Research Institution Software Development Process Improvement: to Produce High Quality Research Software & Assessment on Technical and Software Package Installation

Sea Chong Seak*, Chang Pei Shan¹, Wong Hon Loon² and Dahlia Din³

Information System Security Lab, MIMOS Berhad, Technology Park Malaysia, 57000 Kuala Lumpur Malaysia.
E-mail: cssea@mimos.my*, ps.chang@mimos.my¹, hl.wong@mimos.my², dahlia.din@mimos.my³

Abstract. Our objective as a research engineer is to produce high quality research software, from the simple software application to extremely large complex software system platforms able to support large organisation. However, lack of appreciation for the significance of software as a research output often leads to software application created in research institutions being treated as a secondary concern. Commercial software engineers tend to follow more disciplined approach to software development principles & follow best practice, this is the reason why they can create their software application generally more sustainable and usable. We combine our research experience with an appreciation of good software engineering principles to build more readable, reliable, efficient code and continue develop sustainable software framework. In this paper, we design and built a simple framework to allow the software engineer who lack of research context & network security knowledge, able to install and setup the lab authentication system platform. Hence, the system engineer can integrate the software platform with prepared virtual machine image and configure the status of resources via the given guideline in product technical manual. The product quality improvement and verification process will be available during the proposed experiments. The results show that the problems of inconvenience software engineer in installing process and other issues are resolved; at the same time, convenience for the software deployment and the experience & skill of the system engineer also improved.

1. Introduction
The nature of research software is often developed quickly to solve one-off problems, leading to fragile code that is generally not sustainable or usable beyond the lifetime of a given project, and is hard for other researchers to read and understand it. Many project collaborations with commercial software company fail due to lack of understanding of the research context. Our work is not just about producing research software on a first-class research output, we also work collaboratively, providing the tools, advice and researchers need in order to follow best practice and continue to build a high quality & sustainable software infrastructure.

In today society, the software applications to be deployed, and the computing environments in which they will operate, range from very simple to extremely large and complex. Configuring a server can be a labour intensive and challenging process [1]. Despite advances in technology, the integrated installation of multiple computer servers can be a difficult task. For example, a system engineer installing multiple servers, including operating system and application software, may not have enough knowledge or expertise to install and configure the servers correctly. Installation on
multiple servers can involve complex installation and configuration tasks [2]. It is often necessary to employ a specialized Information Technology (IT) professional to correctly install and configure software components to provide the desired functionality on the server.

With the increasing popularity of personal computers came a trend toward easier, more user-friendly software installation, as software vendors recognized that it was no longer reasonable to assume that a person with a high degree of technical skill would be performing every installation process [3]. Advances in this area have been made in recent years; many software packages use some sort of automated or self-installing procedure. Commonly, automated method often provides a file as an installation media and it is the easier way to let system engineer to start an installation. However, for network security product that are involved large scale applications may be not working for the automation method. There might be large number of parameters have to compute during the installation of a particular application package, this process might be complicated for a person who lack of network security knowledge and the configuration may even vary from different deployment environment.

Therefore, to deliver safe and high-quality security products that can be used without suspicions, we aimed to ensure product quality in the development and production stages. The related solutions in this paper proposed use of a framework for software installation packages and addressed many of these problems of the pass deployment faced, enabling the installation process to be simplified for software engineers as well as for the software developers who must prepare their software for an efficient, trouble-free installation, and define several techniques for improving installation of software packages [3].

2. Backgrounds

The authors select one of the research output system produced by the laboratory for this experiment and improvement for better installation, configuration and build the sustainable software infrastructure.

2.1. Research Output Software Platform

The organization Information System Security research lab has developed a multimodal authentication platform to provide a convenient and user-friendly service for both organizations and individual users. It introduces six authentication methods available for users to choose. The key feature of this authentication platform can simplifies the complexity of managing multiple user IDs with seamless identity management using a trust model approach. It is also utilizes Single Sign-On (SSO) for a seamless environment operation and adaptive authentication and threat response without modifying existing applications.

As per design, this authentication platform has several modules to integrate which is showing in Figure 1. For example, the functional modules for authentication server is the Identity Provider (IDP) component. Each modules implemented on a virtual machine (VM). Hence, it will turn into a set of modules that able to implement on physical server formerly become the authentication platform.

2.2. Problem Statement

During our initial research findings, we have addressed some challenges and issues related with current platform deployment process. The major problems of the platform deployment or installation is not able to perform by external parties other than our team member, it is due to as listed following:

(1) Complexity in system installation because the platform required to integrate multiple modules inside different server with their own service to be installed and configured.

