

Automating Patching Utility

Mamatha¹, Dakshayini M²

¹(M.Tech. Student, Department of ISE, BMSCE, Bangalore, India)

²(Professor, Department of ISE, BMSCE, Bangalore, India)

Abstract

Many companies releases product for the customers. Not all products will be perfect without bugs. There may be some bugs which are critical in their function and needs to be fixed to the software. After fixing the bugs in the software with regression testing, the company releases patches to the customer which needs to be installed in the software in order to incorporate those bugs or fixes. Currently Oracle has a tool called OPatch to install the bugs or fixes in the software which should be done manually consuming sufficient time. In today's era, time is very important in the human life. Hence, we have implemented and presented an automated tool to install the patches in the software to save the precious time and also reduce the error while installing the patches. The implementation results have demonstrated this achievement.

Keywords

bugs, manual Patching Utility, automation and Weblogic

I. INTRODUCTION

IT companies may be a product based or service based. Product based companies develop their own product and sell to the stakeholders or customers whereas service based companies acquire projects from other companies or customers [1]. Both type of companies should have management plans before developing any software or product. Project Management plan includes identifying requirements, Analysis, Design, Implementation, Testing, Release, Maintenance. Each and every step is important while developing any product or software.

After the Release of the product, there may be bugs in the future. In order to fix the bugs in the customer installation patches are released by the company. Patch can be applied manually or automatically without human interventions. Manually bug-fixing or applying patch may lead to error prone. The goal of automatic patch apply is to reduce the error and apply the patches to the software in a correct way without causing software regression. By applying the patches automatically to the software saves the human time. Patches can be released in terms of many bug fixes or one bug fix.

If any bugs needs to be updated in the software or product, patches are released. Currently, some companies to install or apply the patches for the software, OPatch [2] is used which is Oracle product.

But O Patch is not automated, users should manually apply the patch to the software which may lead to error and time consuming.

Before applying a patch for the software, some pre-requisites need to be handled. But if users forget to meet those pre-requisites then the software may end abruptly or lead to many errors. And also if user forgets to take backup of the software or product before applying a patch then the user may lose the data like database configurations. So in-order to eradicate the problems mentioned above, a tool or utility is developed which is automated for applying the patch to the product or software.

Any product which is developed and tested is released for the customers. After some days software may run into some issues called bugs. In order to fix the bugs in the customer installation patches are released by the company. Patches can be released in terms of many bug fixes or one bug fix.

Patch can be applied manually or automatically without human interventions. Manually bug-fixing or applying patch may lead to error prone. The goal of automatic patch apply is to reduce the error and apply the patches to the software in a correct way without causing software regression. By applying the patches automatically to the software saves the human time.

A) Objectives

Below are the objectives for the automating patching utility:

- The tool should apply the patch without error prone.
- The tool should handle all the error case like if file not found.
- The tool shall able to rollback the patch

II. REQUIREMENT SPECIFICATION

Software requirement specification is important document for product or software which acts as agreement between the company and stakeholders or customers. It includes functional requirement, non-functional requirement, hardware specification and software specification.

A) Functional Requirement

- The tool shall take backup of the installation.
- The tool shall check for the Pre-requisites.

- The tool shall apply the patch to the installation.
- The tool shall restart the servers after successful applying the patch to the installation.
- The tool shall apply the pre-patch and post patch configurations if needed.
- The tool shall check for the patch version before applying the patch.
- The tool shall able to roll back the patch if needed.

