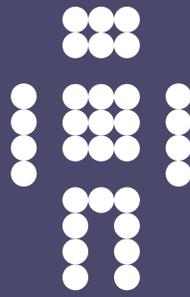


CFEngine



CFEngine Nova Mission Portal

CFEngine Enterprise Documentation
for version 2.1.0, 19. September 2011

CFEngine

Table of Contents

1	Introduction	1
2	Mission Portal	3
2.1	Mission Portal Rooms	3
2.1.1	Mission Status	4
2.1.2	Mission Engineering	5
2.1.3	Mission Planning	7
2.1.4	Mission Library	8
2.2	Finders	8
2.2.1	Host finder	9
2.2.2	Class finder	9
2.2.3	Promise finder	9
2.2.4	Report finder	10
2.2.5	Summary report finder	11
2.2.6	CDP report finder	11
2.2.7	Topic finder	12
2.3	Viewers	13
2.3.1	Host viewer	13
2.3.2	Bundle viewer	13
2.3.3	Promise viewer	15
2.3.4	Vital signs viewer	16
2.3.5	Topics viewer (Knowledge map)	16
2.3.6	Weakest hosts viewer	17
2.3.7	Report viewer	18
2.4	Editors	19
2.4.1	Policy editor	19
2.4.2	Integration with subversion	20
2.5	Mission Portal Administration	21
2.5.1	Mission Portal Settings	21
2.5.2	Mission Portal User Admin	21
2.5.3	Troubleshooting - Mission Portal Logs	22
Appendix A	Configuration of external authentication	23
A.1	Configure LDAP	23
A.2	Configure Active Directory	24
A.3	Directory structure	24

1 Introduction

CFEngine Nova is a commercially licensed version of the core CFEngine software¹ with enterprise library extensions. All of the documentation for CFEngine 3 applies to CFEngine Nova. This document is a supplement describing the graphical user interphase used in CFEngine Nova: the Mission Portal.

Knowledge management has become a new focus in IT management, and it is a core focus at CFEngine. Comprehending the growing complexity of IT operations is one of the main challenges in IT today. CFEngine Nova has a number of key features for knowledge management, including automated documentation, report generation and associative inference. The CFEngine Mission Portal is the centerpiece of user interaction with CFEngine Nova and can be accessed after the successful installation of the software on your hub server (policy host). Connect to the server (port 80) with your web browser (for example at <http://123.456.789.123>), you should see a login screen:



Figure: Mission Portal login screen

Default user and password is 'admin' and 'admin' (make sure to change this at first login to prevent unauthorized access).

¹ Major version 3

2 Mission Portal

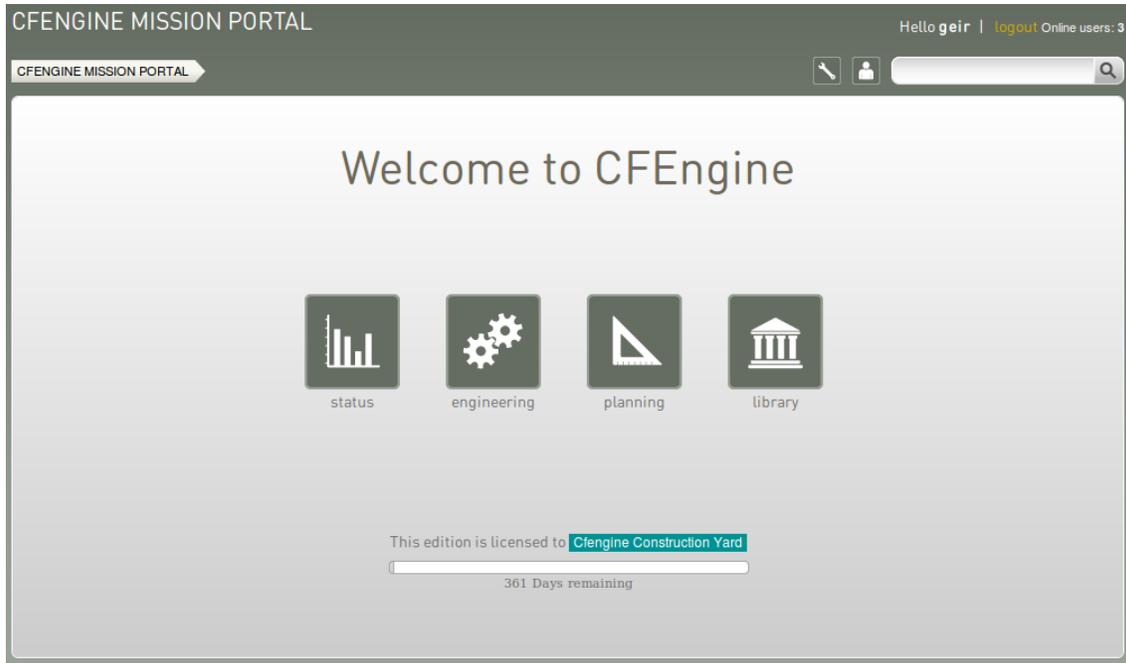


Figure: The mission portal

2.1 Mission Portal Rooms

There are four main rooms in the Mission Portal that offer insight into different aspects of operations:

- Mission status: a top level overview of compliance status and business value
- Mission engineering: a place to see the current state of system repair
- Mission planning: a place to plan and make policy changes
- Mission library: a knowledge bank that connects information together

Each of these rooms is a beginning from which you can refine your overview and search through information.

2.1.1 Mission Status

Mission status is a high level summary of how well the entire system is behaving.

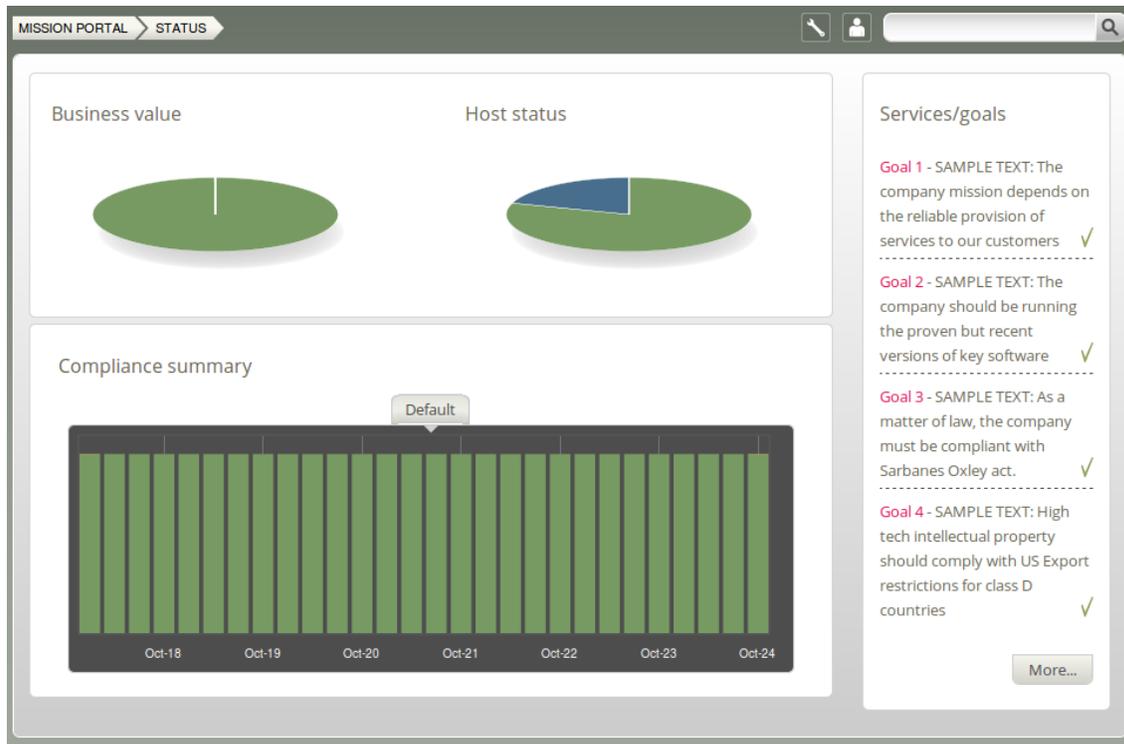


Figure: The status of IT operations.

