHP 5.2 &lt; 5.2.17 / 5.3 &lt; 5.3.5 String To Double Conversion DoS</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>PHP &lt; 5.3.2 / 5.2.13 Multiple Vulnerabilities</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>MS10-070: Vulnerability in ASP.NET Could Allow Information Disclosure (2418042) (uncredentialed check)</td>
<td>Windows</td>
<td>1</td>
</tr>
<tr>
<td>LDAP NULL BASE Search Access</td>
<td>Misc</td>
<td>2</td>
</tr>
<tr>
<td>CGI Generic Command Execution(time-based)</td>
<td>CGI abuses</td>
<td>3</td>
</tr>
<tr>
<td>CGI Generic Command Execution</td>
<td>CGI abuses</td>
<td>4</td>
</tr>
<tr>
<td>Sitefinity CMS Arbitrary File Upload</td>
<td>CGI abuses</td>
<td>1</td>
</tr>
<tr>
<td>CGI Generic Cross-Site Scripting(comprehensive test)</td>
<td>CGI abuses</td>
<td>2</td>
</tr>
<tr>
<td>CGI Generic Cookie Injection Scripting</td>
<td>CGI abuses</td>
<td>3</td>
</tr>
</tbody>
</table>
The data in the table was collected from October 2011 to September 2012. This data was taken from Nessus report for all the projects in organization within the time frame. All the vulnerability that stated in the table was in Medium, High and Critical severity. Figure 7 describes a summary of the number of vulnerabilities detected and the plugin type for the data collected in Table 1.

Figure 7: Summary of vulnerability detected

From Figure 7, CGI abuses plugin has contributed the highest number of the vulnerability detection in our organization. While for the Web Servers plugin has contributed 21% of the total vulnerability detected.

Based from the report by Cyber Security team [10] and statistic [11], the latest cyber security crime is towards Intrusion; which is the web server plugin. This is related to the web application or unpatched servers, mostly web servers that running on IIS and Apache.

Based from the report, the number of Malicious Code attack; which is CGI abuse, contributed 353 cases in January 2012 until June 2012.

From OWASP link [13], out of 10 items, 9 items are related to the programming. Only one item is related to the software or framework configuration. This is similar to the current trend in our organization, where the highest vulnerability is regarding the programming and software configuration.

Based from these sources, it showed that programming was still the major cause for the cyber-crime issues until now.

6. Improvement

The process improvement in this section is the collaboration with our Corporate Quality department to improve the current security process.

6.1 Embedded JUnit testing and Sonar during the development phase to improve the quality of code

As what has been mentioned in previous section in Table 2, the quality of the program written by developer is the major cause of the vulnerability detected. Based from the article [14] also agreed that a good practise of secure software need to be in place to overcome this problem.

Our organization was started to enforce all the project teams to start using JUnit testing and Sonar in year 2010. The purpose of implement this activity is to have the quality of the code by having the code analysis. JUnit testing was able to detect the validation of any input by the user. It can prevent the attacker to pass any unnecessary value for the input text in their application. While for Sonar, it will check the code coverage for the application. Several rules need to be configured to check the code coverage. This is where the vulnerability criteria were defined. The example of the vulnerability that able to be detected by Sonar is SQL Injection [15].

The figure below is the example of the project that implements Sonar report for their code coverage analysis.

Figure 8: Example of Sonar code coverage analysis

In our current practise, the quality of the codes is depends on the project team members. However, Quality Engineer will provide a guideline for the team how to define the level of the quality of codes for each project. For this project, the example is in the below table:

Table 3: Quality of code standard for project A

<table>
<thead>
<tr>
<th>Quality Criteria</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line Coverage</td>
<td>&gt;95%</td>
</tr>
<tr>
<td>Branch Coverage</td>
<td>&gt;70%</td>
</tr>
<tr>
<td>Unit Test success rate</td>
<td>100%</td>
</tr>
<tr>
<td>Code Violation</td>
<td>Blocker = 0, Critical = 0, Major &lt;50</td>
</tr>
</tbody>
</table>

Based from the table, for the project A, the branch coverage and unit test success rate meet the quality of code standard. For the line coverage and code violation, it is below the quality of standard. At this stage, project A teams member need to fix the code violation for major issues.
6.2 Obfuscation

Obfuscation is one of the approaches that had been highlighted to the project team member. With obfuscation program, it will make the code harder to be understood. The obfuscation program is mainly purpose for security reason. It is to prevent tampering, deter reverse engineering or challenge someone from reading our own source code. The main methods of obfuscating are layout obfuscation, data obfuscation and control obfuscation.

In our organization, we already starts using ProGuard[16] as our Java obfuscation program. ProGuard is a free Java class file shrinker, optimizer, obfuscator and pre-verifier that can detects and removed unused classes, fields, methods and attributes. This practise has been adapted to all the projects team members starting from end of 2010.

7. Conclusion

As a conclusion, security scanning is an important activity that is required in the software development life cycle. It can be one of the prevention actions by organization in order to avoid cybercrime attack from outsiders.

Nessus and Acunetix are used to perform the scanning activity. A proper study regarding cost for maintenance and licensing, ease of use and resource knowledge about the tools need to identify and analyze before the organization purchase any tools for the activity.

The implementation that was discussed in this paper is the best practise for the organization. The challenges, research on the vulnerability trends and process improvement described are the experience from security personnel, where it can be a guideline for the other organization before adapting the process to their software life cycle. As security scanning process is a living process that needs consistent improvement over time.

Acknowledgements

We would like to thank to the top management for the support in order for us to implement security scanning activity in the project life cycle. The security scanning process is a collaboration effort from the Corporate Quality, Infrastructure Group, Internal Audit Group and Information System Laboratory as the owner and finder for the security scanning process implementation.

References