

nis-util

Reference Manual

Peter Miller

pmiller@opensource.org.au

This document describes nis-util version 1.0
and was prepared 30 November 2012.

This document describing the nis-util program, and the nis-util program itself, are
Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

This program is free software; you can redistribute it and/or modify it under the terms of the
GNU General Public License as published by the Free Software Foundation; either version 2 of
the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY;
without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If
not, see <<http://www.gnu.org/licenses/>>.

	The README file	1
	Release Notes	2
	The BUILDING file	3
nis-util-afpvol-gen(1)	generate AFP volume maps from automount maps	7
nis-util-auto-master-check(1)	check the contents of <i>auto.master(5)</i> files	9
nis-util-ethers(1)	check and manipulate <i>ethers(5)</i> files	11
nis-util-group(1)	check and manipulate <i>group(5)</i> files	13
nis-util-group-members(1)	list the members of Unix groups	15
nis-util-hosts(1)	check and manipulate <i>hosts(5)</i> files	17
nis-util-license(1)	GNU General Public License	19
nis-util-netgroup(1)	check and manipulate netgroup files	24
nis-util-netid(1)	check <i>inetd.conf(5)</i> files	26
nis-util-netmasks(5)	check and manipulate <i>netmasks(5)</i> files	27
nis-util-networks(1)	check and manipulate networks files	29
nis-util-passwd(1)	check and manipulate passwd files	31
nis-util-protocols(1)	check and manipulate protocols files	33
nis-util-services(1)	check and manipulate services files	35
nis-util-conf(5)	nis-util configuration file	37

Table of Contents(nis-util)

	3
nis-util-afpvol-gen(1)	7
nis-util-afpvol-gen(1)	7
nis-util-auto-master-check(1)	9
nis-util-auto-master-check(1)	9
nis-util-afpvol-gen(1)	7
nis-util-ethers(1)	11
nis-util-group(1)	13
nis-util-hosts(1)	17
nis-util-netmasks(1)	
nis-util-networks(1)	29
nis-util-passwd(1)	31
nis-util-protocols(1)	33
nis-util-services(1)	35
nis-util-netgroup(1)	24
nis-util-auto-master-check(1)	9
nis-util-netid(1)	26
	3
	3
	3
	3
	3
nis-util-auto-master-check(1)	9
nis-util-netid(1)	26
nis-util-conf(5)	37
nis-util-conf(5)	37
nis-util-auto-master-check(1)	9
nis-util-conf(5)	37
nis-util-ethers(1)	11
nis-util-ethers(1)	11
nis-util-conf(5)	37
nis-util-auto-master-check(1)	9
nis-util-ethers(1)	11
nis-util-group(1)	13
nis-util-hosts(1)	17
nis-util-netgroup(1)	24
nis-util-netid(1)	26
nis-util-netmasks(1)	
nis-util-networks(1)	29

Table of Contents(nis-util)

nis-util-afpvol-gen - generate AFP volume maps from automount maps	
nis-util-afpvol-gen - generate AFP volume maps from automount maps	
nis-util-auto-master-check - check the contents of I]auto.master(5) files	
nis-util-auto-master-check - check the contents of I]auto.master(5) files	
nis-util-afpvol-gen - generate AFP volume maps from automount maps	
nis-util-ethers - check and manipulate I]ethers(5) files	
nis-util-group - check and manipulate I]group(5) files	
nis-util-hosts - check and manipulate I]hosts(5) files	
nis-util-netmasks - check and manipulate I]netmasks(5) files	
nis-util-networks - check and manipulate I]networks(5) files	
nis-util-passwd - check and manipulate I]passwd(5) files	
nis-util-protocols - check and manipulate I]protocols(5) files	
nis-util-services - check and manipulate I]services(5) files	
nis-util-netgroup - check and manipulate netgroup files	
nis-util-auto-master-check - check the contents of I]auto.master(5) files	
nis-util-netid - check I]inetd.conf(5) files	
check the contents of I]auto.master(5) files	
check the contents of I]auto.master(5) files	
check the contents of I]auto.master(5) files	
check the contents of I]auto.master(5) files	
nis-util-auto-master-check - check the contents of I]auto.master(5) files	
nis-util-netid - check I]inetd.conf(5) files	
/etc/nis-util.conf - nios-util configuration file	
/etc/nis-util.conf - nios-util configuration file	
nis-util-auto-master-check - check the contents of I]auto.master(5) files	
/etc/nis-util.conf - nios-util configuration file	
nis-util-ethers - check and manipulate I]ethers(5) files	
nis-util-ethers - check and manipulate I]ethers(5) files	
/etc/nis-util.conf - nios-util configuration file	
nis-util-auto-master-check - check the contents of I]auto.master(5) files	
nis-util-ethers - check and manipulate I]ethers(5) files	
nis-util-group - check and manipulate I]group(5) files	
nis-util-hosts - check and manipulate I]hosts(5) files	
nis-util-netgroup - check and manipulate netgroup files	
nis-util-netid - check I]inetd.conf(5) files	
nis-util-netmasks - check and manipulate I]netmasks(5) files	
nis-util-networks - check and manipulate I]networks(5) files	

nis-util-passwd(1)	31	nis-util-passwd - check and manipulate I]passwd(5)	files
nis-util-protocols(1)	33	nis-util-protocols - check and manipulate I]protocols(5)	files
nis-util-services(1)	35	nis-util-services - check and manipulate I]services(5)	files
nis-util-afpvol-gen(1)	7	nis-util-afpvol-gen - generate AFP volume maps	from automount maps
nis-util-afpvol-gen(1)	7	nis-util-afpvol-gen -	generate AFP volume maps from automou maps
nis-util-afpvol-gen(1)	7	nis-util-afpvol-	gen - generate AFP volume maps from automount maps
nis-util-group(1)	13	nis-util-group - check and manipulate I]	group(5) files
nis-util-group(1)	13	nis-util-	group - check and manipulate I]group(5) files
nis-util-group-members(1)	15	nis-util-	group-members - list the members of Unix groups
nis-util-group-members(1)	15	nis-util-group-members - list the members of Unix	groups
nis-util-hosts(1)	17	nis-util-hosts - check and manipulate I]	hosts(5) files
nis-util-hosts(1)	17	nis-util-	hosts - check and manipulate I]hosts(5) file
nis-util-auto-master-check(1)	9	nis-util-auto-master-check - check the contents of	I]auto.master(5) files
nis-util-ethers(1)	11	nis-util-ethers - check and manipulate	I]ethers(5) files
nis-util-group(1)	13	nis-util-group - check and manipulate	I]group(5) files
nis-util-hosts(1)	17	nis-util-hosts - check and manipulate	I]hosts(5) files
nis-util-netid(1)	26	nis-util-netid - check	I]inetd.conf(5) files
nis-util-netid(1)	26	nis-util-netid - check I]	inetd.conf(5) files
nis-util-netmasks(1)		nis-util-netmasks - check and manipulate	I]netmasks(5) files
nis-util-networks(1)	29	nis-util-networks - check and manipulate	I]networks(5) files
nis-util-passwd(1)	31	nis-util-passwd - check and manipulate	I]passwd(5) files
nis-util-protocols(1)	33	nis-util-protocols - check and manipulate	I]protocols(5) files
nis-util-services(1)	35	nis-util-services - check and manipulate	I]services(5) files
nis-util-group-members(1)	15	nis-util-group-members -	list the members of Unix groups
nis-util-ethers(1)	11	nis-util-ethers - check and	manipulate I]ethers(5) files
nis-util-group(1)	13	nis-util-group - check and	manipulate I]group(5) files
nis-util-hosts(1)	17	nis-util-hosts - check and	manipulate I]hosts(5) files
nis-util-netmasks(1)		nis-util-netmasks - check and	manipulate I]netmasks(5) files
nis-util-networks(1)	29	nis-util-networks - check and	manipulate I]networks(5) files
nis-util-passwd(1)	31	nis-util-passwd - check and	manipulate I]passwd(5) files
nis-util-protocols(1)	33	nis-util-protocols - check and	manipulate I]protocols(5) files
nis-util-services(1)	35	nis-util-services - check and	manipulate I]services(5) files
nis-util-netgroup(1)	24	nis-util-netgroup - check and	manipulate netgroup files
nis-util-afpvol-gen(1)	7	nis-util-afpvol-gen - generate AFP volume maps from automount	maps
nis-util-afpvol-gen(1)	7	nis-util-afpvol-gen - generate AFP volume	maps from automount maps
nis-util-auto-master-check(1)	9	nis-util-auto-master-check - check the contents of I]auto.	master(5) files
nis-util-auto-master-check(1)	9	nis-util-auto-	master-check - check the contents of I]auto.master(5) files
nis-util-group-members(1)	15	nis-util-group-	members - list the members of Unix group
nis-util-group-members(1)	15	nis-util-group-members - list the	members of Unix groups