Business Value and Host Status: The two pie charts show the business value of the promises kept/not kept and well as host status, respectively. Business value is associated with the value of promises as defined in policy files. In the Host Status chart, each host represents a slice of the pie and is classified into red, yellow, green and blue according to the level of their compliance. A host is red if less than 80% of its promises are kept, yellow if 20% or more of its promises were repaired and host is now compliant, green if more than 80% of its promises are kept, and blue if there is no contact between the hub and the client host.

Compliance Summary: The row of bar meters shows the compliance (average percentage of promises kept, repaired or not kept) of all registered hosts¹ in blocks of 6 hours for the past week. It summarizes performance and anomalous behavior in a simple red (prommises not kept), yellow (promises repaired) and green (promises kept) scale. Click on a bar to see which promises were kept/not kept.

Services/Goals: A summary of Mission goals (as defined in user policy files; these examples are from 'company_knowledge.cf'). Edit the file in the policy editor (Planning room -> repository), or edit the file in your own text editor, to change these goals.

¹ Blue hosts will not appear here

2.1.2 Mission Engineering

Mission engineering illustrates the state of the system in relation to the desired state at all scales. Zoom in to specific areas and examine the impact of promises, query data, and extract reports.

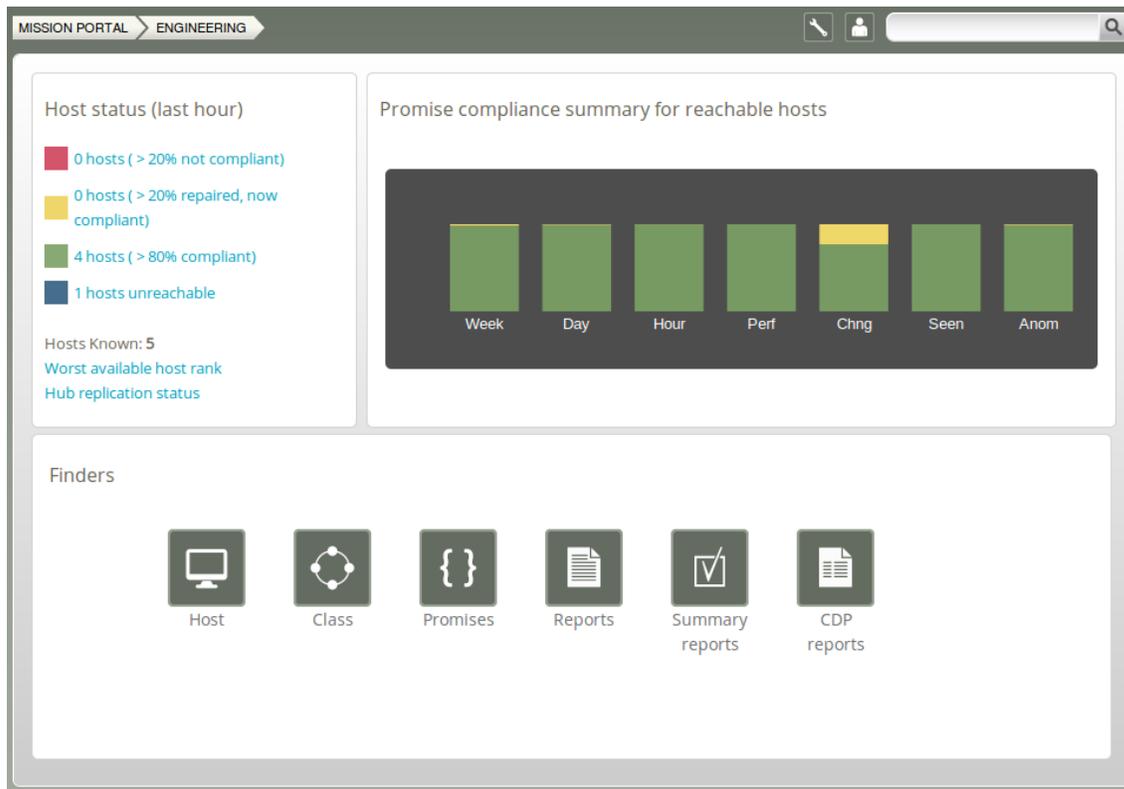


Figure: Mission Engineering

Host Status:

- The hosts are classified into red, yellow, green and blue according to the status of their compliance. A host is red if less than 80% of its promises are kept, yellow if 20% or more of its promises were repaired and host is now compliant, green if more than 80% of its promises are kept, and blue if there is no contact between the hub and the client. Clicking a link produces a list of the hosts in that category.
- Worst available host rank: Display the weakest hosts (that have been in contact with the hub) over the last hour.
- Hub replication status: Display status of redundant monitoring hubs (if activated).

Promise compliance summary for reachable hosts: The row of bar meters shows the compliance (average percentage of promises kept, repaired or not kept) of all registered hosts over the past week, the past day and the past hour. It summarizes performance and anomalous behavior in a simple red (promises not kept), yellow (promises repaired), and green (promises kept) scale. The "Chng" bar relates to the amount of changes made to files monitored by a CFEngine policy in the last hour (change watch). It is green if no changes have been made. The level of yellow increases as changes occur (but it will never be red). For the "Seen" bar, CFEngine monitors the average time between connections to

the clients; a last seen promise is considered not kept if the hub has not been in touch with a host for said average time plus two standard deviations, repaired for said average time plus the interval between one and two standard deviations, and kept for said average time plus the interval between zero and one standard deviation. The "Anom" bar relates to anomalies and is generated from monitoring data (vitals) for the last week in the following fashion: all observations (over the whole week) that are more than two std deviations away from the average is counted as not kept. They are counted as repaired if the distance is between one and two standard deviations from the average. Otherwise, they are kept & green.

Finders: The Mission Engineering room comes with six finder functions (modules that make it simple and intuitive to browse and search for objects of a particular type, see section on Finders): host, class, promises, reports, summary reports, and CDP (Content Driven Policies) reports.

2.1.3 Mission Planning

Make changes to policies, goals determined by promises and implement specific tactics to achieve the desired state. Interact with data, approve changes and anomalies. Get an overview of users logged on to the Mission Portal, as well as their current activity.