(2) Technical and domain knowledge in our authentication platform is required to understand the installation procedure, otherwise, hands-on guidance is required to run the installation steps.

---

![Figure 1. Authentication platform modules](image_url)
Information in the software installation manual is incomplete, ambiguous and not structured.
Configuration parameters in each server might not updated due to human mistake.
There have been past experiences where the authentication platform software builds have failed or broken due to technical and incompatibility issue, or discrepancies found in the guidance documentation.

3. Related Works

3.1. The Experimental Environment

Table 1. Virtual machine with minimum system requirement

| Virtual Machine Server(s)            | Hardware Requirement |
|--------------------------------------|----------------------|
| LDAP, Registration, Authentication Gateway, Trust Engine, TCK, OTP | 2-Cores  4  30       |
| Authentication                       | 6-Cores  8  30       |
| MySQL                               | 8-Cores  4  30       |

For the authentication platform system installation, it is required to prepare a physical host with 8 mandatory Virtual Machines (VM) describe above. For the physical host, it is recommended to use a 32-core CPU, 48GB RAM and 500GB disk storage.

3.2. Experimental Objective
The main objective of this product improvement is to improve the security software packaging quality and installation approach. An aspect of the paper provides a mechanism to install the software with less effort and less risky of having deployment defects.
An improvement attributes of this paper is at improving the install-ability which minimize the installation steps to enable more effective and reliable deployment of software package in a flexible manner. In addition, deploy the software packages that attempt to meet the needs of many varied target environment and the skills of many different system engineer.

3.3. Experimental Solution
A variety of technologies related to the integrated installation of multiple virtual machine servers can be applied. The experiment approach is as following:

(1) Provide golden image of VM for each modules.
   Packaging of the authentication software to be redesigned to ease of installation. The new approach is to have the platform packaging image made into VMs per server, so the system engineer could deploy and configure the VMs faster and in a simpler way.
(2) Update and verify the guidance document.
   Documentation related with installation guidance of the authentication software is required to update for providing complete and clear information to the system engineer. These manuals provided textual information on how to perform the installation of a particular VM module such as LDAP or MySQL should be numbered with a dot following the number and then separated by a single space:
We propose the solution framework as shown in Figure 2. First, the system developer of the authentication software has to prepare the image of VMs as in Step 1 and 2. In addition, system developer also has to prepare the installation manual for the VM images installation which listed the steps to configure the particular module. This is to minimize the effort and complexity in installation and configuration for the system engineers who may lack of network security knowledge. Once the installation manual is ready for system engineer to perform installation, it has to store inside centralized storage as well.

After that, system engineer to setup the authentication software by deploy with the proposed solution. First, system engineer have to download all the VM images and installation document from the centralized storage. Then, physical server has to ready as requirement inside the installation manual. Also, numbers of VMs have to create based on the listed specification for different modules as shown in Table 1 and install with the downloaded VM image accordingly. After the VMs are setup, identifying the networking setting based on the physical server configuration. The most important stage is to configure the VM according to different module by follow the step which listed inside the installation manual. In order to evaluate the performance of the package installation quality, measurement data shall be collected from step 3 to step 5 for this experiment.

**Figure 2.** Experiment solution framework
4. Experimental Results

Table 2. Measurement parameter before and after solution deploy

| No. | Mi-UAP Server               | Installation (Staff Days) | Configuration upon installation (Staff Days) | Total Effort (Staff Days) |
|-----|-----------------------------|---------------------------|---------------------------------------------|--------------------------|
|     |                             | from binaries             | from Virtual image (VM)                     | from binaries            | from Virtual image (VM) |
| 1   | LDAP Server                 | 2 hours                   | 1/2 hour                                    | 2 hours                  | 1/2 hour               | Half day               | 1 hour                 |
| 2   | Registration Server         | 2 hours                   | 4 hours                                     | 1 hour                   | ½ day                  | 2 hours                |
| 3   | Authentication Server       | 2 hours                   | 1/2 hour                                    | 4 hours                  | 2 hours                | ½ day                  | 2 ½ hours              |
| 4   | Authentication Gateway Server| 2 hours                   | 1/2 hour                                    | 2 hours                  | 1 ½ hours              | Half day               | 2 hours                |
| 5   | OTP Server                  | 2 hours                   | 1/2 hour                                    | 2 hours                  | 1 hour                 | Half day               | 1 ½ hours              |
| 6   | TCK/20 Barcode Server       | 2 hours                   | 1 hour                                      | 2 hours                  | 1 hour                 | Half day               | 2 hours                |
| 7   | Trust Engine Server         | 2 hours                   | 1 hour                                      | 2 hours                  | 1/2 hour               | Half day               | 1 ½ hours              |
| 8   | MySQL Database Server       | 2 hours                   | 1 hour                                      | 2 hours                  | 1 hour                 | Half day               | 2 hours                |
|     | Total                       | 16 hours                  | 6 hours                                     | 20 hours                 | 8 ½ hours              | 4 ½ days               | 14 ½ hours             |

