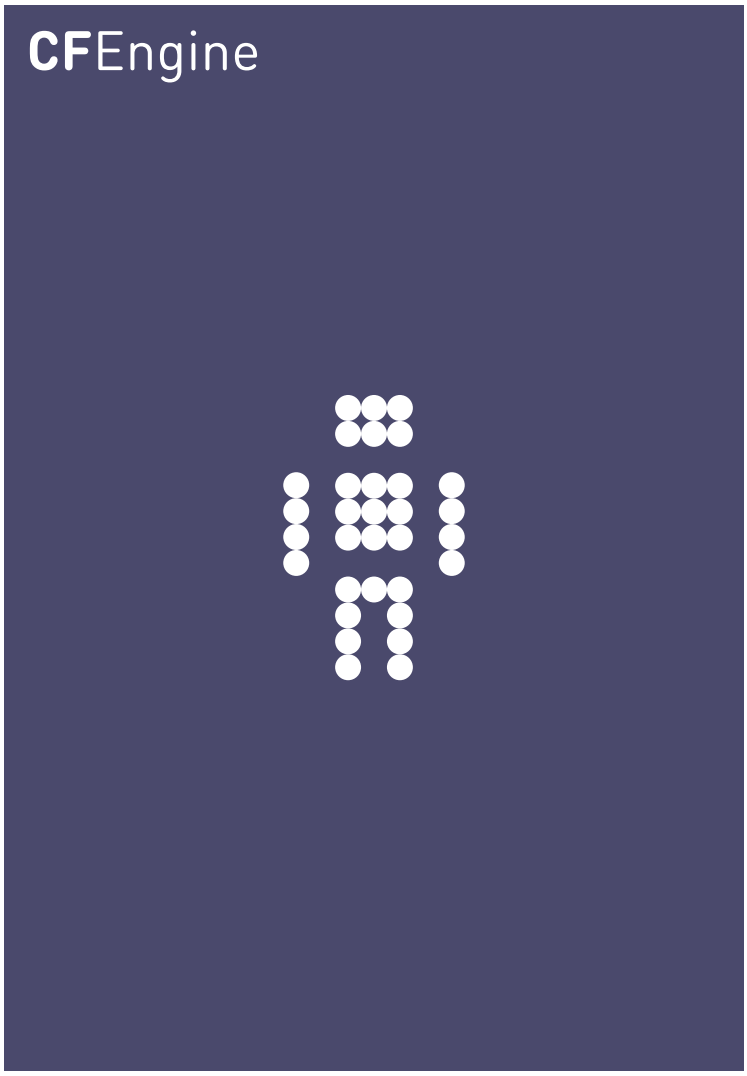


# Igniting CFEngine Nova!

A CFEngine Handbook

CFEngine AS

This short guide explains how to install the software and get it running.

# Table of Contents

1 Installing CFEngine Nova . . . . . 1
1.1 Installing the software on a new host . . . . . 1
1.2 Upgrading Nova . . . . . 3
1.2.1 Troubleshooting upgrade . . . . . 4
1.3 What is the default configuration? . . . . . 4
1.4 Frequently Asked Questions . . . . . 5
1.4.1 How do I install the prerequisites for the hub manually? . . . . . 5
1.4.2 Why do I get a promise failed with the message `Can't stat^^M/var/cfengine/master_software_updates/SOME-OS` on some hosts? . . . . . 5
1.4.3 I did bootstrap the hub *before* obtaining a license file, what should I do? . . . . . 6
1.4.4 On my hub, I get messages of connection failures to a database. For example, in `messages`, I can see something like `!! Could not open connection to report database for saving`. What should I do? . . . . . 6
1.4.5 How do I upgrade from community version to Nova? . . . . . 7
1.4.6 Let's say I'd like to deploy Nova on my Debian/Ubuntu network. Describe this step by step? . . . . . 7

# 1 Installing CFEngine Nova

CFEngine Nova is designed to be simple to install in its default configuration. The installation process has three phases:

- Unpacking the software.
- Obtaining a license.
- Adapting policy.

## 1.1 Installing the software on a new host

You should start from a blank system. If you have been using CFEngine Community Edition and you have already developed a policy, set aside this policy during the installation process. You will be able to integrate it back later.

Nova is provided in two packages, '`cfengine-nova`' and '`cfengine-nova-expansion`'. The main software package (for each operating system) must be installed on every host. The second package is only installed on the *hub* or *policy server*. You should install and set up the hub first.