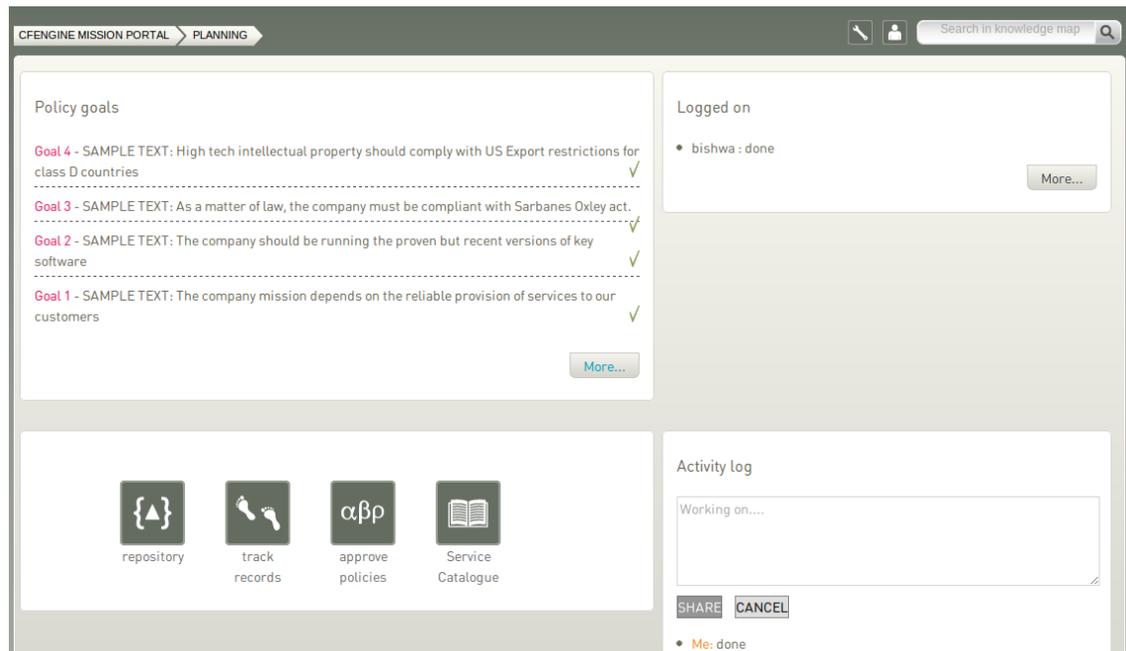


Figure: Mission Planning

Policy Goals: List of policy goals as defined in policy files; these examples are from 'company_knowledge.cf'. Edit the file in the policy editor (Planning room -> repository) or edit the file in your own text editor. The "More..." button links to the Service Catalogue, click to see which bundles contribute to these policy goals.

Action buttons:

- Repository: Edit policy files in the integrated policy editor (requires Subversion)
- Track records: Overview of promises repaired or not kept
- Approve policies: To be developed
- Service catalogue: See which bundles contribute to policy goals

Logged on: Shows users currently logged on to the Mission Portal and their activity.

Activity log: Shows the latest activity entries. Type in a new activity to keep colleagues posted on current work.

2.1.4 Mission Library

The Library contains finders for documents, topics, a notes archive, and (external) link to the CFEngine community.

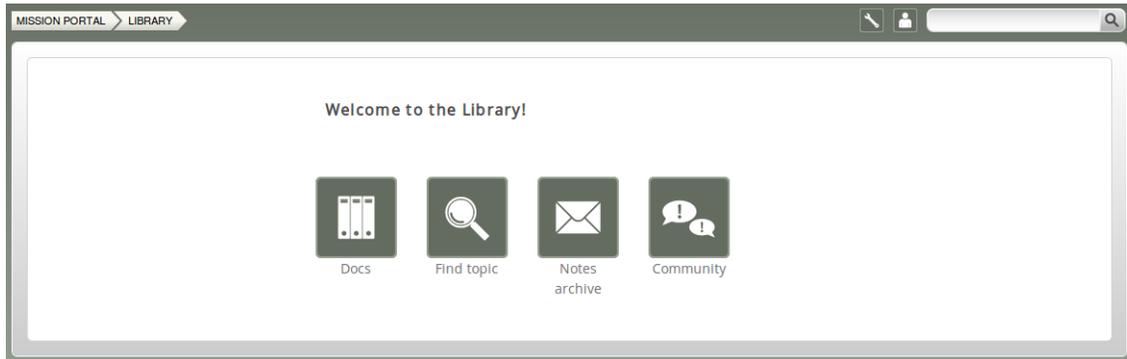


Figure: Mission Library

Library buttons:

- Docs: Overview of documentation that was packaged with CFEngine Nova.
- Find Topic: Search for topics either by scrolling through the alphabetical list or by typing in the search box (same as the search box on top right of page).
- Notes Archive: Get overview of all notes made in regard to hosts or reports.
- Community: External link to the CFEngine community

2.2 Finders

Finders are modules that make it simple and intuitive to browse and search for objects of a particular type. Some are located in the Mission Engineering room, others in the Mission Library or the generic search box in the top right corner.

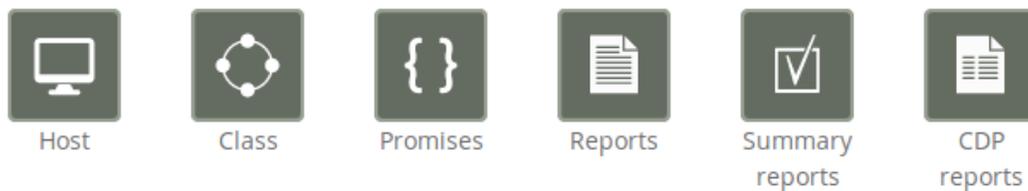


Figure: Finders in the Mission Engineering room

2.2.1 Host finder

The host finder is located in the Mission Engineering room and will display a list of hosts. Browse by scrolling through the list, click a letter corresponding to the first letter of a host name, or search for hosts in the search box (top right corner). Clicking on a host name will bring you to the host viewer.



Figure: Host finder

2.2.2 Class finder

The class finder is located in the Mission Engineering room and will display a list of classes. Browse by scrolling through the list, click a letter corresponding to the first letter of a class name, or search for classes in the search box (top right corner; choose between searching all, time, soft or IP classes). Clicking on a class will bring you to a report for that class profile.

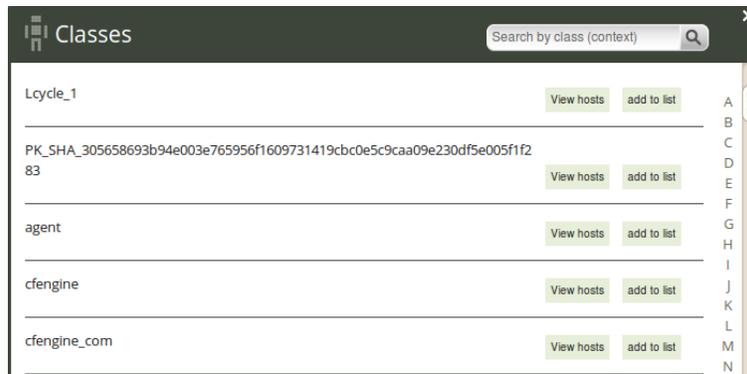


Figure: Class finder.

2.2.3 Promise finder

The promise finder is located in the Mission Engineering room and will display a list of promises. Browse by scrolling through the list, click a letter corresponding to the first letter of a promiser/bundle/handle name (set alternative in searchbox and click a letter in the right column), or search for promiser/bundle/handle in the search box (top right corner; choose between searching promiser, bundle, or handle). Clicking on a promise/bundle will bring you to the promise/bundle viewer.

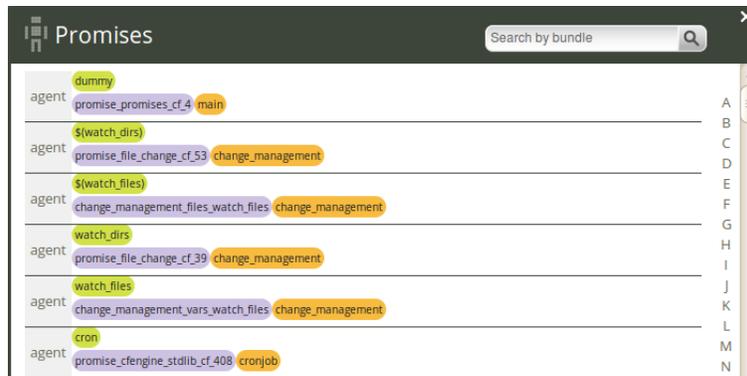


Figure: Promise finder

2.2.4 Report finder

The report finder is located in the Mission Engineering room and will display a list of standard report categories. Browse by scrolling through the list. Clicking on a report category will bring up a query table that will produce a report according to the entered query criteria.

