cconfig 0.1

Nico Schottelius (nico-linux-cconfig ((at)) schottelius.org)

2005-08-28

Contents

1 Introduction

cconfig is a proposal for configurations on Unix-alike systems[1]. This includes, but is not limited to services, packages, user settings or the system configuration. cconfig main idea is

- to simplify the layout of configurations

- to make it easier to configure

- to allow smooth editing from automatically running programs,

- thus to avoid parsing where possible,

- to use the available resources (libraries, functions, ...)

- and to prevent doubled implementations.

2 Layout

Every cconfig starts below a given directory. This directory should be named after the process using the configuration (auth, cinit, mini-lpd, yourutils) and be put under the appropriate configuration directory (/etc, /usr/packages/yourutil/etc, ...). A cconfig consists of objects and attributes.

2.1 Objects

An object is a subdirectory below your configuration directory. An object can also be a subobject, means a subdirectory of a directory (this may be continued as deep as you need[2].

2.2 Attributes

Attributes describe a specific configuration option. They may exist directly below the configuration directory or below objects.

2.3 Links

With links you can reuse attributes or objects under a different name or at a different location[3].

---

[1]In fact it can be used on any system, which fits the requirements. See 3, 4 for more information

[2]Have a look at the portability issue, 3.5, page 5.

[3]Have a look at the portability issue, 3.4, page 4.

3   Portability

cconfig should in general be portable between Unix-systems, still there *can* be problems exchanging cconfig data. To prevent them, take care of the following sections.

3.1   Exchanging

Transferring a cconfig should be done encapsulated in a tar[5] archive. Tar saves the most important information (date, rights, names, ...) and is available on every Unix-system[4]. If the configuration is very big, consider compressing it with bzip2[6].

3.2   File names

The file- and directory names and the content of the files should either be

- plain ASCII

- or UTF-8[3]

Currently there is no specification, that specifies that a filesystem must support UTF-8, but practically most modern systems do have UTF-8 support[5].

3.3   File contents

The same rules as in "'File names"' apply.

3.4   Using (symbolic) links (absolute and relative)

Hard links are not a problem, because when transferring the data they look as different files, tar will not see the difference between a totally different file and a hard link.

When using symbolic links one has to distinguish between absolute and relative links. Using a symbolic link can be a problem in general, if you link outside of the configuration directory. If you link to other files or directories in the configuration directory you should prefer relative links. When using absolute links the linked files may be non-existent, when using relative links to files outside the files may be unreachable, because the cconfig is extracted below a different hierarchy position.

---

[4]As of the current date tar seems **not** to be able to save ACLs (access control lists).

[5]Reiserfs, ext2, ext3, xfs should work fine, jfs is reported to had problems with some characters.

Some operating system implement links differently, you need to have a tool, which converts standard links (as used on POSIX systems) to those used on other system and the other way round[6].

3.5   Quantity of objects

Some filesystems limit the count of directories (how deep directories may be nested). For normal configurations you'll never reach this limit[7].

4   Examples

So far said, let us see some examples after the theoretic part.

4.1   printer daemon (mini-lpd)

mini-lpd is a small, non-queueing lpd implementation. It uses `/etc/mini-lpd` as its configuration directory. Every directory below is the name of a queue:

```
[15:09] hydrogenium:~# ls -l /etc/mini-lpd
total 0
drwx------  2 root root 6 Aug 28 15:09 lp0
drwx------  2 root root 6 Aug 28 15:09 lp1
```

Those "'objects"' (directories) contain an attribute "'device"', which is where the mini-lpd writes the incoming data to:

```
[15:10] hydrogenium:~# ls -l /etc/mini-lpd/lp*
/etc/mini-lpd/lp0:
total 0
lrwxrwxrwx  1 root root 9 Aug 28 15:10 device -> /dev/null

/etc/mini-lpd/lp1:
total 0
lrwxrwxrwx  1 root root 8 Aug 28 15:10 device -> /dev/lp0
```

You could add other attributes like "'filter"' (unsupported currently), which could be a link to a program filtering the input before mini-lpd would write it to the device.

---

[6]Those systems are most likely **not** Unix-alike and have a general different design with complex and broken registries.