The '`cfengine-nova`' package may be installed on a a wide range of supported operating systems, including Linux, Solaris, Windows, etc. The '`cfengine-nova-expansion`' package currently only supports the following Linux operating systems:

```
Debian
Ubuntu
Red Hat
SuSE
```

This means you must have one Linux computer running as your Nova hub. No special software is required on other machines in your network. CFEngine bundles all dependencies in the Nova package.

On a new host, installation follows the following procedure. References to package managers assume that additional packages might need to be installed of the policy server, e.g. the Apache Web Server, MySQL database, etc. Remember, root privilege is required for the installation.

1. Verify that the machine's network connection is working and that it can resolve names. On the hub, verify that package managers `yum`, `zypper` or `apt-get` are working. They will be used to install a web, database and php server (if not already installed). If you are not able to set up a package manager and repository on the hub, please look in the Frequently Asked Questions below for manual installation.
2. Copy the Nova packages to the system. On the hub or policy server:

   `cfengine-nova-2.`*`xxx.[rpm|deb]`*
   `cfengine-nova-expansion-2.`*`xxx.[rpm|deb]`*

   On all other machines:

   `cfengine-nova-2.`*`xxx.[rpm|deb]`*
3. Unpack the software:

   *Red Hat family*

   `host# rpm -ihv` *`packages`*

*SUSE family*

```
host# rpm -ihv packages
```

*Debian family*

```
host# dpkg --install  packages
```

4. On the hub, a public key has now been created in '`/var/cfengine/ppkeys/localhost.pub`' as part of the package installation. You should send this public key to CFEngine Support as an attachment in the ticket system, to obtain a license file '`license.dat`'. *** Save the returned license file to '`/var/cfengine/masterfiles/license.dat`' on the hub before continuing ***. See more details for the software licensing here; https://cfengine.com/software/Licensing.pdf
5. The remaining steps apply to all hosts, but you should install the hub or policy server first. Find the hostname or IP address of the hub (policy server), here we assume '`123.456.789.123`' is the address.

   ```
   hub # /var/cfengine/bin/cf-agent --bootstrap --policy-server 123.456.789.123
   ```

   Use the same command on all hosts, i.e. *** do not bootstrap the policy server with a localhost address *** If you mistype the address of the hub, we recommend doing the following steps to re-boostrap.

   ```
   hub # /var/cfengine/bin/cf-agent --bootstrap --policy-server 123.456.789.124
   hub # killall cf-execd cf-serverd cf-monoitord cf-hub
   hub # rm -rf /var/cfengine/inputs/*
   hub # rm -f /var/cfengine/policy_server.dat
   hub # /var/cfengine/bin/cf-agent --bootstrap --policy-server 123.456.789.123
   ```

6. The software should now be running.
7. To complete licensing setup, you should make a promise to accept the license terms by editing '`/var/cfengine/masterfiles/promises.cf`' on the hub (policy server), changing the line '`host_licenses_paid => "1";`' in '`body common control`' to reflect the correct number of licenses that you have subscribed to.

**How to assess success in this procedure:**

1. Look at the process list on the systems with '`ps waux | grep cf`'. You should be able to see `cf-execd` running, and eventually other processes from the CFEngine suite like `cf-monitord cf-serverd`. On the hub, you should also eventually see `cf-hub`. Note that it may take 5–10 minutes before all the processes get started.
2. Look for files in '`/var/cfengine/inputs`' (Unix) or '`C:\Program Files\Cfengine\inputs`' (Windows). The license file will be copied out from the policy server to the clients as part of the normal distribution of policy. Each Unix machine should get a copy of the '`license.dat`' file in '`/var/cfengine/inputs`' (Unix) or '`C:\Program Files\Cfengine\inputs`' (Windows).
3. On the hub, the file '`/var/cfengine/promise_knowledge.cf`' should have been created, and should contain data.
4. Finally, try to connect to the web server at port 80 on the hub / policy host. You should see a login screen, default user adn password is "admin" and "admin". See also separate document on the CFE Nova Mission Portal.


The opening page of the CFEngine Nova Mission Portal.

## 1.2 Upgrading Nova

When upgrading the software from an earlier version, you should upgrade the hub (policy server) machine first. Any other hosts in your network that act as servers for encrypted copy operations would preferably be upgraded first. This is because a stronger form of encryption-hash is used in newer versions, which the older servers cannot understand.