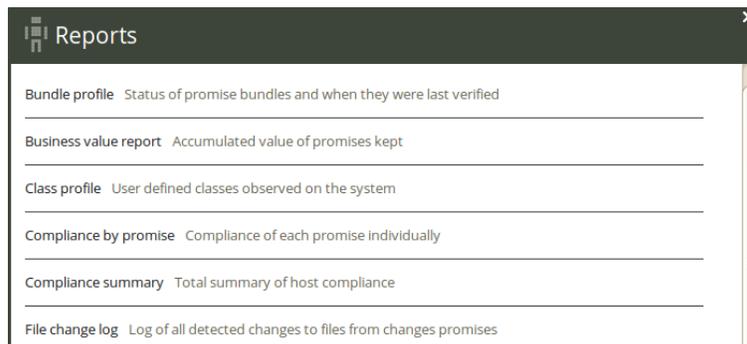


Figure: Report finder

Standard reports:

- Bundle profile: Status of promise bundles and when they were last verified
- Business value report: Accumulated value of promises kept
- Class profile: User defined classes observed on the system
- Compliance by promise: Compliance of each promise individually
- Compliance summary: Total summary of host compliance
- File change log: Log of all detected changes to files from changes promises
- File change diffs: Delta/difference comparison showing file changes
- Last saw hosts: Log of when neighbouring hosts were last observed online
- Patches available: A list of patches currently claimed to be available by the local package manager
- Patch status: A list of (un)applied patches according to the local package manager
- Performance: Execution time used to verify selected promises
- Promises repaired log: Log of actual repairs made to the system
- Promises repaired summary: Cumulative (histogram) summary of promises repaired
- Promises not kept log: Log of promises that could not or would not be kept

- Promises not kept summary: Cumulative (histogram) summary of promises not kept
- Setuid/gid root programs: Current list of observed setuid/setgid root programs
- Software installed: List of software packages claimed to be installed according to the local package manager
- Variables: Table of variable values last observed

2.2.5 Summary report finder

Find the current state of knowledge about promise compliance. Search by promise handle and/or host group (host class) and/or specific host. Click the `Help?` link to open the Promise finder or Classes finder to view available promise handles or host groups (host classes), respectively.

Figure: Summary report finder

The result shows the number of hosts and their compliance when a single promise handle is specified, and an expectation value¹ of the number of hosts that match a search criteria when a search string (regex) is specified. Click `New search` to change or refine the query.

Compliant hosts	Non-compliant hosts	Total hosts verified	Unreachable hosts	This was last verified between
3	0	3	500	Mon August 01 09:00:46 2011 - Mon August 01 09:11:18 2011

Classes : any

Promise handles :
update_files_usr_local_sbin

Figure: Summary report

2.2.6 CDP report finder

A Content-Driven Policy (CDP) is a text file with lines containing semi-colon separated fields, like a spreadsheet or tabular file. Each line in the file is parsed and results in a specific type of promise being

¹ Predicted mean in the time frame shown

made, depending on which type the Content-Driven Policy is. The CDP report finder lists template CDPs that come with CFEngine Nova, click to view a report on the selected topic.

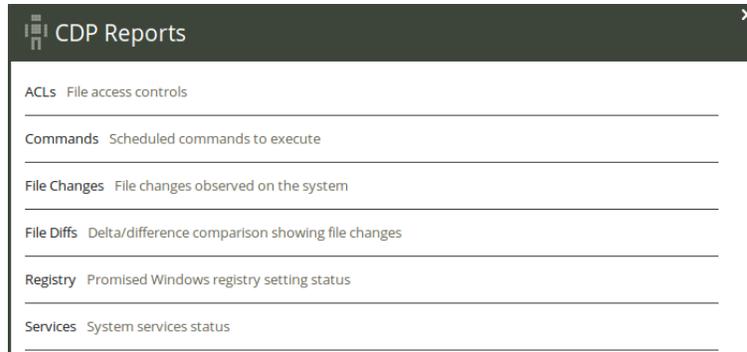


Figure: CDP report finder

2.2.7 Topic finder

The Topic finder is located in the Mission Library and will display a list of common topics. Browse by scrolling through the list, click a letter corresponding to the first letter of a topic name, or search for topics in the search box (top right corner). Clicking on a topic will bring you either to a document, web page or the Topic viewer (Knowledge map).

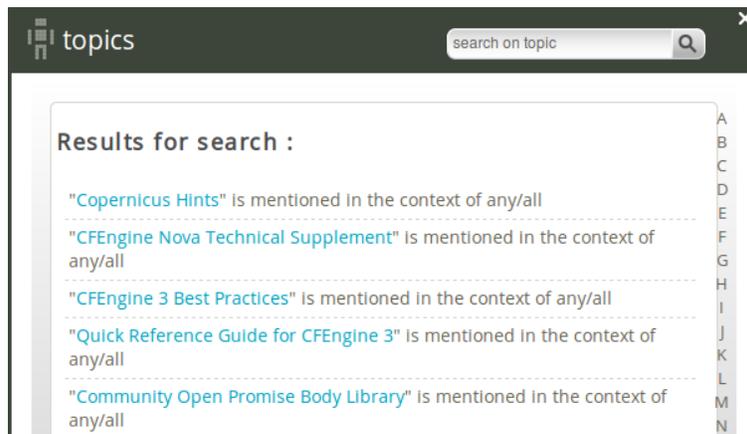


Figure: Topic finder

2.3 Viewers

Viewers show info about the main objects at different scales of the system.

2.3.1 Host viewer

Shows information about hosts, including name, status, operating system, vital signs, promises not kept, standard and custom reports, and more. View and make notes about the host.

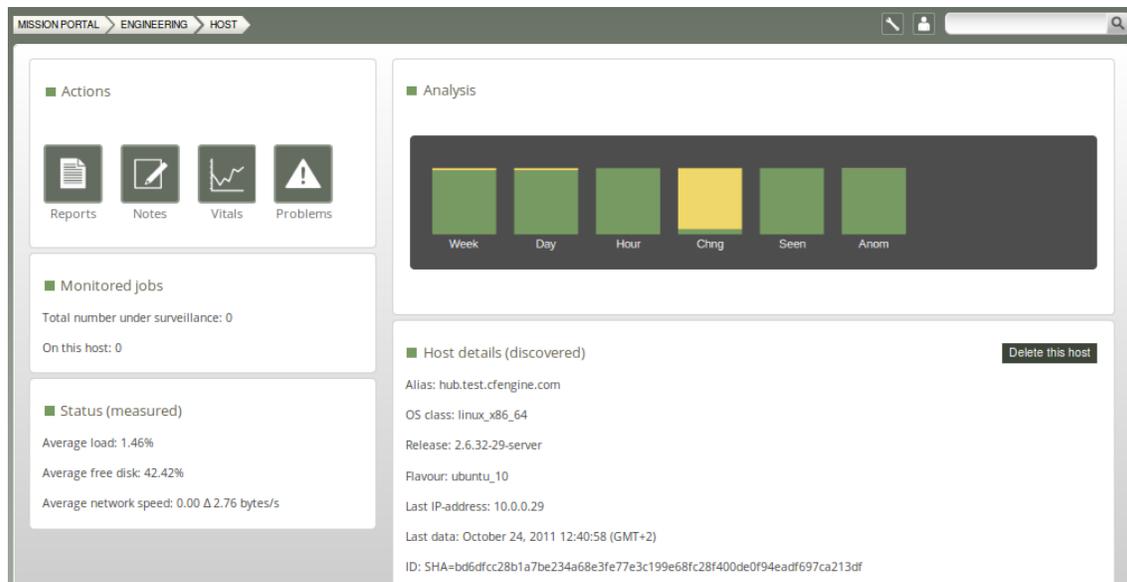


Figure: Host viewer

Action buttons:

- Reports: Tabular summary of the host's internal information, tailored to a particular topics
- Notes: View and make comments about this host
- Vitals: Overview of monitoring data for this host and its current performance statistics
- Problems: Overview of promises not kept by this host

Monitored jobs: To be developed.