[7]Please report if you ever reach this limit on a filesystem, I would be interested in how deep the limit is. Currently one person reported that after 125 subdirectories the limit is reached in the operating system with the blue screen.

4.2   Samba

Samba[7] is the well-known Unix SMB server. It exists and does **not** use
cconfig for its configuration, but this describes how it could: "'/etc/samba"'
is currently the configuration directory. Instead of the smb.conf you would
have the object "'config"'. Bool variables[8] could simply be represented by
"'file exists"' (true) or "'file does not exist"'. Variables with text could be
realised with writing the text into the file: "'/etc/samba/config/workgroup"'
could contain "'cworkgroup"'. Shares could be objects below "'/etc/samba/shares"':
"'/etc/samba/shares/cdrom"' could contain a file name "'path"' which links
to the real path, a file "'groups"' which would contain a \n seperated list of
groups which may access this share.

4.3   init system (cinit)

Cinit[2] is a fast init system, which uses cconfig. "'/etc/cinit"' is its basic
configuration directory. Below that directory exist the basic configuration:

```
wasserstoff# ls -l /etc/cinit
total 8
drwxr-xr-x  10 root root 4096 Aug 20 20:55 getty
drwxr-xr-x   3 root root   56 Aug 24 22:04 init
drwxr-xr-x   2 root root    1 Aug 20 20:38 local-services
drwxr-xr-x   5 root root   32 Aug 24 22:14 local-tuning
drwxr-xr-x  10 root root   72 Aug 24 06:40 mount
drwxr-xr-x   7 root root   48 Aug 28 13:57 network
drwxr-xr-x   4 root root   16 Aug 20 20:51 remote-services
drwxr-xr-t   2 root root   60 Aug 28 14:00 tmp
```

"'/etc/cinit/init"' is the first service cinit normally boots, from there it has
dependencies to other services:

```
wasserstoff# ls -l /etc/cinit/init/*
lrwxrwxrwx  1 root root  9 Aug 20 20:43 /etc/cinit/init/on -> /bin/echo
-rw-r--r--  1 root root 22 Aug 20 20:43 /etc/cinit/init/on.params

/etc/cinit/init/wants:
total 0
lrwxrwxrwx  1 root root 12 Aug 20 20:43 getty -> ../../getty/
lrwxrwxrwx  1 root root 21 Aug 20 20:51 local-services -> ../../local-services/
lrwxrwxrwx  1 root root 19 Aug 20 20:42 local-tuning -> ../../local-tuning/
lrwxrwxrwx  1 root root 12 Aug 20 20:51 mount -> ../../mount/
lrwxrwxrwx  1 root root 14 Aug 20 20:51 network -> ../../network/
lrwxrwxrwx  1 root root 22 Aug 20 20:51 remote-services -> ../../remote-services/
```

---

[8]Those which can either be TRUE or FALSE.

The "'wants"' describe soft-dependencies, "'on"' specifies what to start and "'on.params"' are the parameters to pass.

5   Problems

cconfig is not a standard currently. No tools exists to convert to other formats like XML. There is no description language (like DTD or XSLT for XML). The portability is ensured between most Unices, but Win32 will most likely be broken as long as nobody writes a "'cconfigtar2win32fs"' convert tool.

6   Personal statement

I wrote this because I think configuring things this way is very logic and helps to intuitively use your system. Furthermore it is really easy to see what is configured (`find /the/cconfig/dir`). I implemented variants of cconfig in cinit[2] and mini-lpd[4]. Many people I know, who use xml[8] and have problems, because the parser is too slow, were very happy when I told them about cconfig. I hope you can also benefit from it. I would really like to hear some feedback from you at nico-linux-cconfig ((at)) schottelius.org.

References

[1] Posix and SUS;
    http://www.opengroup.org/platform/single_unix_specification/

[2] cinit;
    http://linux.schottelius.org/cinit/

[3] UTF-8;
    http://www.utf-8.com/;
    http://www.cl.cam.ac.uk/~mgk25/unicode.html

[4] mini-lpd;
    http://linux.schottelius.org/mini-lpd/

[5] tar;
    http%3A//www.gnu.org/software/tar/tar.html

[6] bzip2;
    http://www.bzip.org/

[7] Samba;
    http://www.samba.org/

[8] XML;
    http://www.w3.org/XML/