Table of Contents(nis-util)

Table of Contents(nis-util)

nis-util-netgroup(1) 24

nis-util-netgroup(1) 24

nis-util-netid(1) 26

nis-util-netmasks(1)

nis-util-netmasks(1)

nis-util-networks(1) 29

nis-util-networks(1) 29

nis-util-conf(5) 37

nis-util-afpvol-gen(1) 7

nis-util-auto-master-check(1) 9

nis-util-conf(5) 37

nis-util-ethers(1) 11

nis-util-group(1) 13

nis-util-group-members(1) 15

nis-util-hosts(1) 17

nis-util-netgroup(1) 24

nis-util-netid(1) 26

nis-util-netmasks(1)

nis-util-networks(1) 29

nis-util-passwd(1) 31

nis-util-protocols(1) 33

nis-util-services(1) 35

nis-util-passwd(1) 31

nis-util-passwd(1) 31

nis-util-protocols(1) 33

nis-util-protocols(1) 33

nis-util-services(1) 35

nis-util-services(1) 35

nis-util-group-members(1) 15

nis-util-afpvol-gen(1) 7

nis-util-auto-master-check(1) 9

nis-util-conf(5) 37

nis-util-netgroup - check and manipulate netgroup files

nis-util-netid - check I]inetd.conf(5) files

nis-util-netmasks - check and manipulate I]netmasks(5) files

nis-util-networks - check and manipulate I]networks(5) files

/etc/nis-util.conf - nios-util configuration file

nis-util-afpvol-gen - generate AFP volume maps from automount maps

nis-util-auto-master-check - check the contents of I]auto.master(5) files

/etc/nis-util.conf - nios-util configuration file

nis-util-ethers - check and manipulate I]ethers(5) files

nis-util-group - check and manipulate I]group(5) files

nis-util-group-members - list the members of Unix groups

nis-util-hosts - check and manipulate I]hosts(5) files

nis-util-netgroup - check and manipulate netgroup files

nis-util-netid - check I]inetd.conf(5) files

nis-util-netmasks - check and manipulate I]netmasks(5) files

nis-util-networks - check and manipulate I]networks(5) files

nis-util-passwd - check and manipulate I]passwd(5) files

nis-util-protocols - check and manipulate I]protocols(5) files

nis-util-services - check and manipulate I]services(5) files

nis-util-passwd - check and manipulate I]passwd(5) files

nis-util-protocols - check and manipulate I]protocols(5) files

nis-util-services - check and manipulate I]services(5) files

nis-util-group-members - list the members of Unix groups

nis-util-afpvol-gen - generate AFP volume maps from automount maps

nis-util-auto-master-check - check the contents of I]auto.master(5) files

/etc/nis-util.conf - nios-util configuration file

netgroup - check and manipulate netgroup files

netid - check I]inetd.conf(5) files

netmasks(5) files

netmasks - check and manipulate I]netmasks(5) files

networks(5) files

networks - check and manipulate I]networks(5) files

nios-util configuration file

nis-util-afpvol-gen - generate AFP volume maps from automount maps

nis-util-auto-master-check - check the contents of I]auto.master(5) files

nis-util.conf - nios-util configuration file

nis-util-ethers - check and manipulate I]ethers(5) files

nis-util-group - check and manipulate I]group(5) files

nis-util-group-members - list the members of Unix groups

nis-util-hosts - check and manipulate I]hosts(5) files

nis-util-netgroup - check and manipulate netgroup files

nis-util-netid - check I]inetd.conf(5) files

nis-util-netmasks - check and manipulate I]netmasks(5) files

nis-util-networks - check and manipulate I]networks(5) files

nis-util-passwd - check and manipulate I]passwd(5) files

nis-util-protocols - check and manipulate I]protocols(5) files

nis-util-services - check and manipulate I]services(5) files

passwd(5) files

passwd - check and manipulate I]passwd(5) files

protocols(5) files

protocols - check and manipulate I]protocols(5) files

services(5) files

services - check and manipulate I]services(5) files

Unix groups

util-afpvol-gen - generate AFP volume maps from automount maps

util-auto-master-check - check the contents of I]auto.master(5) files

util configuration file

Table of Contents(nis-util)

nis-util-conf(5)	37
nis-util-ethers(1)	11
nis-util-group(1)	13
nis-util-group-members(1)	15
nis-util-hosts(1)	17
nis-util-netgroup(1)	24
nis-util-netid(1)	26
nis-util-netmasks(1)	
nis-util-networks(1)	29
nis-util-passwd(1)	31
nis-util-protocols(1)	33
nis-util-services(1)	35
nis-util-afpvol-gen(1)	7

Table of Contents(nis-util)

/etc/nis-	util.conf - nios-util configuration file
nis-	util-ethers - check and manipulate I]ethers(5) files
nis-	util-group - check and manipulate I]group(files
nis-	util-group-members - list the members of Unix groups
nis-	util-hosts - check and manipulate I]hosts(5) files
nis-	util-netgroup - check and manipulate netgroup files
nis-	util-netid - check I]inetd.conf(5) files
nis-	util-netmasks - check and manipulate I]netmasks(5) files
nis-	util-networks - check and manipulate I]networks(5) files
nis-	util-passwd - check and manipulate I]passwd(5) files
nis-	util-protocols - check and manipulate I]protocols(5) files
nis-	util-services - check and manipulate I]services(5) files
nis-util-afpvol-gen	- generate AFP volume maps from automount maps

NAME

nis-util – NIS administration utilities

DESCRIPTION

The *nis-util* package is a collection of powerful tools for manipulating NIS map files.