Analysis: The bar meter shows the host-summary status of a number of key performance indicators.

- Week: The average level of promise-compliance over the whole past week.
- Day: The average level of promise-compliance over the past day.
- Hour: The average level of promise-compliance over the past hour.
- Perf: The average performance status of the system, compared to the learned norm.
- Chng: Software update status of the system (only shows on hub, not displayed on clients).
- Seen: The average level of connectivity compliance (to the hub) over the past week .
- Anom: Level of anomalous site-wide activity on the system.

2.3.2 Bundle viewer

The bundle viewer provides an interface to explore the context (class) in which a bundle is used and the promises made within that bundle. Tabs display other bundles using the one currently viewed and

a general overview of all bundles. Navigate to interact with other views and get a complete picture of context, leads, references, affected objects, similar topics, and more.

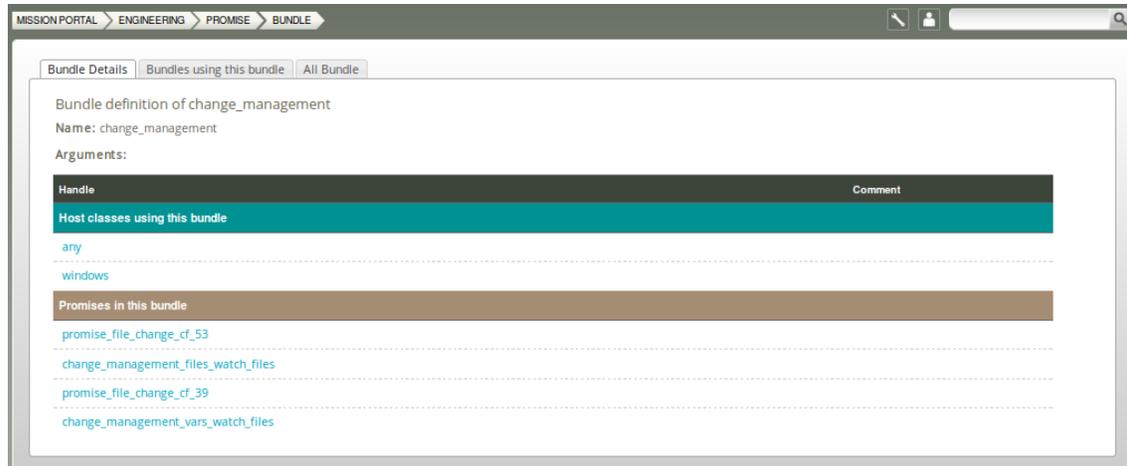
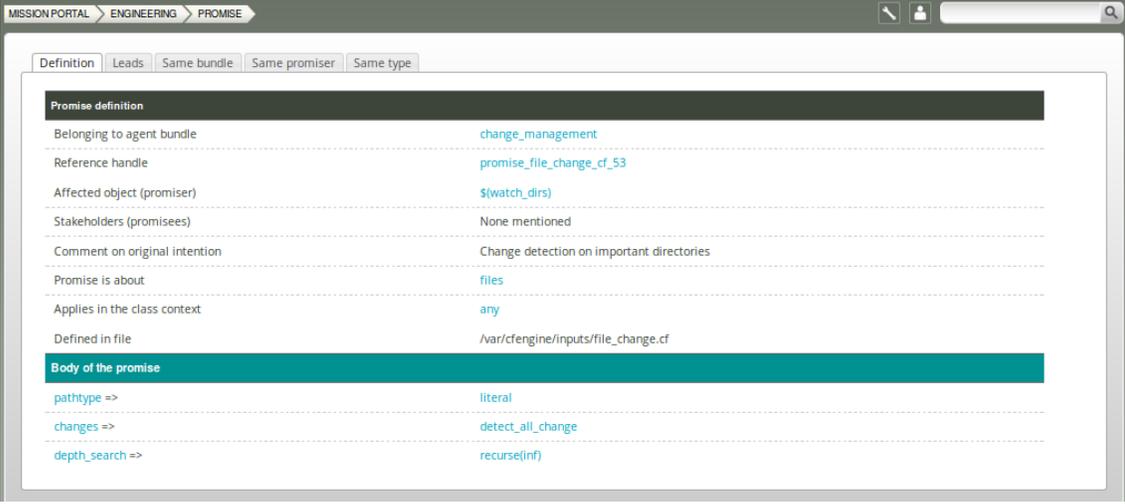


Figure: Bundle viewer

2.3.3 Promise viewer

The promise viewer shows a promise definition and body. There are tabs for viewing leads (promise type, context, dependencies), other promises used in same bundle, other promises made by same promiser, and other promises of same type.



Promise definition	
Belonging to agent bundle	change_management
Reference handle	promise_file_change_cf_53
Affected object (promiser)	\${watch_dirs}
Stakeholders (promisees)	None mentioned
Comment on original intention	Change detection on important directories
Promise is about	files
Applies in the class context	any
Defined in file	/var/cfengine/inputs/file_change.cf

Body of the promise	
pathtype =>	literal
changes =>	detect_all_change
depth_search =>	recurse(Inf)

Figure: Promise viewer

2.3.4 Vital signs viewer

The Vital signs viewer shows an overview of monitoring data from each host and its current performance statistics. In order to see data in these graphs, each host in the CFEngine managed network must be running `cf-monitor` and `cf-serverd`. This is the default behavior for a Nova installation.

