Spruce user manual

Contents

Prerequisites

To be able to test several file systems in a normal level some tools are needed. The set of necessary tools can be divided into three groups:

- Mandatory globally
- Mandatory for a single file system
- Optional

Globally mandatory tools

- A C++ compiler.
- CMake system - needed for the configuration step
- GNU Make - to build the system
- xsltproc - needed by the engine for test generation

FS mandatory tools

All the packages listed below contain the corresponding mkfs.XXX tools. The names of the packages may vary in different GNU/Linux distributions.

- Ext4 : e2fsprogs.
- JFS : jfsutils.
- XFS: xfsprogs.
- BtrFS: btrfs-progs.

Optional tools

There are some extra functionalities which can be tested only having some extra tools in the system.

- quota - contains the quotacheck binary. Needed by the quotactl tests to enable quota on the test device.
- libattr-dev - contains the function prototypes for extended attribute manipulations.
- e2fslibs-dev - contains the Ext4 FS ioctl command definitions.
- xfsprogs-dev - contains the XFS ioctl command definitions.

In case of x86_64 platform, the Spruce system functionality can be extended even more. If the g++multilib library is installed then Spruce can also test the compatibility or the kernel.

If any of these packages is missing then warnings will show up during the configuration step. **Important. After the necessary package is installed, all the files in the project build folder must be removed and the configuration process must be restarted.**

Installation

After downloading the source archive unpack it. Lets assume the project is unpacked in the folder */home/me/workspace/spruce.x.y*. Next create a new folder (let's say */home/me/workspace/build)* and enter that folder. The Spruce system configuration step is done using the CMake tool set. It is included in the package repositories of all the main GNU/Linux distributions. But if you do not find it there, it still can be found on CMake official site. Follow the steps described in the documentation to install the configuration system.

Spruce uses out-of-source build technology that's why another folder is needed. In the build folder execute the following command:

cmake -DCMAKE_INSTALL_PREFIX=/path/to/use ../spruce.x.y

On this stage all the dependencies are satisfied and the system get's ready to be built. Read all the warning and error messages that would be found during the configuration phase. To be able to use the Spruce system completely all the warnings and error must be eliminated. So the configuration step must be repeated untill there are no warnings there any more.

To start the build process execute the following command:

make -j N

where *N* is the number of CPU cores you want to be activated to parallel the build process. The build time depends on many factors which can bring to several minutes long build. Parallel execution may reduce this time.

Now the system can be installed:

sudo make install

If all the steps were passed without errors then the system is installed under the folder CMAKE_INSTALL_PREFIX, defined on the configuration step.

Usage

The Spruce system consists of several modules. Also it supports configuration files. So to define what exactly you want Spruce to do create a configuration file (or reuse one provided by the system: *CMAKE_INSTALL_PREFIX/share/spruce/config/config_XXX*). Spruce supports the following configuration values

Configuration key	Description	Default value	Example
fs	The list of file systems to be verified	None	ext4;xfs;btrfs
modules	The list of modules to be activated	None	syscall;fs-spec
partition	The device to use for the testing	None	/dev/sda5
browser	The browser to be used to show the test results	None	firefox
logfolder	The folder to store the result log	/tmp	/home/me/spruce_logs
mount_at	The folder to mount the device at	/tmp/ spruce_test	/mnt/spruce_test
mount_opts	Options to pass the mount program (via -o option)	None	ro,quota
exclude_tests	Test names (including fs and module names, , mount options, testset name) to be excluded	None	ext4.ro.syscall.ioctl.IoctlGetSetVersion;
run_tests	Test names (including fs and module names, , mount options, testset name) to run	All the tests	ext4.ro.syscall.ioctl.IoctlGetSetVersion;

To run the system just execute the main binary file and pass the configuration file in the following way:

CMAKE_INSTALL_PREFIX/bin/spruce -c /path/to/config/file

Important: Spruce can be executed only as root.

The Spruce system operates in the following way:

- Creates the corresponding file system on the mentioned device
- Mounts the device to the mentioned folder
- Enters that folder
- Runs the mentioned modules and collects the output
- Saves the output to the log file (in XML format)
- Opens the results in the browser using XSLT to show as HTML

The Spruce system detects several kinds of errors like dysfunctionality, wrong (or missing) error paths, memory leaks.