I wrote nis-util because NIS maps are highly interrelated. In order to provide a stable working environment for users, it is essential the the system administrator have tools which report inconsistencies between NIS maps in a timely manner.

The nis-utils package contains a number of tools which can be used to generate input to *makedbm(8)*, in place of the *awk(1)* scripts traditionally used. These tools are far more exacting, and perform a number of input validations, in addition to the necessary reformatting.

In addition, each of these tools had a *checking* mode, which allows them to be validated against other files, *e.g.* validating *passwd(5)* files against *group(5)* files to make sure that the default groups actually exist.

ARCHIVE SITE

The latest version of *nis-util* is available on the Web from:

URL:	http://miller.emu.id.au/pmiller/software/nis-util/	
File:	index.html	# the nis-util page
File:	nis-util-1.0.README	# Description, from the tar file
File:	nis-util-1.0.lsm	# Description, LSM format
File:	nis-util-1.0.spec	# RedHat package specification
File:	nis-util-1.0.tar.gz	# the complete source
File:	nis-util-1.0.pdf	# Reference Manual

BUILDING NIS-UTIL

Full instructions for building *nis-util* may be found in the *BUILDING* file included in this distribution.

COPYRIGHT

version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *program comes with ABSOLUTELY NO WARRANTY*; for details use the '*-VERsion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*-VERsion License*' command.

AUTHOR

Peter Miller	E-Mail:	pmiller@opensource.org.au
^^*	WWW:	http://miller.emu.id.au/pmiller/

RELEASE NOTES

This section details the various features and bug fixes of the various releases. For excruciating and complete detail, and also credits for those of you who have generously sent me suggestions and bug reports, see the *etc/CHANGES.** files.

Version 1.0

No public release.

NAME

build nis-util – how to build the NIS administration utilities

SPACE REQUIREMENTS

You will need about 1MB to unpack and build the *nis-util* package. Your mileage may vary.

BEFORE YOU START

There are a few pieces of software you may want to fetch and install before you proceed with your installation of *nis-util*.

GNU Groff

The documentation for the *nis-util* package was prepared using the GNU Groff package (version 1.14 or later). This distribution includes full documentation, which may be processed into PostScript or DVI files at install time – if GNU Groff has been installed.

GCC You may also want to consider fetching and installing the GNU C Compiler if you have not done so already. This is not essential. *nis-util* was developed using the GNU C++ compiler, and the GNU C++ libraries.

The GNU FTP archives may be found at `ftp.gnu.org`, and are mirrored around the world.

SITE CONFIGURATION

The **nis-util** package is configured using the *configure* program included in this distribution.

The *configure* shell script attempts to guess correct values for various system-dependent variables used during compilation, and creates the *Makefile* and *include/config.h* files. It also creates a shell script *config.status* that you can run in the future to recreate the current configuration.

Normally, you just *cd* to the directory containing *nis-util*'s source code and then type

```
% ./configure
...lots of output...
%
```

If you're using *csh* on an old version of System V, you might need to type

```
% sh configure
...lots of output...
%
```

instead to prevent *csh* from trying to execute *configure* itself.

Running *configure* takes a minute or two. While it is running, it prints some messages that tell what it is doing. If you don't want to see the messages, run *configure* using the quiet option; for example,

```
% ./configure --quiet
%
```

To compile the **nis-util** package in a different directory from the one containing the source code, you must use a version of *make* that supports the *VPATH* variable, such as *GNU make*. *cd* to the directory where you want the object files and executables to go and run the *configure* script. *configure* automatically checks for the source code in the directory that *configure* is in and in *..* (the parent directory). If for some reason *configure* is not in the source code directory that you are configuring, then it will report that it can't find the source code. In that case, run *configure* with the option `--srcdir=directory`, where *DIR* is the directory that contains the source code.

By default, *configure* will arrange for the *make install* command to install the **nis-util** package's files in */usr/local/bin*, and */usr/local/man*. There are options which allow you to control the placement of these files.

`--prefix=path`

This specifies the path prefix to be used in the installation. Defaults to */usr/local* unless otherwise specified.

`--exec-prefix=path`

You can specify separate installation prefixes for architecture-specific files files. Defaults to *{prefix}* unless otherwise specified.

`--bindir=path`

This directory contains executable programs. On a network, this directory may be shared between machines with identical hardware and operating systems; it may be mounted read-only. Defaults to `/${exec_prefix}/bin` unless otherwise specified.

`--mandir=path`

This directory contains the on-line manual entries. On a network, this directory may be shared between all machines; it may be mounted read-only. Defaults to `/${prefix}/man` unless otherwise specified.

`configure` ignores most other arguments that you give it; use the `--help` option for a complete list.

On systems that require unusual options for compilation or linking that the `nis-util` package's `configure` script does not know about, you can give `configure` initial values for variables by setting them in the environment. In Bourne-compatible shells, you can do that on the command line like this:

```
$ CXX='g++ -traditional' LIBS=-lposix ./configure
...lots of output...
$
```

Here are the `make` variables that you might want to override with environment variables when running `configure`.

Variable: CXX

C++ compiler program. The default is `c++`.

Variable: CPPFLAGS

Preprocessor flags, commonly defines and include search paths. Defaults to empty. It is common to use `CPPFLAGS=-I/usr/local/include` to access other installed packages.

Variable: INSTALL

Program to use to install files. The default is `install` if you have it, `cp` otherwise.

Variable: LIBS

Libraries to link with, in the form `-lname ...`. The `configure` script will append to this, rather than replace it. It is common to use `LIBS=-L/usr/local/lib` to access other installed packages.

If you need to do unusual things to compile the package, the author encourages you to figure out how `configure` could check whether to do them, and mail diffs or instructions to the author so that they can be included in the next release.

BUILDING NIS-UTIL

All you should need to do is use the

```
% make
...lots of output...
%
```

command and wait. When this finishes you should see a directory called `bin` containing three files: `nis-util`, `nis-util` and `nis-util`.

nis-util `nis-util` program is used to manipulate and convert EPROM load files. For more information, see `nis-util(1)`.

nis-util The `nis-util` program is used to compare EPROM load files. For more information, see `nis-util(1)`.

nis-util The `nis-util` program is used to print information about EPROM load files. For more information, see `nis-util(1)`.

If you have GNU Groff installed, the build will also create a *etc/reference.ps* file. This contains the README file, this BUILDING file, and all of the man pages.

You can remove the program binaries and object files from the source directory by using the

```
% make clean
...lots of output...
%
```

command. To remove all of the above files, and also remove the *Makefile* and *include/config.h* and *config.status* files, use the

```
% make distclean
...lots of output...
%
```

command.

The file *etc/configure.ac* is used to create *configure* by a GNU program called *autoconf*. You only need to know this if you want to regenerate *configure* using a newer version of *autoconf*.