Table 2 shows the results of parameter before and after the proposed approach deploy. The experiment split to 2 groups which is installation and module configuration time. Last column is the total effort to install each virtual machine in unit man day. In total effort to setup the authentication software package, it needs 1 week of staff time virtually. In addition, the collected parameters are provided from system developer who has experience and knowledge on the product. The installation of the authentication software is failed when test engineer (quality team member in this experiment) try to install from binaries. This shows that the complexity of the product setup is complex and unable to perform the installation by external parties other than the network security team.

The data shows after using the proposed approach is reduced and total efforts taken to setup the authentication software is 14.5 hours which is less than 2 man days. Furthermore, the parameter is provided by the system engineer who is inexperienced for the security product. The system engineer performed the installation experiment by follow the steps in Figure 2 and it is successfully installed. This is proved that the proposed approach able to let 3rd party deploy the authentication platform by minimize their effort to understand the security product.

![COMPARISON OF EXPERIMENTAL RESULTS](image)

**Figure 3.** Statistic result of two experiment

Figure 3 indicates the comparison statistics of 2 experiments performed before and after deploy the proposed solutions. The results showed that the overall effort to setup each VM is reduced more than
50%. Thus, the experiments and results shown in this paper can be verified that the security product packaging quality is improved and is working for external party.

5. Discussion and Conclusion
With the increasing popularity of computing devices came a trend toward easier, more user-friendly software installation, as software vendors recognized that it was no longer reasonable to assume that a person with a high degree of technical skill would be performing every installation process. However, a number of problem areas remained because of the lack of a standard, consistent approach to software installation across product and vendor boundaries [3]. For this purpose, enhancing software integrity through installation and verification is important for every software products.

First of all in this paper, a problem pointed out by third party saying that the installation process of our authentication software is difficult which leading to confusion for system engineers. These problems being exacerbated when the installation manual has been addressed out hardly to understand by system engineer who is network security vulnerability. Therefore, this paper proposed a solution to improve the product quality by simplify the software installation process as described in session III above.

The results of this paper indicate that the proposed approach which did the installation from ready VM image is positively associated with product quality improvements. The evidence also indicates that the quality improvements achieved through the external party are equivalent to those experienced developer [4] [5]. In addition, the solution on this paper avoids the misinterpretations of system engineer on the security product that might lead to wrong installation and affects the software performance capability. Also, it able to raise the adaptability to multiple deploy environment and increase the productivity as a better installation process is provided [6]. For future work to continuous improve the product quality, it is suggested to conduct training to deliver the installation knowledge for system engineer and raise the skill levels of those in charge of product development, so that to prevent not only reoccurrence of past defects, but also share new cases found during the developing process.

As a conclusion, product quality improvement and verification is helping to improve product quality in long term and offers a great result to boost production and productivity by reduce defects.

6. References
[1] Andrew T. Mason, William A. Laing, John MacIntyre, Richard A. Pletcher, “Role based server installation and configuration,” US7793284B2, 7 Sept 2010.
[2] Michael Kramer, Eric Watson, Leszek Mazur, Neil Fishman, Drew Spencer, “Integrated multi-server installation”, US20070234345A1, 22 Feb 2006
[3] James Fox, Robert Leah, John McGarvey, “Object model and framework for installation of software packages using a distributed directory,” US2003018964A1, 23 Jan 2003
[4] Bojan Acko*, Simon Brezovnik, Boris Sluban, “Verification of Software Applications for Evaluating Interlaboratory Comparison Results”, 24th DAAAM International Symposium on Intelligent Manufacturing and Automation 2013
[5] Shusaku Tsumotoa, Shoji Hiranoa, “Data Mining oriented Software Quality Estimation”, Information Technology and Quantitative Management (ITQM 2016)
[6] Gregory Hamel, “Benefits of Continuous Quality Improvement (CQI)”, http://smallbusiness.chron.com/benefits-continuous-quality-improvement-cqi-40534.html