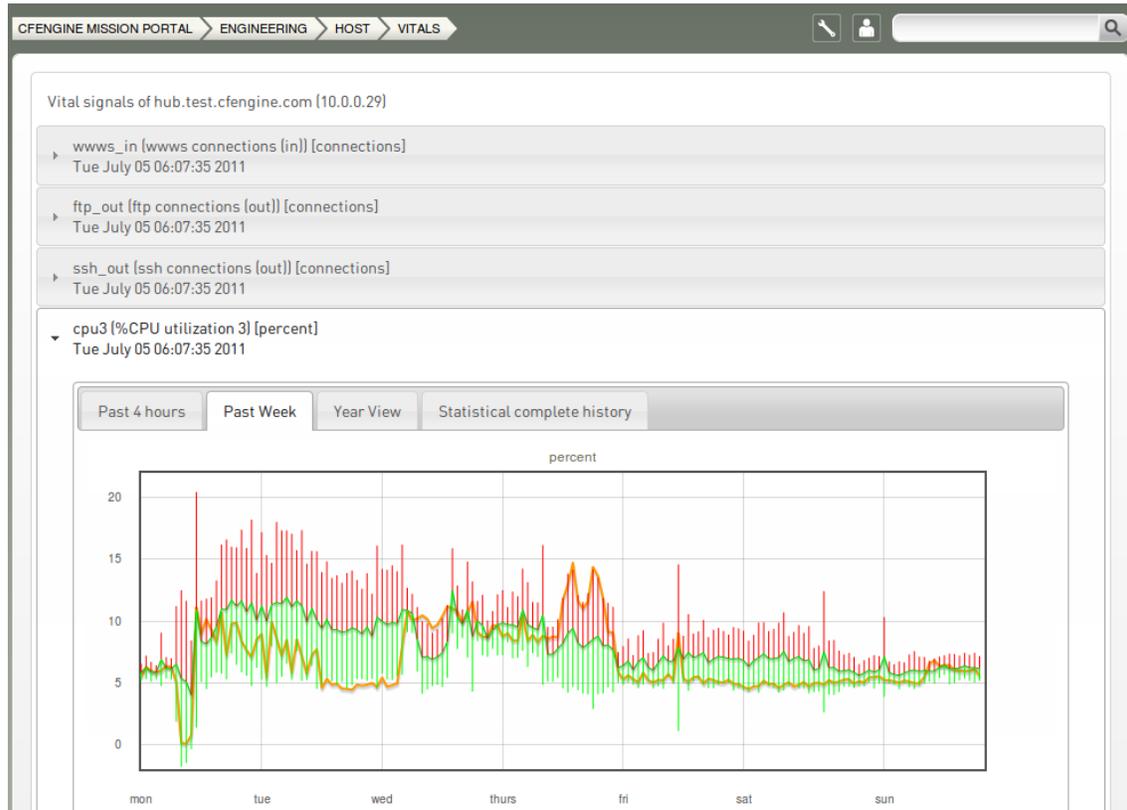


Figure: Vital signs viewer

2.3.5 Topics viewer (Knowledge map)

The Topics viewer, or Knowledge map, is a *semantic web* of subject references and document pointers. In a semantic web, you are presented with links to documents about your chosen topic, as usual; in addition you are offered *leads* and possible pathways to topics that are known to be related. These leads don't just point you to more documents, but explain *how* neighboring issues are related. The aim is to help the user learn from the experience of browsing, by conveying the meaning of the current topic in relation to other issues in the system. This is how *knowledge transfer* occurs.

The Knowledge Map can be found by searching for a topic in the top right corner or through the topic finder in the Mission Library (this will sometimes also lead directly to a document or web page instead).

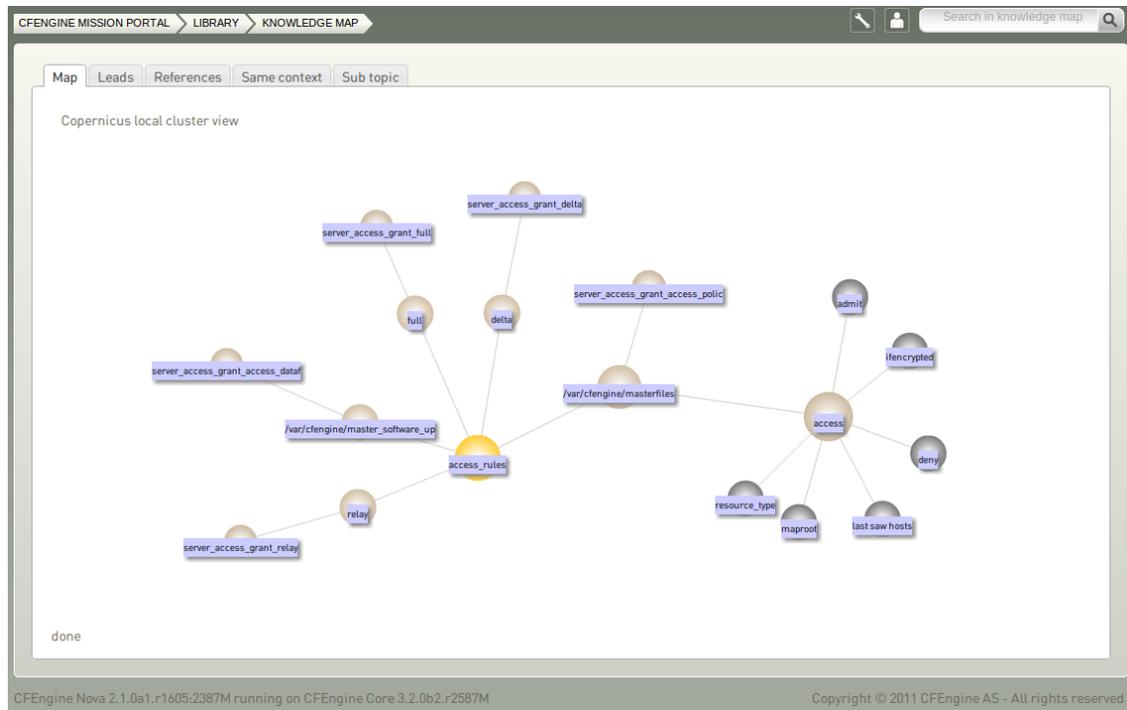


Figure: Topics viewer (Knowledge map)

The yellow sphere represents the current topic, surrounding (blue) spheres represent related topics, and the size of the spheres represents the number of associations each topic has. This map is navigable: click on a different topic to see a new view centered on that topic and its associations. The tabs will show leads, references, topics in the same context, and sub topics in the same context. Links can also lead to other viewers in the Mission Portal, documents and web pages related to the current topic.

2.3.6 Weakest hosts viewer

A link to the worst ranked hosts is located in the Mission Engineering room. This ranks all hosts that have reported to the hub¹ according to their level of promises not kept during the last hour, with the worst host on top.

¹ Blue hosts will not appear

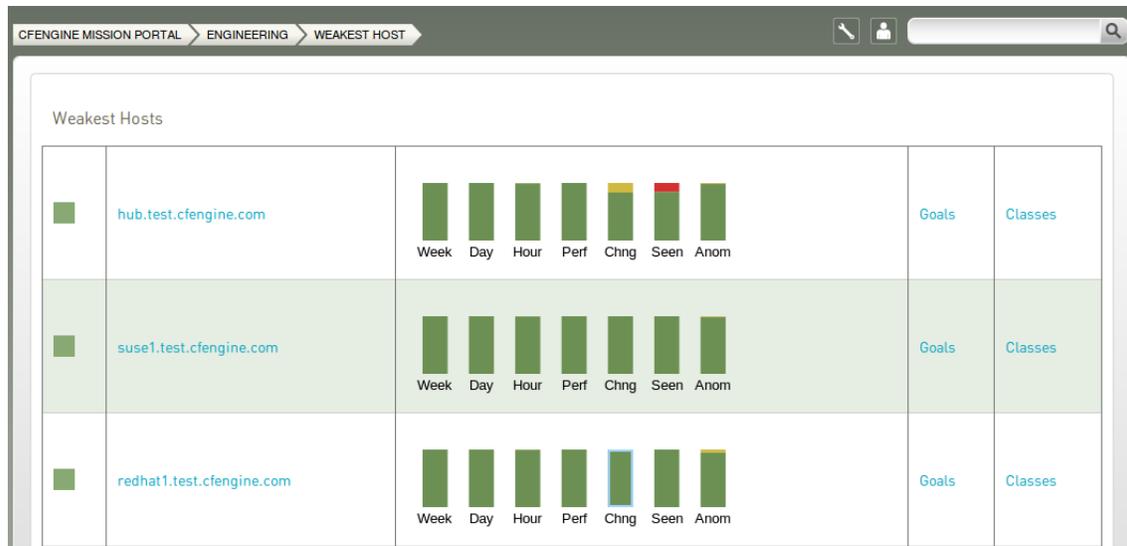


Figure: Weakest hosts

2.3.7 Report viewer

A significant capability of CFEngine Nova is the existence of automated system reporting. A report is a tabular summary of CFEngine's internal information, tailored to a particular purpose. Reports describe attributes and qualities of managed hosts and can be filtered through the text boxes on top of each column.