TESTING NIS-UTIL

The *nis-util* package comes with a test suite. To run this test suite, use the command

```
% make sure
...lots of output...
Passed All Tests
%
```

The tests take a few seconds each, but it varies greatly depending on your CPU.

If all went well, the message

```
Passed All Tests
```

should appear at the end of the make.

INSTALLING NIS-UTIL

As explained in the *SITE CONFIGURATION* section, above, the *nis-util* package is installed under the */usr/local* tree by default. Use the *--prefix=path* option to *configure* if you want some other path. More specific installation locations are assignable, use the *--help* option to *configure* for details.

All that is required to install the *nis-util* package is to use the

```
% make install
...lots of output...
%
```

command. Control of the directories used may be found in the first few lines of the *Makefile* file and the other files written by the *configure* script; it is best to reconfigure using the *configure* script, rather than attempting to do this by hand.

GETTING HELP

If you need assistance with the *nis-util* package, please do not hesitate to contact the author at

```
Peter Miller <pmiller@opensource.org.au>
```

Any and all feedback is welcome.

When reporting problems, please include the version number given by the

```
% nis-util -version
nis-util version 1.0.D110
...warranty disclaimer...
%
```

command. Please do not send this example; run the program for the exact version number.

COPYRIGHT

nis-util version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util* package is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

It should be in the *LICENSE* file included with this distribution.

COPYRIGHT

version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *program* comes with *ABSOLUTELY NO WARRANTY*; for details use the '*-VERsion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*-VERsion License*' command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
^/* WWW: http://miller.emu.id.au/pmiller/

NAME

nis-util-afpvol-gen – generate AFP volume maps from automount maps

SYNOPSIS

nis-util-afpvol-gen [*option...*]

nis-util-afpvol-gen -Help

nis-util-afpvol-gen -VERSion

DESCRIPTION

The *fP(n)* program is used to generate AFP volume maps from automount maps. You must use exactly one of the **-Direct** or **-Indirect** options.

OPTIONS

The following options are understood:

-Direct *filename*

This options is used to read the named file, as a direct map.

-Help Provide some help with using the *fP(1)* program.

-HOsts *filename*

This option may be used to specify a hosts map to validate the host names against. Must be given early on the command line.

-Indirect *filename*

This options is used to read the named file, as an indirect map.

-Output *filename*

By default, the map is written to the standard output. This options is used to select the output file to be written.

-VERSion

Print the version of the *fP(1)* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (_) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments "**-help**", "**-HEL**" and "**-h**" are all interpreted to mean the **-Help** option. The argument "**-hlp**" will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-afpvol-gen* are long, this means ignoring the extra leading '-'. The "**--option=value**" convention is also understood.

EXIT STATUS

The *nis-util-afpvol-gen* command will exit with a status of 1 on any error. The *nis-util-afpvol-gen* command will only exit with a status of 0 if there are no errors.

COPYRIGHT

nis-util-afpvol-gen version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-afpvol-gen* program comes with ABSOLUTELY NO WARRANTY; for details use the '*nis-util-afpvol-gen -VERSion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*nis-util-afpvol-gen -VERSion License*' command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
^^* WWW: http://miller.emu.id.au/pmiller/

NAME

`nis-util-auto-master-check` – check the contents of *auto.master*(5) files

SYNOPSIS

`nis-util-auto-master-check` [*option...*] *filename*

`nis-util-auto-master-check` **-Help**

`nis-util-auto-master-check` **-VERSion**

DESCRIPTION

The *nis-util-auto-master-check*(1) command is used to read an `/etc/auto.master` file, and the files it references, and cross checks the contents for consistency and common errors.

OPTIONS

The following options are understood:

-DEpends

Usually the files are checked. This option causes the names of the associated map files to be printed, for use generating *make*(1) dependency files.

-Idirectory

This option may be used to add a directory to the list of directories check within when attempting to locate an associated map file. This option may be used more than once. If no directories are specified, the list defaults to “.” (the current directory).

-Help Provide some help with using the *nis-util-auto-master-check* program.

-suffix=string

This option may be used to add a filename suffix to search for, when attempting to locate the associated map files. This option may be used more than once.

-VERSion

Print the version of the *nis-util-auto-master-check* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (`_`) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments “`-help`”, “`-HEL`” and “`-h`” are all interpreted to mean the **-Help** option. The argument “`-hlp`” will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-auto-master-check* are long, this means ignoring the extra leading ‘`-`’. The “`--option=value`” convention is also understood.

EXIT STATUS

The *nis-util-auto-master-check* command will exit with a status of 1 on any error. The *nis-util-auto-master-check* command will only exit with a status of 0 if there are no errors.

COPYRIGHT

nis-util-auto-master-check version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-auto-master-check* program comes with ABSOLUTELY NO WARRANTY; for details use the ‘*nis-util-auto-master-check -VERSion License*’ command. This is free software and you are welcome to redistribute it under certain conditions; for details use the ‘*nis-util-auto-master-check -VERSion License*’ command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
^^* WWW: http://miller.emu.id.au/pmiller/

NAME

nis-util-ethers – check and manipulate *ethers(5)* files

SYNOPSIS

```

nis-util-ethers -Ethers filename -Hosts filename
nis-util-ethers -ByName filename [ -Output filename ]
nis-util-ethers -ByAddress filename [ -Output filename ]
nis-util-ethers -Help
nis-util-ethers -VERSion

```

DESCRIPTION

The *nis-util-ethers* program is used to check and manipulate *ethers(5)* files.

Checking The File

The first synopsis line shows how to validate the contents of an *ethers(5)* file. Each field is checked to make sure it contains valid contents.

MAC Address

The MAC Address column is required to be valid, and in the canonical form. The exact form is necessary because NIS clients query by string, not by the more logical 6-byte binary representation. No duplicates are permitted. The file is required to be sorted by MAC address, this catches many of the typographical errors when entering the numbers.

Host Name

The host name column is required to be a valid host name, according to RFC xxxx. The host name is required to appear in the hosts file. The use of an IP address instead is discouraged, because many NIS clients expect a host name and can't cope with getting an IP address instead.

The overall line length (not including including the newline) must be less than 512 characters. This is a NIS protocol limitation, but it isn't usually a problem with this file.

NIS Databases

The NIS databases served by the NIS server, *ypserv(8)*, have their tables constructed by the *makedbm(8)* program. It needs to have its input as the index key, then a tab, and then the data to be indexed. The **-ByName** and **-ByAddress** options are used to read an *ethers(5)* file and output this format; some input checking is performed.

OPTIONS

The following options are understood:

-Ethers *filename*

This option is used to specify the *ethers(5)* file to be checked.

-Hosts *filename*

This option is used to specify the hosts file, used to verify the correctness of the host name column of the *ethers(5)* file.

-ByName *filename*

This option is used to project the given *ethers(5)* file into a suitable *makedbm(8)* input, indexed by host name. (Where an IP address appears instead of a host name, the lines will not be emitted.)

-ByAddress *filename*

This option is used to project the given *ethers(5)* file into a suitable *makedbm(8)* input, indexed by MAC address.

-Output *filename*

This option may be used with **-ByName** and **-ByAddress** to specify where to send the output. The default is to send it to the standard output.

