

# AppleSUS

## From DEGWiki

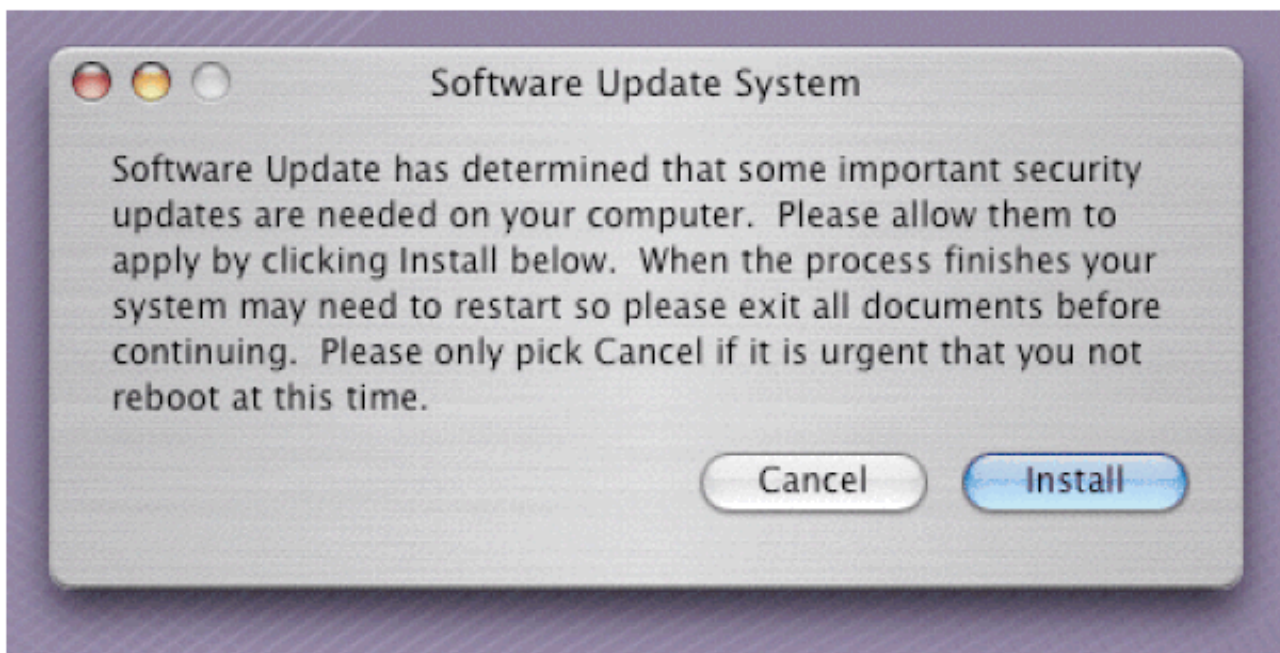
Deploying security updates to deployed workstations has proven itself to have been of the utmost importance industry wide. MacOS X is not immune from this requirement, but out of the box Apple has provided no methodology by which we can select which packages are authorized for deployment in our environment from a stability sake.

To address this, the Desktop Engineering Group has developed a method for doing this. We call it the Apple Software Update Service, or AppleSUS for short :). This document outlines the behavior of AppleSUS for you the technician so you are aware of it's behavior. We will discuss both the user impact, and the behind the scenes elements in this document.

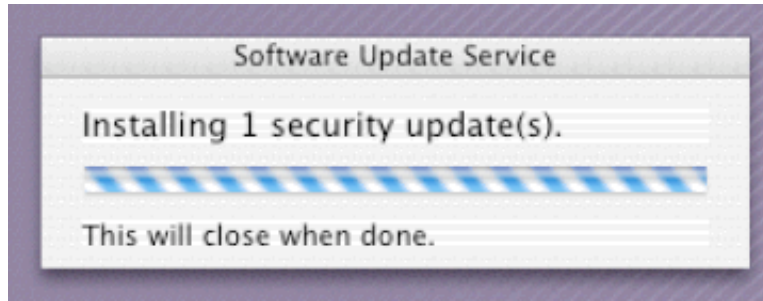
## First, the user experience

When a workstation determines that an approved update is needed the following dialog is shown. If the person chooses to Cancel this, they will get the same dialog the next day, and so on, and so on.

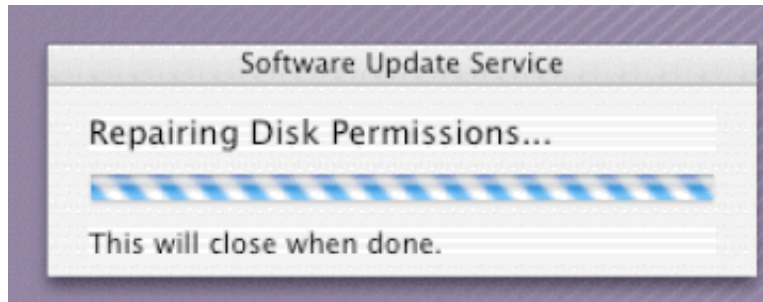
*Note that although this system is automated, it is still incumbent on all of us to verify that these patches have been applied. The best tool for doing this is NetOctopus.*



Once the end user clicks on Install above, a dialog box will show the installation of the needed updates as shown:



Once the updates are applied, the workstation's permissions are fixed to correct any permission problems. The status of this is also shown to the user.



Once all updates have been applied the user will be prompted for a reboot if necessary on Panther, but on Tiger they will always need to reboot. This is due to changes in the way that command line softwareupdate works.

## Behind the scenes

The actually detailed process behind this system is detailed below...

1. A cron job fires daily at about 6 minutes after midnight, or 6 minutes after machine power up
2. The shell script checks for en0 or en1 to be up for connectivity (to avoid dialup updates)
3. If en0 or en1 are up, shell script checks for connectivity to liveupdate.timeinc.com once every 10 seconds for 10 minutes to confirm reliable connection
4. If connection not reliable, process aborts
5. If good connection exists, the script copies text file contents (approval list) from liveupdate.timeinc.com into a local variable
6. Softwareupdate -l run on the local workstation--the output is redirected to a variable for comparison to the approval list
7. The above 2 variables are compared for any matches in strings
8. If there is a match, it means that an approved update is required
9. Initial dialog to user displays at this point if someone is logged in (if no user, no dialog)
10. The script counts the matches between the 2 variables to know the number of updates needed (only for user display)
11. Softwareupdate is then run with the entire update list from the approved variable
12. User dialog displays "applying patches" dialog
13. Because the whole list is used--if the patch is needed, it applies--if not, it is skipped
14. If the patch requires a reboot, that reboot string is captured and is used to tell Apple SUS to reboot when

done. On Tiger a reboot is always required.

15. Once all needed patches are applied, permissions repair is launched

16. If noone is logged in, reboot happens automatically if needed

17. If logged a user is logged in, 60 second applescript dialog prompts for reboot, but then reboot on timeout

18. Process Complete

Retrieved from "<https://dsg.it.timeinc.com/mwiki/index.php/AppleSUS>"

---

- This page was last modified 13:46, 20 May 2005.