HOST	PROMISE HANDLE	LAST KNOWN STATE	PROBABILITY KEPT	UNCERTAINTY	LAST SEEN
policy.test.cfengine.com	cfengine_correct_cftwin_files_cftwin	Compliant	100	0	October 24, 2011 13:01:18 (GMT+2)
policy.test.cfengine.com	cfengine_correct_cftwin_files_lbtwin	Not Compliant	0	0	October 24, 2011 13:01:18 (GMT+2)
policy.test.cfengine.com	cfengine_limit_robot_agents_processes_check_cf_execd	Compliant	100	0	October 24, 2011 13:01:18 (GMT+2)
policy.test.cfengine.com	cfengine_limit_robot_agents_processes_check_cf_monitor	Compliant	100	0	October 24, 2011 13:01:18 (GMT+2)
policy.test.cfengine.com	cfengine_management_correct_cftwin	Compliant	100	0	October 24, 2011 13:01:18 (GMT+2)

Figure: Report viewer

Reports are updated at different intervals, the default values are every 5 minutes or 6 every hours (this can be changed by the user). Below is a list of standard reports, updated every 5 minutes unless otherwise noted:

- Bundle profile: Status of promise bundles and when they were last verified
- Business value report: Accumulated value of promises kept (6 hrs)
- Class profile: User defined classes observed on the system
- Compliance by promise: Compliance of each promise individually

- Compliance summary: Total summary of host compliance
- File change log: Log of all detected changes to files from changes promises
- File change diffs: Delta/difference comparison showing file changes
- Last saw hosts: Log of when neighbouring hosts were last observed online
- Patches available: A list of patches currently claimed to be available by the local package manager (6 hrs)
- Patch status: A list of (un)applied patches according to the local package manager (6 hrs)
- Performance: Execution time used to verify selected promises
- Promises repaired log: Log of actual repairs made to the system
- Promises repaired summary: Cumulative (histogram) summary of promises repaired
- Promises not kept log: Log of promises that could not or would not be kept
- Promises not kept summary: Cumulative (histogram) summary of promises not kept
- Setuid/gid root programs: Current list of observed setuid/setgid root programs (6 hrs)
- Software installed: List of software packages claimed to be installed according to the local package manager (6 hrs)
- Variables: Table of variable values last observed (6 hrs)

2.4 Editors

2.4.1 Policy editor

The CFEngine Nova Mission Portal provides an editor for working on CFEngine language. The editor provides syntax high-lighting and look-up to make working with CFEngine's extensive language easier. There is a tie-in for Subversion version control repositories; the Mission Portal will prompt you for the path and login credentials. Setup of a subversion repository has to be done separately.

The main key commands in the editor Window are:

Autocompletion: Ctrl+Space

Shows a pop-up menu of possible items. This is context sensitive, e.g. it also works inside lists (e.g. `bsdflags`) to provide possible values.

Undo: Ctrl+Z

In Safari, Ctrl-backspace may be used.

Redo: Ctrl+Y

Undo an undo operation, i.e. reverse the direction of transaction roll.

Indent: TAB

Format a file to a standard indentation.

Multiple documents appear as tabs along the top of the screen.

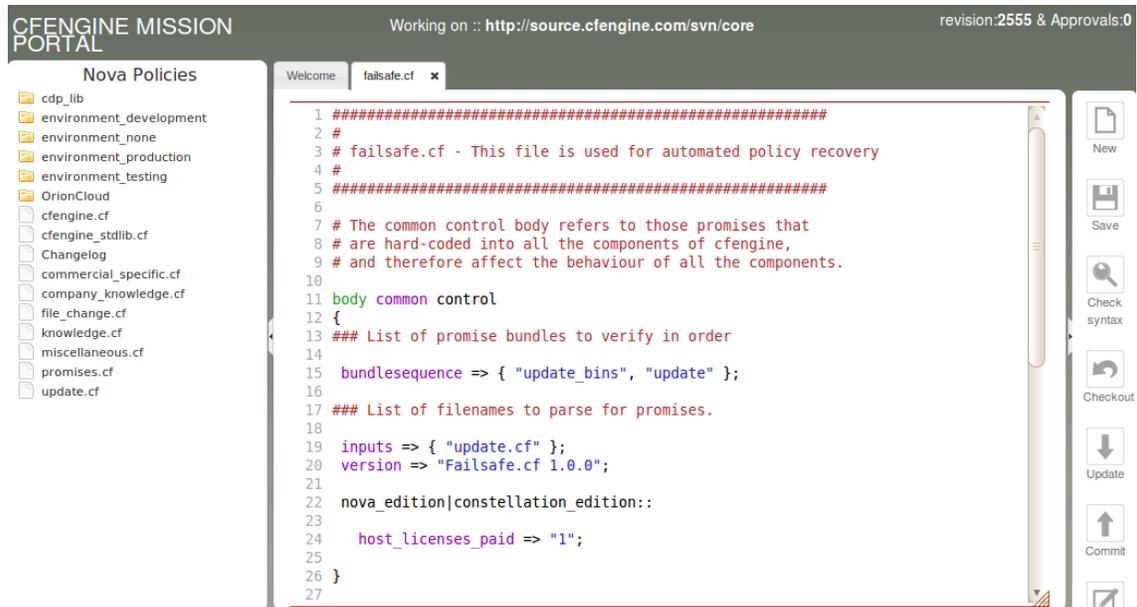


Figure: The Policy Editor

The CFEngine Nova policy editor detects syntax errors and highlights these in red to avoid mistakes when editing. In addition, by using the **Check syntax** button, it is possible to pre-test the policy before committing changes to a repository. This will run `promises.cf` through the `cf-promises` parser.

The main menu on the left hand panel shows the main workflow items for policy editing. Clicking the arrow in the panel divider collapses the menu and gives full-screen editing.

The right hand panel shows basic file and Subversion commands. The **Save** button will store a local/version of the opened files without committing to the repository.

2.4.2 Integration with subversion

The default architecture proposes to work in close dialogue with a version repository. CFEngine recommends the use of such a repository and currently supports subversion.

For the moment, user management in the Mission Portal only concerns access to the Mission Portal settings. It does not affect the ability to edit or approve policies in the policy editor. Access control and authorization for policies are handled by subversion's authentication system so that edits made in the Mission Portal can be interleaved with edits from other sources. The subversion repository thus becomes the centerpiece of the distributed coordination, as is the intended function of version control systems.

When working with a subversion repository, you first do a "checkout". This allows you to select the repository you want to download and creates a working environment based on it. Note that the editor currently only views files in the checked out repository, so if you have a directory hierarchy, you need to check out multiple times. After doing a checkout, you can view the path to the current repository at the top.

The "update" command downloads the newest files from the already checked out subversion repository, while "commit" updates the repository with your working copy. You also have the ability to see the repository log of the 20 last changes. This includes changed files, users and log messages. For more detailed information on working with subversion, see <http://svnbook.red-bean.com>.

After editing you may or may not commit changes. When closing an edited file, you are also given the option to commit at once.

The menu choice to commit a current version commits all files in the configuration repository. Commits require a mandatory comment. CFEngine recommends you use this to explain why a change was made, or relate it to an incident number, etc. The details of *what* was changed are already documented by the version control logs.

2.5 Mission Portal Administration

2.5.1 Mission Portal Settings

Configure and view various settings for the Mission Portal.