-Help

Provide some help with using the *nis-util-ethers* program.

-VERsion

Print the version of the *nis-util-ethers* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (_) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments "-help", "-HEL" and "-h" are all interpreted to mean the **-Help** option. The argument "-hlp" will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-ethers* are long, this means ignoring the extra leading '-'. The "--option=value" convention is also understood.

EXIT STATUS

The *nis-util-ethers* command will exit with a status of 1 on any error. The *nis-util-ethers* command will only exit with a status of 0 if there are no errors.

SEE ALSO

ethers(5), *ypserv(8)*, *makedbm(1)*

COPYRIGHT

nis-util-ethers version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-ethers* program comes with ABSOLUTELY NO WARRANTY; for details use the '*nis-util-ethers -VERsion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*nis-util-ethers -VERsion License*' command.

AUTHOR

Peter Miller	E-Mail:	pmiller@opensource.org.au
^^*	WWW:	http://miller.emu.id.au/pmiller/

NAME

nis-util-group – check and manipulate *group(5)* files

SYNOPSIS

nis-util-group **-Passwd** *filename* **-Group** *filename*
nis-util-group **-ByName** *filename* [**-Output** *filename*]
nis-util-group **-ByGid** *filename* [**-Output** *filename*]
nis-util-group **-Help**
nis-util-group **-VERSion**

DESCRIPTION

The *nis-util-group* program is used to check and manipulate *group(5)* files.

Checking The File

The first synopsis line show how to validate the contents of the *group(5)* file. Each field is checked to make sure it contains valid contents. Group names and GIDs are checked for duplicates.

- name** The group name is required to start with a lower-case letter followed by lower-case letters, digits, underscores (_) or hyphens (-). Group names are limited to at most 8 characters.
- gid** The group ID is required to be a non-negative decimal number, in its simplest form. Group IDs are required to be in ascending order, because this catches many typographical errors in the numbers.
- members** List members are required to be existing users from the passwd file. List members are required to be unique, and in alphabetical order (the ordering makes duplicates easier to detect). List members may not name users who have the group as their default group; this helps avoid the line length limit.

The overall line length (not including including the newline) must be less than 512 characters. This is a NIS protocol limitation. Large and complex sites can run into this limit for groups with many members.

There is a maximum number of groups that a single user may belong to. (See *sysconf(3)*, the `_SC_NGROUPS_MAX` item.) However, it isn't the maximum number of groups on the server that matters, but the maximum number of groups on the most limited *client*. For this reason, users are limited to 16 groups.

NIS Databases

The NIS databases served by the NIS server, *ypserv(8)*, have their tables constructed by the *makedbm(8)* program. It needs to have its input as the index key, then a tab, and then the data to be indexed. The **-ByName** and **-ByGid** option are used to read a *group(5)* file and output this format; some input checking is performed.

OPTIONS

The following options are understood:

-ByName *filename*

This option is used to project the given *group(5)* file into a suitable *makedbm(8)* input, indexed by group name.

-ByGid *filename*

This option is used to project the given *group(5)* file into a suitable *makedbm(8)* input, indexed by gid.

-Group *filename*

This option may be used to specify the path of the file containing *group(5)* entries.

-Group_Name_Maximum *number*

The special value "sysconf" uses *sysconf(3)* to determine the system maximum. This option may be used to adjust the maximum length of group names.

-Help

Provide some help with using the *nis-util-group* program.

-Output *filename*

This option may be used with **-ByName** and **-ByGid** to specify where to send the output. The default is to send it to the standard output.

-Passwd *filename*

This option may be used to specify the path of the file containing *passwd(5)* entries.

-Passwd_Name_Maximum *number*

This option may be used to adjust the maximum length of login names. The special value "sysconf" uses *sysconf(3)* to determine the system maximum.

-VERsion

Print the version of the *nis-util-group* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (_) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments "-help", "-HEL" and "-h" are all interpreted to mean the **-Help** option. The argument "-hlp" will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-group* are long, this means ignoring the extra leading '-'. The "--option=value" convention is also understood.

EXIT STATUS

The *nis-util-group* command will exit with a status of 1 on any error. The *nis-util-group* command will only exit with a status of 0 if there are no errors.

SEE ALSO

group(5), *makedbm(1)* *passwd(5)*, *ypserv(8)*,

COPYRIGHT

nis-util-group version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-group* program comes with ABSOLUTELY NO WARRANTY; for details use the '*nis-util-group -VERsion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*nis-util-group -VERsion License*' command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
 /\^* WWW: http://miller.emu.id.au/pmiller/

NAME

`nis-util-group-members` – list the members of Unix groups

SYNOPSIS

`nis-util-group-members` [*option...*] *group-name...*

`nis-util-group-members -Help`

`nis-util-group-members -VERsion`

DESCRIPTION

The *nis-util-group-members* program is used to list the members of the groups named on the command line.

OPTIONS

The following options are understood:

-Group *filename*

This option may be used to specify the path of the file containing *group(5)* entries.

-Group_Name_Maximum *number*

The special value “sysconf” uses *sysconf(3)* to determine the system maximum. This option may be used to adjust the maximum length of group names.

-Help

Provide some help with using the *nis-util-group-members* program.

-Passwd *filename*

This option may be used to specify the path of the file containing *passwd(5)* entries.

-Passwd_Name_Maximum *number*

This option may be used to adjust the maximum length of login names. The special value “sysconf” uses *sysconf(3)* to determine the system maximum.

-VERsion

Print the version of the *nis-util-group-members* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (`_`) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments “-help”, “-HEL” and “-h” are all interpreted to mean the **-Help** option. The argument “-hlp” will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-group-members* are long, this means ignoring the extra leading ‘-’. The “--*option=value*” convention is also understood.

EXIT STATUS

The *nis-util-group-members* command will exit with a status of 1 on any error. The *nis-util-group-members* command will only exit with a status of 0 if there are no errors.

COPYRIGHT

nis-util-group-members version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-group-members* program comes with ABSOLUTELY NO WARRANTY; for details use the ‘*nis-util-group-members -VERsion License*’ command. This is free software and you are welcome to redistribute it under certain conditions; for details use the ‘*nis-util-group-members -VERsion License*’ command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
^^* WWW: http://miller.emu.id.au/pmiller/

NAME

nis-util-hosts – check and manipulate *hosts(5)* files

SYNOPSIS

nis-util-hosts **-Check** *filename*
nis-util-hosts **-ByName** *filename* [**-Output** *filename*]
nis-util-hosts **-ByAddress** *filename* [**-Output** *filename*]
nis-util-hosts **-Help**
nis-util-hosts **-VERSion**

DESCRIPTION

The *nis-util-hosts* program is used to check and manipulate *hosts(5)* files.

Checking The File

The first synopsis line shows how to validate the contents of a *hosts(5)* file. Each field is checked to make sure it contains valid contents.

IP Address

The IP Address column is required to be valid, and in the canonical form. The exact form is necessary because NIS clients query by string, not by the more logical 4-byte binary representation. No duplicates are permitted. The file is required to be sorted by IP address, this catches many of the typographical errors when entering the numbers.