CFEngine packages its software in operating system compatible package formats (RPM, DEB, 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 sub-directory 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.
- Users are encouraged to take advantage of changes to the Nova base policy, e.g. the '`update.cf`' policy. We have reduced the number of files distributed and managed by CFEngine in 2.0.0. To gain the full benefits of the software, you should use the latest versions of these files:

```
cdp_inputs/  cfengine.cf         failsafe.cf     knowledge.cf  promises.cf   update.cf
cdp_lib/     cfengine_stdlib.cf  file_change.cf  OrionCloud/
```

  Note that some log files have new names, and that the audit database is no longer used by default. If you do not upgrade the files above, make sure you rotate the log files and perform garbage collection as detailed in these files.
- You can upgrade systems manually by copying the relevant package to the local host and running the appropriate package update command.
- 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. See full details for the software upgrade here; https://cfengine.com/software/UpgradingCfengineNova.pdf

### 1.2.1 Troubleshooting upgrade

You might experience some authentication errors during secure copy operations during upgrade from older versions of Nova. These will self-repair once the servers have been upgraded.

To avoid these altogether, you can temporarily disable copy verification, i.e. set

```
verify => "false";
```

in your `copy_from` bodies, should you wish. For standard installations, this setting is typically part of the definition of '`secure_cp`' defined in the '`cfengine_stdlib.cf`' file.

## 1.3 What is the default configuration?

Following the above procedure, you should have a fully functional CFEngine on all clients. However, in the default configuration, CFEngine does nothing other than looking after itself, and looking for possible policy updates from '`/var/cfengine/masterfiles`' on the policy server. The policy server is configured to collect data from non-policy server machines and generate reports that are integrated into the knowledge base.

To alter policies, you need to change the files on the policy hub, in the directory '`/var/cfengine/masterfiles`'. To begin with most of the policy examples are commented out in these files:

```
cdp_inputs/  cfengine.cf         failsafe.cf     knowledge.cf  promises.cf   update.cf
cdp_lib/     cfengine_stdlib.cf  file_change.cf  OrionCloud/
```

To change this, you can go to the main file '`promises.cf`', and include additional pre-made bundles of promises. You should always verify the contents of the bundles you include before activating and deploying to new machines.

```
bundle agent main
{
methods:

  any::

   "general" usebundle => def;

#   "jobs" usebundle => system_scheduling;
#   "security" usebundle => change_management;
#   "security" usebundle => security_files;
#   "windows boxes" usebundle => active_directory;

#  windows::
#   "windows boxes" usebundle => software_local;
#   "windows boxes" usebundle => app_baseline;
#   "windows boxes" usebundle => win_services;
#   "windows boxes" usebundle => win_registry;
#   "windows boxes" usebundle => win_emergency;

#  !windows::
#   "security"    usebundle => system_xinetd;
#   "maintenance" usebundle => garbage_collection;
}
```

## 1.4 Frequently Asked Questions

### 1.4.1 How do I install the prerequisites for the hub manually?

Here is a list of dependencies for the hub to be checked if The Mission Portal displays nothing;

- Red Hat/CentOS/Fedora

  `httpd, mysql, mysql-server, php, php-bcmath, subversion`
- SUSE

  `apache2, apache2-mod_php5, apache2-prefork, mysql, php5, subversion`
- Debian/Ubuntu

  `apache2, mysql-server, php5`

To install all of these, you might want to use `yum` on Red Hat/CentOS/Fedora, `zypper` on SUSE or `apt` on Debian/Ubuntu.

### 1.4.2 Why do I get a promise failed with the message `Can't stat /var/cfengine/master_software_updates/SOME-OS` on some hosts?

There is a built-in promise to automatically upgrade the Nova binaries. By default, the clients will check for an update package every time Nova runs. So if the clients find that there is no source directory to download the files from, the message will be displayed.

To fix the problem, simply create an empty directory mentioned in the message on the hub.

```
hub # mkdir /var/cfengine/master_software_updates/SOME-OS
```

### 1.4.3 I did bootstrap the hub *before* obtaining a license file, what should I do?

Four steps need to be followed to correct this minor issue.

1. obtain a working license file and copy it to '/var/cfengine/masterfiles'
   ```
   hub # cp /tmp/license.dat /var/cfengine/masterfiles
   ```
2. killall Nova running processes
   ```
   hub # killall cf-execd cf-serverd cf-monitord cf-hub
   ```
3. wipe out '/var/cfengine/inputs '
   ```
   hub # rm -rf /var/cfengine/inputs
   ```
4. bootstrap the policy hub
   ```
   hub # /var/cfengine/bin/cf-agent --bootstrap --policy-server 123.456.789.123
   ```

### 1.4.4 On my hub, I get messages of connection failures to a database. For example, in `messages`, I can see something like `!! Could not open connection to report database for saving`. What should I do?