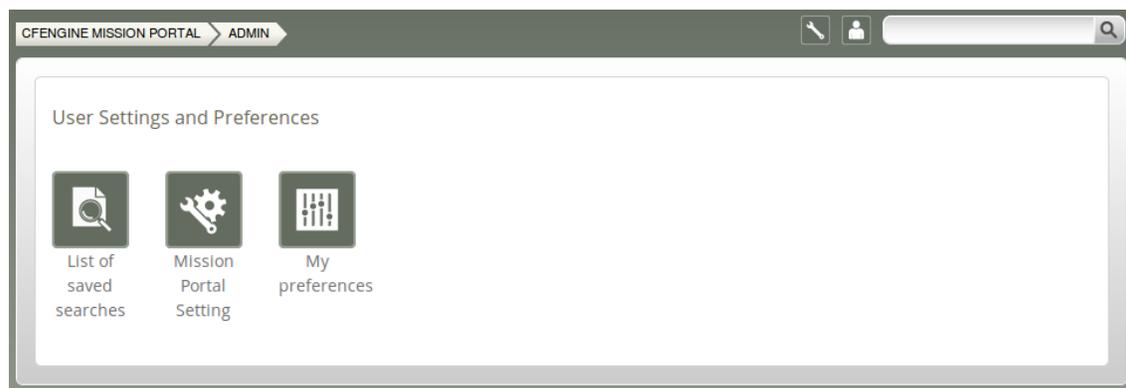


Figure: Mission Portal Settings

- List of saved searches: View saved searches (e.g. searches saved from reports viewer/finder)
- Mission Portal settings: Configure access authentication mode (internal database, LDAP, Active Directory)
- My preferences: Turn tooltips on/off and enter default number of rows to be shown in reports

2.5.2 Mission Portal User Admin

For the moment, user management in the Mission Portal only concerns access to the Mission Portal settings page. It does not affect the ability to edit or approve policies in the policy editor. Access control and authorization for policies are handled by subversion's authentication system, see section on "Integration with subversion".

On a fresh install the default user "admin" belongs to the "admin" user group and has access to the settings page. Add users and define user groups (Admin, Faculty, Developer, etc.). Alternatively, configure the Mission Portal for use of external authentication (LDAP, Active Directory), see Appendix A. Note that the actual setup of external authentication (definition of users, groups, directory hierarchy, etc.) has to be done independently and is not covered in this document.



The screenshot shows the 'Users' tab in the CFEngine Mission Portal Admin interface. The table lists five users with their email addresses, assigned groups, status, and action icons (edit, delete, and a third icon).

USERS	EMAIL	GROUPS	STATUS	ACTION
admin		admin	Active	 
testuser1	test.user1@cfengine.com	admin	Active	  
testuser2	test.user2@cfengine.com	admin, Superadmin	Active	  
testuser3	test.user3@cfengine.com	Developers	Active	  
testuser4	test.user4@cfengine.com	Managers	Active	  

Figure: Mission Portal User Admin

2.5.3 Troubleshooting - Mission Portal Logs

The Mission Portal will log php errors, errors related to the configuration of external authentication (LDAP/Active Directory) and occurrences of the fall-back solution if external authentication fails. The log can be found in `DOCROOT/application/logs` (i.e. `/var/www/application/logs` for Ubuntu and `/var/www/html/application/logs` for Red Hat). The logs folder must be writable by apache, i.e. by 'www-data' user.

Appendix A Configuration of external authentication

External authentication is available for CFEngine Nova 2.1 and up, but by default the Mission Portal will use the embedded database to store user information (default username and password are "admin" and "admin"). Note that users in the default database will be locked out of the Mission Portal upon configuration of external authentication. They will regain access if external authentication is deactivated by selecting the Database button on the Mission Portal Settings page (see below).

To enable external authentication on a fresh install, log on to the Mission Portal with the default user and password and go to "Settings". Click "Mission Portal Settings" and enter the appropriate configuration for LDAP or Active Directory as described below.

A.1 Configure LDAP

Select the LDAP button and enter the appropriate configuration settings for your system.

The screenshot shows the 'Mission Portal settings' page in a web browser. The breadcrumb trail is 'MISSION PORTAL > ENGINEERING > ADMIN > SETTINGS'. The page contains the following fields and options:

- Administrative email ***:
- Administrative group**:
- Authentication method ***:
 - Internal
 - LDAP
 - Active Directory
- LDAP host ***:
- Base dn ***:
- Login attribute ***:
- User directory ***:
- bind dn:** cn=users,dc=cf022osx,dc=cfengine,dc=com
- Member attribute (secondary)**:
- Encryption**:
 - None
 - SSL
 - STARTTLS
- Test settings**:
- Fall-back group (if authentication server down)**:
- Submit**:

Figure: Configure LDAP

Form fields:

- LDAP host: Address of the LDAP machine
- Base dn: LDAP root, the top entry (starting point) in the directory
- Login attribute: Field name used to match username, e.g. uid.
- User directory: Directory name where usernames are stored, e.g. cn=users or ou=people
- Member attribute: Field name used to match user group (only needed for OPENLDAP), e.g. memberUid

Always check that the entered configuration is correct by clicking the "Test it" button before submitting changes. Enter a valid username and password in the popup to test LDAP bind. Submitting an incorrect configuration will put LDAP down and lock out all users, CFE Nova therefore comes with a fallback solution. Select a fallback administrator group in the database from the dropbox, CFE Nova will look for users in this group when an incorrect configuration has been passed. A member of this group will then be able to log on to the Mission Portal using his database username and password and restart the external authentication configuration.

Also select which user group will have general administrator privileges in the Mission Portal. This group will be able to access the Mission Portal settings and configuration tools.

A.2 Configure Active Directory

Select the Active Directory button.

The screenshot shows the 'Mission Portal settings' page. The 'Authentication method' is set to 'Active Directory'. The 'LDAP host' is 'cf022osx.cfengine.com', 'Base dn' is 'dc=cf022osx,dc=cfengine,dc=com', 'Login attribute' is 'uid', 'User directory' is 'cn=users', and 'Active directory domain' is 'bind dn: cn=users,dc=cf022osx,dc=cfengine,dc=com'. The 'Encryption' is set to 'SSL'. The 'Fall-back group' is 'Superadmin'. A 'Test settings' button is located below the 'Encryption' options.

Figure: Configure Active Directory

Form fields:

- LDAP host: Address of the LDAP machine
- Base dn: LDAP root, the top entry (starting point) in the directory
- Login attribute: Field name used to match username, e.g. uid.
- User directory: Directory name where usernames are stored, e.g. cn=users or ou=people
- Active directory domain: Field name used to match directory domain on Windows machines, e.g. windows1.test.cfengine.com

Always check that the entered configuration is correct by clicking the "Test it" button before submitting changes. Enter a valid username and password in the popup to test LDAP bind. Submitting an incorrect configuration will put LDAP down and lock out all users, CFE Nova therefore comes with a fallback solution. Select a fallback administrator group in the database from the dropdown, CFE Nova will look for users in this group when an incorrect configuration has been passed. A member of this group will then be able to log on to the Mission Portal using his database username and password and restart the external authentication configuration.

Also select which user group will have general administrator privileges in the Mission Portal. This group will be able to access the Mission Portal settings and configuration tools.

A.3 Directory structure

The Mission Portal imposes some limitations on directory structure for the use of external authentication. All users have to be stored in a direct child directory of the root (base dn). You may use

subdirectories if you need to organize users into categories, but do not put more than one user directory directly under the root (base dn) as the user search will only cover the first encountered directory.

The same counts for group directories.