Host Names

The host names are required to be valid host names, according to RFC xxxx.

The overall line length (not including including the newline) must be less than 512 characters. This is a NIS protocol limitation.

NIS Databases

The NIS databases served by the NIS server, *ypserv(8)*, have their tables constructed by the *makedbm(8)* program. It needs to have its input as the index key, then a tab, and then the data to be indexed. The **-ByName** and **-ByAddress** options are used to read a *hosts(5)* file and output this format; some input checking is performed.

OPTIONS

The following options are understood:

-HosTs *filename*

This option is used to specify the *hosts(5)* file to be checked.

-ByName *filename*

This option is used to project the given *hosts(5)* file into a suitable *makedbm(8)* input, indexed by host name.

-ByAddress *filename*

This option is used to project the given *hosts(5)* file into a suitable *makedbm(8)* input, indexed by IP address.

-Output *filename*

This option may be used with **-ByName** and **-ByAddress** to specify where to send the output. The default is to send it to the standard output.

-Help

Provide some help with using the *nis-util-hosts* program.

-VERSion

Print the version of the *nis-util-hosts* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (_) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both,

case is not important.

For example: the arguments "--help", "--HEL" and "-h" are all interpreted to mean the **-Help** option. The argument "--hlp" will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-hosts* are long, this means ignoring the extra leading '-'. The "--option=value" convention is also understood.

EXIT STATUS

The *nis-util-hosts* command will exit with a status of 1 on any error. The *nis-util-hosts* command will only exit with a status of 0 if there are no errors.

SEE ALSO

hosts(5), *makedbm(1)* *ypserv(8)*,

COPYRIGHT

nis-util-hosts version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-hosts* program comes with ABSOLUTELY NO WARRANTY; for details use the '*nis-util-hosts -VERSiOn License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*nis-util-hosts -VERSiOn License*' command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
^/^* WWW: http://miller.emu.id.au/pmiller/

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on

consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

one line to give the program's name and a brief idea of what it does.

Copyright (C) 19yy *name of author*

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19yy *name of author*

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

NAME

nis-util-netgroup – check and manipulate netgroup files

SYNOPSIS

```

nis-util-netgroup -Check filename -Hosts filename -Passwd filename
nis-util-netgroup -Map filename [ -Output filename ]
nis-util-netgroup -ByHost filename [ -Output filename ]
nis-util-netgroup -ByUser filename [ -Output filename ]
nis-util-netgroup -Help
nis-util-netgroup -VERSion

```

DESCRIPTION

The *nis-util-netgroup* program is used to check and manipulate *netgroup(5)* files.

Checking The File

The first synopsis line shows how to validate the contents of the network groups file, which is expected to be in *netgroup(5)* format. Each field is checked to make sure it contains valid contents. Netgroup names and member lists are checked for duplicates.

- name The netgroup name is required to start with a lower-case letter followed by lower-case letters, digits, underscores (_) or hyphens (-). User names are limited to at most 8 characters.
- host If a host is specified, it is required to be named in the hosts file specified by the **-Host** option. This file is expected to be in *hosts(5)* format.
- user If a user is specified, it is required to be named in the password file specified by the **-Passwd** option. This file is expected to be in *passwd(5)* format.
- domain You are required to leave the domain field empty. (The domain field refers to the domain in which the triple is valid, not the domain containing the the trusted host, thus setting it to anything other than the wildcard is of no practical value.)

If one netgroup reference another, the referenced netgroup is required to exist. Infinite loops of mutually referencing netgroups are not permitted.

The overall line length (not including including the newline) must be less than 512 characters. This is a NIS protocol limitation.

NIS Databases

The NIS databases served by the NIS server, *ypserv(8)*, have their tables constructed by the *makedbm(8)* program. It needs to have its input as the index key, then a tab, and then the data to be indexed. The **-Map**, **-ByHost** and **-ByUser** options are used to read a *netgroup(5)* file and output this format; some input checking is performed.

OPTIONS

The following options are understood:

-By-Host *filename*

This option is used to specify the name of a *netgroup(5)* file to be inverted by host name. This generates output suitable for use as *makedbm(1)* input. Some input checking is performed.

-Map *filename*

This option is used to specify the name of a *netgroup(5)* file to be prepared for use as *makedbm(1)* input. All comments and line continuations are removed. Some input checking is performed.

-By-User *filename*

This option is used to specify the name of a *netgroup(5)* file to be inverted by user name. This generates output suitable for use as *makedbm(1)* input. Some input checking is performed.

-Check *filename*

This option is used to specify the name of a *netgroup(5)* file to be checked for correctness.

-Help

Provide some help with using the *nis-util-netgroup* program.

-Hosts

This option may be used to tell the `-Check` option the location of a *hosts(5)* file to validate against.

-Output

This option may be used to specify the output file for the `-Map`, `-ByHost` and `-ByUser` options. The default is to use the standard output.

-Passwd filename

This option may be used to specify the path of the file containing *passwd(5)* entries.

-Passwd_Name_Maximum number

This option may be used to adjust the maximum length of login names. The special value "sysconf" uses *sysconf(3)* to determine the system maximum.

-VERsion

Print the version of the *nis-util-netgroup* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (`_`) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments "`-help`", "`-HEL`" and "`-h`" are all interpreted to mean the **-Help** option. The argument "`-hlp`" will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-netgroup* are long, this means ignoring the extra leading `'-'`. The "`--option=value`" convention is also understood.

EXIT STATUS

The *nis-util-netgroup* command will exit with a status of 1 on any error. The *nis-util-netgroup* command will only exit with a status of 0 if there are no errors.

COPYRIGHT

nis-util-netgroup version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-netgroup* program comes with ABSOLUTELY NO WARRANTY; for details use the '*nis-util-netgroup -VERsion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*nis-util-netgroup -VERsion License*' command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
 ^^* WWW: http://miller.emu.id.au/pmiller/

NAME

nis-util-netid – check *inetd.conf*(5) files

SYNOPSIS

nis-util-netid [*option...*] *filename...*

nis-util-netid -Help

nis-util-netid -VERSion

DESCRIPTION

The *nis-util-netid* program is used to

OPTIONS

The following options are understood:

-Help

Provide some help with using the *nis-util-netid* program.

-VERSion

Print the version of the *nis-util-netid* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (`_`) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments "`-help`", "`-HEL`" and "`-h`" are all interpreted to mean the **-Help** option. The argument "`-hlp`" will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-netid* are long, this means ignoring the extra leading `'-'`. The "`--option=value`" convention is also understood.

EXIT STATUS

The *nis-util-netid* command will exit with a status of 1 on any error. The *nis-util-netid* command will only exit with a status of 0 if there are no errors.