This message comes from the `cf-hub` process. It is responsible for pulling reports from hosts that have contacted the hub to get policy updates. When these reports are fetched, they are stored in a local MongoDB database on the hub, and connecting to this database is what is failing.

Probably, the issue is that the database server is not running on your hub. Run the `ps` command to check this.

```
hub # ps -e | grep mongod
hub #
```

If the `mongod` process *is* running, it must be misconfigured or in some bad state. Please look at the newest entry in '/var/log/mongod.log' to diagnose the problem, and contact CFEngine Technical Support if necessary.

If the `mongod` process *is not* running, please follow the steps below.

- Run hub # `/var/cfengine/bin/cf-twin -Kvf failsafe.cf > /tmp/cfout`
- Check again if the `mongod` is running, if so, the problem is probably fixed now.
- If `mongod` is still not running, please search the output file for lines starting as follows.

  ```
  ...
  nova>  -> Making a one-time restart promise for mongod
  ...
  ...
  nova>  -> Executing '/var/cfengine/bin/mongod....
  nova>  -> Backgrounding job /var/cfengine/bin/mongod...
  nova>  -> Completed execution of /var/cfengine/bin/mongod...
  ...
  ```

  If you don't see the first line above, Nova does not try to start `mongod` — so check if you bootstrapped your hub correctly. If you see all lines, Nova starts `mongod`, but the process just terminates immideatley after. If so, continue to the next step.
- Look at the newest entry in '/var/log/mongod.log'. It should give you more details of why the `mongod` process refuses to start. The two most common scenarios are described next.

- If `mongod` has been terminated unexpectedly, it might have left a lock-file behind that stops it from starting again. Try deleteting '`/var/cfengine/state/mongod.lock`' if it exists.
- If the database is corrupted, you can have '`mongod`' create a new one my moving '`/var/cfengine/state/cf-report.*`' out of the way. There are also tools and documentation for repairing a database at http://www.mongodb.org/.

  Note that almost all of the `cfreport` database is completely recreated with data collected from clients every six hours, so deleting it is normally acceptable. But CFEngine AS or CFEngine Inc can not be held responsible for data loss in this respect.

### 1.4.5 How do I upgrade from community version to Nova?

There is no shortcut for this task. We urge you to set aside your current community policy while you install Nova, setup the Nova hub by following this/Nova supplement document, and then integrate your existing policy to the hub manually, in small testable steps.

CFEngine Nova is compatible with the Community Edition of CFEngine 3, but some process files are now managed by CFEngine for user convenience.

### 1.4.6 Let's say I'd like to deploy Nova on my Debian/Ubuntu network. Describe this step by step?

Here we go:

**Debian/Ubuntu Installation Example:**

- **Hub(policy-server)**
  1. Verify that the package manager is working (eg. `apt-get update`)
  2. Download the Nova and Nova Supplement package
  3. Unpack the software:

     ```
     hub # dpkg --install cfengine-nova_2.0.1-1_x86_64.deb
     hub # dpkg --install cfengine-nova-expansion_2.0.1-1_x86_64.deb
     ```
  4. Send the file: `"/var/cfengine/ppkeys/localhost.pub"` to Cfengine Support (OTRS ticket system)
  5. You will receive a license file: `license.dat`
  6. Copy the license file to: `"/var/cfengine/masterfiles/license.dat"`
  7. Bootstrap the hub:

     ```
     hub # /var/cfengine/bin/cf-agent --bootstrap --policy-server <ip address of the hub
      *** Warning: do not use 127.0.0.1 as the ip address ***
     ```
  8. Open `"/var/cfengine/masterfiles/promises.cf"`, and change `'host_licenses_paid => "<Number of licenses subscribed>";'`
- **Clients**
  1. Verify that the package manager is working (eg. `apt-get update`)
  2. Download the Nova package
  3. Unpack the software:

     ```
     hub # dpkg --install cfengine-nova_2.0.1-1_x86_64.deb
     ```
  4. Bootstrap the Client to the hub:

```
client # /var/cfengine/bin/cf-agent --bootstrap --policy-server <ip address of the
```

To check if the installation went as expected, see "How to assess success in this procedure" section in the Nova_Supplement Guide.