



Upgrading CFEngine Nova

A CFEngine Special Topics Handbook

CFEngine AS

This document describes how software updates work in CFEngine Nova.

v

Table of Contents

What does upgrading mean?	3
Why do I need to upgrade?	3
How does upgrading work?	3
How can I do phased deployment?	4
What if I have multiple operating system platforms?	4
How do Nova policies update if I already have my own policy?	4
Appendix A Manual package upgrade commands	5

What does upgrading mean?

A software upgrade involves obtaining a new version of the CFEngine software from software.CFEngine.com and installing it in place of the old. When software is updated, the previous version of the software is retained.

From version 1.1 of CFEngine Nova, CFEngine is fully capable of managing its own updates and service restarts with a minimum of manual work on the policy server.

Existing users of version 1.0 will need to upgrade the software manually on the affected systems, or use the existing CFEngine to assist in the manual process. Please contact CFEngine Professional Services for assistance (see Appendix).

Why do I need to upgrade?

Bug fixes and new features are included in new software releases. To gain access to these fixes, you need to upgrade the software. Changes to the standard Community Open Promise Body Library might make use of new features, so upgrading brings you access to these new methods.

How does upgrading work?

CFEngine packages its software in operating system compatible package formats (RPM, PKG, MSI, etc). New packages are made available on the software.CFEngine.com website; these can be downloaded and installed in the standard way.

1. Go to the `software.cfengine.com` website
2. Select the version of the software you require
3. Download the particular package for your operating systems to a correspondingly named subdirectory on your policy server under `'/var/cfengine/master_software_updates'`, or simply mirror the entire file tree on your policy server under `'/var/cfengine/master_software_updates'`.
 - You do not need to delete the contents of the work directory, e.g. `'/var/cfengine'` to upgrade. An upgrade only involves changing the binary code base.
 - You might wish to take advantage of changes to the Nova base policy, e.g. the `'update.cf'` policy.
 - You can upgrade systems manually by copying the relevant package to the local host and running the appropriate package update command (see summary below).
 - To upgrade all of your systems automatically, you only need to copy a new package release to a specially designated location on your policy server, and CFEngine will do the rest. Each CFEngine client will detect the presence of the update, download it, install and restart CFEngine services locally. Thus you can deploy software using a single point of management.

Commonly used commands used to install packages (see also Appendix A):

```
Debian or Ubuntu: dpkg --install cfengine-nova-xxx.deb
Windows:         Double click or run msixec /qn /i cfengine-nova-xxx.msi
RPM families:   rpm -ivh cfengine3-nova-xxx.rpm
FreeBSD:        pkg_add cfengine3-nova-xxx.tbz
Solaris:        gunzip cfengine3-nova-xxx.pkg.gz \
                && pkgadd -d cfengine3-nova-xxx.pkg all
```

How can I do phased deployment?

Using CFEngine classes to select a subset of machines, you can deploy of updates on a small number of test systems first.

What if I have multiple operating system platforms?

As of version 1.1 of CFEngine Nova, CFEngine will look for updates in an operating specific location on the policy server. To update a particular operating system, you only need to place its package in the correct subdirectory and the client host will know where to look.

The current directory structure mirrors that at the software.cfengine.com website.

```
centos_5_i686/      freebsd_7.0_i386/      suse_10_i686/
centos_5_x86_64/   freebsd_7.2_amd64/    suse_11_i686/
debian_3_i686/     freebsd_7.2_i386/     suse_11_x86_64/
debian_4_i686/     freebsd_8.0_i386/     suse_9_i686/
debian_5_i686/     redhatenterpriseas_3_i686/ ubuntu_8_i686/
debian_5_x86_64/   redhatenterpriseas_4_i686/ ubuntu_8_x86_64/
fedora_10_i686/    redhatenterpriseas_4_x86_64/ ubuntu_9_i686/
fedora_10_x86_64/ redhatenterpriseserver_5_i686/ ubuntu_9_x86_64/
fedora_11_i686/    redhatenterpriseserver_5_x86_64/ windows_i686/
fedora_11_x86_64/ sunos_5.10_i86pc/      windows_x86_64/
fedora_12_i686/    sunos_5.10_sparc/
fedora_12_x86_64/ sunos_5.9_sparc/
```

How do Nova policies update if I already have my own policy?

New CFEngine package updates will not overwrite your existing policy. That means that they will also not improve your current updating arrangement without your approval.

If you want to make use of CFEngine's enhancements to standard policy files like 'update.cf', you need to examine and integrate these changes to your policy server manually.

Update the software before updating policy, as new policy might require new features in the software.

If you require assistance upgrading, contact CFEngine Support.

Appendix A Manual package upgrade commands

Upgrading manually involved uninstalling and then reinstalling a package. You do not need to delete CFEngine working data when upgrading. Commonly used commands used to install packages:

Debian or Ubuntu:

```
dpkg --purge    cfengine-nova-old.deb
dpkg --install cfengine-nova-xxx.deb
```

Windows:

```
msiexec /qn /x cfengine-nova-old.msi
msiexec /qn /i cfengine-nova-xxx.msi
```

RPM families:

```
rpm -e --nodeps cfengine3-nova-old.rpm
rpm -ivh cfengine3-nova-xxx.rpm
```

FreeBSD:

```
pkg_delete cfengine3-nova-old.tbz
pkg_add cfengine3-nova-xxx.tbz
```

Solaris:

```
pkgrm -n -a cfengine3-nova-old.pkg

gunzip cfengine3-nova-xxx.pkg.gz \
&& pkgadd -d cfengine3-nova-xxx.pkg all
```

Note that repository commands like `yum` or `zypper` can not be used during manual upgrade, unless you have arranged a private local repository for the CFEngine Nova packages.