COPYRIGHT

nis-util-netid version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-netid* program comes with ABSOLUTELY NO WARRANTY; for details use the `'nis-util-netid -VERSion License'` command. This is free software and you are welcome to redistribute it under certain conditions; for details use the `'nis-util-netid -VERSion License'` command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
 ^\^* WWW: http://miller.emu.id.au/pmiller/

NAME

nis-util-netmasks – check and manipulate *netmasks(5)* files

SYNOPSIS

```

nis-util-netmasks -Check filename -Netmasks filename
nis-util-netmasks -By_Service_Name filename [ -Output filename ]
nis-util-netmasks -By_Name filename [ -Output filename ]
nis-util-netmasks -Help
nis-util-netmasks -VERSion

```

DESCRIPTION

The *nis-util-netmasks* program is used to check and manipulate *netmasks(5)* files.

Checking the File

The first synopsis line shows how to validate the contents of a *services(5)* file. Each field is checked to make sure it contains valid contents.

network number

The network number column is required to be a correctly formed dotted quad. Duplicate network numbers are not permitted.

network mask

The network mask column is required to be a correctly formed dotted quad. The network mask is required to be consistent with the network number.

NIS Databases

The NIS databases served by the NIS server, *ypserv(8)*, have their tables constructed by the *makedbm(8)* program. It needs to have its input as the index key, then a tab, and then the data to be indexed. The **-ByName** and **-ByServiceName** options are used to read a *netmasks(5)* file and output this format; some input checking is performed.

OPTIONS

The following options are understood:

-By-Service_Name *filename*

This option is used to project the given *netmasks(5)* file into a suitable *makedbm(8)* input, indexed by network number.

-By-Name *filename*

This option is used to project the given *netmasks(5)* file into a suitable *makedbm(8)* input, indexed by network mask.

-Check *filename*

This option is used to specify the *netmasks(5)* file to be checked.

-Help

Provide some help with using the *nis-util-netmasks* program.

-Networks *filename*

This option is used to specify the *networks(5)* file to check the netmasks against.

-Output *filename*

This option may be used with **-ByAddress** and **-ByName** to specify where to send the output. The default is to send it to the standard output.

-VERSion

Print the version of the *nis-util-netmasks* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (_) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments "--help", "--HEL" and "--h" are all interpreted to mean the **--Help** option. The argument "--hlp" will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-netmasks* are long, this means ignoring the extra leading '-'. The "--option=value" convention is also understood.

EXIT STATUS

The *nis-util-netmasks* command will exit with a status of 1 on any error. The *nis-util-netmasks* command will only exit with a status of 0 if there are no errors.

COPYRIGHT

nis-util-netmasks version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-netmasks* program comes with ABSOLUTELY NO WARRANTY; for details use the '*nis-util-netmasks -VERsion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*nis-util-netmasks -VERsion License*' command.

AUTHOR

Peter Miller	E-Mail:	pmiller@opensource.org.au
^^*	WWW:	http://miller.emu.id.au/pmiller/

NAME

nis-util-networks – check and manipulate *networks(5)* files

SYNOPSIS

nis-util-networks **-Check** *filename* **-Netmasks** *filename*
nis-util-networks **-By_Address** *filename* [**-Output** *filename*]
nis-util-networks **-By_Name** *filename* [**-Output** *filename*]
nis-util-networks **-Help**
nis-util-networks **-VERSion**

DESCRIPTION

The *nis-util-networks* program is used to check and manipulate *networks(5)* files.

Checking the File

The first synopsis line shows how to validate the contents of a *networks(5)* file. Each field is checked to make sure it contains valid contents.

- name** A network name may be up to 40 characters taken from the set of lowercase letters, digits, and the punctuation character hyphen. It must start with a letter, and end with a letter or digit. Duplicate names are not permitted.
- number** The network number column is required to be a correctly formed dotted quad. The rows should be in ascending order, as this catches common typographical errors.
- aliases** A network alias may be up to 40 characters taken from the set of letters, digits, and the punctuation character hyphen. It must start with a letter, and end with a letter or digit. Duplicate names are not permitted.

The overall line length (not including including the newline) must be less than 512 characters. This is a NIS service limitation, but it isn't usually a problem with this file.

NIS Databases

The NIS databases served by the NIS server, *ypserv(8)*, have their tables constructed by the *makedbm(8)* program. It needs to have its input as the index key, then a tab, and then the data to be indexed. The **-ByAddress** and **-ByName** options are used to read a *networks(5)* file and output this format; some input checking is performed.

OPTIONS

The following options are understood:

- By-Address** *filename*
This option is used to project the given *networks(5)* file into a suitable *makedbm(8)* input, indexed by network number.
- By-Name** *filename*
This option is used to project the given *networks(5)* file into a suitable *makedbm(8)* input, indexed by network name.
- Check** *filename*
This option is used to specify the *networks(5)* file to be checked.
- Help**
Provide some help with using the *nis-util-networks* program.
- Output** *filename*
This option may be used with **-ByAddress** and **-ByName** to specify where to send the output. The default is to send it to the standard output.
- VERSion**
Print the version of the *nis-util-networks* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (_) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments "--help", "--HEL" and "--h" are all interpreted to mean the **--Help** option. The argument "--hlp" will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-networks* are long, this means ignoring the extra leading '-'. The "--option=value" convention is also understood.

EXIT STATUS

The *nis-util-networks* command will exit with a status of 1 on any error. The *nis-util-networks* command will only exit with a status of 0 if there are no errors.

COPYRIGHT

nis-util-networks version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-networks* program comes with ABSOLUTELY NO WARRANTY; for details use the '*nis-util-networks -VERsion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*nis-util-networks -VERsion License*' command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
^^* WWW: http://miller.emu.id.au/pmiller/

NAME

nis-util-passwd – check and manipulate *passwd(5)* files

SYNOPSIS

```

nis-util-passwd -Passwd filename -Group filename
nis-util-passwd -ByName filename [ -Output filename ]
nis-util-passwd -ByUid filename [ -Output filename ]
nis-util-passwd -Help
nis-util-passwd -VERSIon

```

DESCRIPTION

The *nis-util-passwd* program is used to check and manipulate *passwd(5)* files.

Checking The File

The first synopsis line shows how to validate the contents of the password file, which is expected to be in *passwd(5)* format. Each field is checked to make sure it contains valid contents. Login names and UIDs are checked for duplicates.

- name** The user name is required to start with a lower-case letter followed by lower-case letters, digits, underscores (`_`) or hyphens (`-`). User names are limited to at most 8 characters.
- uid** The user ID is required to be a non-negative decimal number, in its simplest form. User IDs are required to be in ascending order, because this catches many typographical errors in the numbers.
- gid** The group ID is required to be a non-negative decimal number, in its simplest form. The group ID must refer to an existing group in the *group* file.
- home** The home directory is required to be an absolute path. Warnings are issued if it does not have matching user and group IDs.
- shell** The shell is required to be an absolute path.

The overall line length (not including including the newline) must be less than 512 characters. This is a NIS protocol limitation.

NIS Databases

The NIS databases served by the NIS server, *ypserv(8)*, have their tables constructed by the *makedbm(8)* program. It needs to have its input as the index key, then a tab, and then the data to be indexed. The **-ByName** and **-ByUid** options are used to read a passwd file and output this format; some input checking is performed.