III. DESIGN

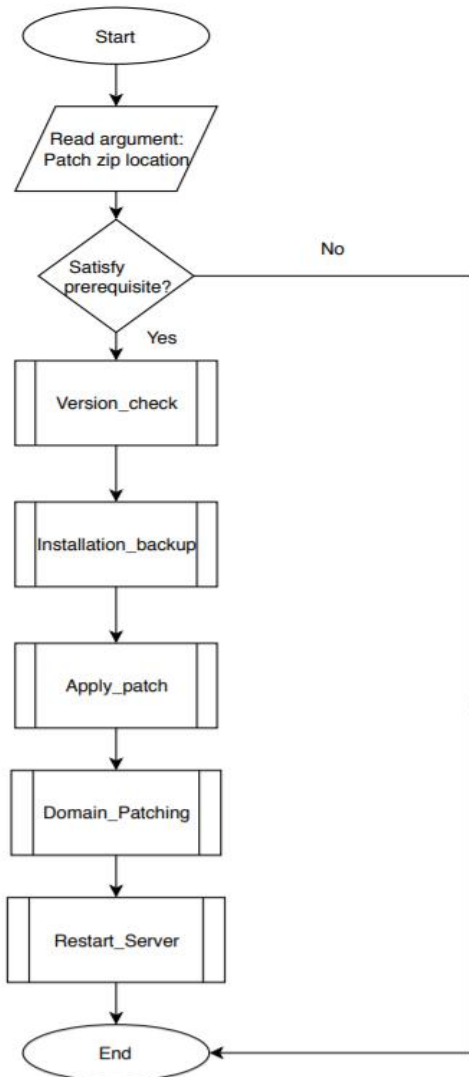


Fig 1: Flow Diagram for patching utility.

Figure 1 shows the flowchart for patching utility. It takes the patch zip locations as an input and it must be in integers only. Before applying patch pre-requisites are checked. If not satisfies then it exits otherwise it continues with the version check of the software and the patch version. It takes the installation backup and then apply the patch. After applying patch, domain patching is done and to redeploy [3] the server needs to be restarted. After all

the processes mentioned above, the tool exits after completion.

IV. IMPLEMENTATION

For any software before applying the patch, Backup should be taken. Sometimes there may be a pre-requisites conditions that needs to be checked before applying the patch. After successfully checking the pre-condition, the tool applies the patch. If the above mentioned steps are done manually then it would be time consuming and also leads to some error. If suppose user forgets to do pre-requisite check then while applying the patch it will fail. From this it will be error prone and time consuming. If it is done automatic then user will save at least approx. 10 mins of their time. Because applying the patch takes some time then if pre-requisites fails then again user has to repeat the steps. After successfully applying the patch, server can be restarted if it is deployed in the server. By automating the patching steps it saves the precious time and error are also reduced drastically. Python language is used to implement the utility.

Rollback option is also available in the utility. Rollback is the very important steps that needs to be taken care of. For each patch the patch number is unique. While applying the patch, user as to give patch location which will be in the zip format and the tool is unzips it and applys to the software and also it note downs the patch number. While roll back, user has to given patch number that they want to uninstall. If patch number is present then tool uninstalls it successfully otherwise the tool exits by saying Patch is not installed.

V. RESULTS

```

Patching directory [//tmp/ocsg-patch] exists. Performing cleanup ...
Extracted patch file to [//tmp/ocsg-patch] successfully
Checking if patch '25800185' has been applied before ...
To backup the OCSG, the tool needs to evaluate its size. This may take a while ...
Estimated size of OCSG home directory is [1.4G]. Kindly make sure that backup directory
has enough disk space. The backup would be stored as a zip file.
Enter backup-directory for OCSG: /scartch
Invalid path. Please try again!
Enter backup-directory for OCSG: /user
Invalid path. Please try again!
  
```

Fig 2. Invalid path

Figure 2 shows the output for invalid path. If user gives an invalid path for taking the backup, the tool again asks for the valid directory to take backup.

VI. CONCLUSION

There may be a critical bugs or fixes that needs to be done to the software. After fixing the bugs in the software with regression testing, the company releases patches to the customer which needs to be installed in the software in order to incorporate those bugs or fixes. Instead of applying the patch manually which leads to error prone and time consuming, user can do automatic. To install the patches in the software is automated so that time can be saved and also reduces the error while installing the patches.

REFERENCES

- [1] Mathias Ekman , Henrik Thane “Dynamic Patching of Embedded Software” Real Time Embedded Technology and Application Symposium in IEEE 1545-3421 2007.
- [2] Opatch is oracle product to apply patch & the link is https://docs.oracle.com/cd/B28359_01/em.111/b31207/oui7_opatch.html
- [3] Weblogic is a server to deploy the application and the link <https://www.oracle.com/in/middleware/weblogic/index.htm>
1