OPTIONS

The following options are understood:

- ByName** *filename*
This option is used to project the given passwd file into a suitable *makedbm(8)* input, indexed by login name.
- ByUid** *filename*
This option is used to project the given passwd file into a suitable *makedbm(8)* input, indexed by uid.
- Group** *filename*
This option may be used to specify the path of the file containing *group(5)* entries.
- Group_Name_Maximum** *number*
The special value “sysconf” uses *sysconf(3)* to determine the system maximum. This option may be used to adjust the maximum length of group names.
- Help**
Provide some help with using the *nis-util-passwd* program.
- Output** *filename*
This option may be used with **-ByName** and **-ByUid** to specify where to send the output. The default is to send it to the standard output.

-Passwd *filename*

This option may be used to specify the path of the file containing *passwd(5)* entries.

-Passwd_Name_Maximum *number*

This option may be used to adjust the maximum length of login names. The special value "sysconf" uses *sysconf(3)* to determine the system maximum.

-VERsion

Print the version of the *nis-util-passwd* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (_) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments "-help", "-HEL" and "-h" are all interpreted to mean the **-Help** option. The argument "-hlp" will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-passwd* are long, this means ignoring the extra leading '-'. The "--option=value" convention is also understood.

EXIT STATUS

The *nis-util-passwd* command will exit with a status of 1 on any error. The *nis-util-passwd* command will only exit with a status of 0 if there are no errors.

SEE ALSO

passwd(5), *group(5)*, *ypserv(8)*, *makedbm(1)*

COPYRIGHT

nis-util-passwd version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-passwd* program comes with ABSOLUTELY NO WARRANTY; for details use the '*nis-util-passwd -VERsion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*nis-util-passwd -VERsion License*' command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
 ^\^* WWW: http://miller.emu.id.au/pmiller/

NAME

nis-util-protocols – check and manipulate *protocols(5)* files

SYNOPSIS

nis-util-protocols –*Check filename*
nis-util-protocols –*By-NAme filename* [–**Output filename**]
nis-util-protocols –*By-NUmber filename* [–**Output filename**]
nis-util-protocols -Help
nis-util-protocols -VERSion

DESCRIPTION

The *nis-util-protocols* program is used to check and manipulate *protocols(5)* files.

Checking the File

The first synopsis line shows how to validate the contents of a *protocols(5)* file. Each field is checked to make sure it contains valid contents.

- name** A protocol name may be up to 40 characters taken from the set of lowercase letters, digits, and the punctuation character hyphen. It must start with a letter, and end with a letter or digit. Duplicate names are not permitted.
- number** The protocol number column is required to be positive, and less than 65536. The rows should be in ascending order, as this catches common typographical errors.
- aliases** A protocol alias may be up to 40 characters taken from the set of letters, digits, and the punctuation character hyphen. It must start with a letter, and end with a letter or digit. Duplicate names are not permitted.

The overall line length (not including including the newline) must be less than 512 characters. This is a NIS protocol limitation, but it isn't usually a problem with this file.

NIS Databases

The NIS databases served by the NIS server, *ypserv(8)*, have their tables constructed by the *makedbm(8)* program. It needs to have its input as the index key, then a tab, and then the data to be indexed. The –**ByName** and –**ByNumber** options are used to read a *protocols(5)* file and output this format; some input checking is performed.

OPTIONS

The following options are understood:

–*By-NAme filename*

This option is used to project the given *protocols(5)* file into a suitable *makedbm(8)* input, indexed by protocol name.

–*By-NUmber filename*

This option is used to project the given *protocols(5)* file into a suitable *makedbm(8)* input, indexed by protocol number.

–*Check filename*

This option is used to specify the *ethers(5)* file to be checked.

-Help

Provide some help with using the *nis-util-protocols* program.

–*Output filename*

This option may be used with –**ByName** and –**ByNumber** to specify where to send the output. The default is to send it to the standard output.

-VERSion

Print the version of the *nis-util-protocols* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (_) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments "--help", "--HEL" and "--h" are all interpreted to mean the **-Help** option. The argument "--hlp" will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-protocols* are long, this means ignoring the extra leading '-'. The "--option=value" convention is also understood.

EXIT STATUS

The *nis-util-protocols* command will exit with a status of 1 on any error. The *nis-util-protocols* command will only exit with a status of 0 if there are no errors.

SEE ALSO

protocols(5)

description of the file format

ypserv(8)

NIS server

makedbm(1)

create or dump a *ypserv*(8) database file

<http://www.iana.org/assignments/service-names>

Towards the top there is a paragraph outlining the rules for protocol names and service names.

<http://www.iana.org/assignments/protocol-numbers>

This contains the official list of protocol names and numbers. (Note that the names do not follow the rules.)

COPYRIGHT

nis-util-protocols version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-protocols* program comes with ABSOLUTELY NO WARRANTY; for details use the '*nis-util-protocols -VERsion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*nis-util-protocols -VERsion License*' command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
 ^^* WWW: http://miller.emu.id.au/pmiller/

NAME

nis-util-services – check and manipulate *services(5)* files

SYNOPSIS

nis-util-services *–Check filename –Protocols filename*
nis-util-services *–By-Name filename [–Output filename]*
nis-util-services *–By-Service-Name filename [–Output filename]*
nis-util-services -Help
nis-util-services -VERSion

DESCRIPTION

The *nis-util-services* program is used to check and manipulate *services(5)* files.

Checking the File

The first synopsis line shows how to validate the contents of a *services(5)* file. Each field is checked to make sure it contains valid contents.

- name** A service name may be up to 40 characters taken from the set of lowercase letters, digits, and the punctuation character hyphen. It must start with a letter, and end with a letter or digit. Duplicate names are not permitted.
- number** The protocol number column is required to be positive, and less than 65536. The rows should be in ascending order, as this catches common typographical errors.
- aliases** A service alias may be up to 40 characters taken from the set of letters, digits, and the punctuation character hyphen. It must start with a letter, and end with a letter or digit. Duplicate names are not permitted.

The overall line length (not including including the newline) must be less than 512 characters. This is a NIS service limitation, but it isn't usually a problem with this file.

NIS Databases

The NIS databases served by the NIS server, *ypserv(8)*, have their tables constructed by the *makedbm(8)* program. It needs to have its input as the index key, then a tab, and then the data to be indexed. The **–ByName** and **–ByServiceName** options are used to read a *services(5)* file and output this format; some input checking is performed.

Note: despite the names, **–byname** actually means to generate a map indexed by port number, and **–byservicename** means to generate a map indexed by service name and service aliases. The original YP authors came up with this brilliant piece of cognitive dissonance, and we are stuck with it.

OPTIONS

The following options are understood:

–By-Name filename

This option is used to project the given *services(5)* file into a suitable *makedbm(8)* input, indexed by service name.

–By-Service-Name filename

This option is used to project the given *services(5)* file into a suitable *makedbm(8)* input, indexed by service number.

–Check filename

This option is used to specify the *ethers(5)* file to be checked.

–Help

Provide some help with using the *nis-util-services* program.

–Output filename

This option may be used with **–ByName** and **–ByServiceName** to specify where to send the output. The default is to send it to the standard output.

-Protocols filename

This option is used to specify the *protocols(5)* file to check the protocol names against.

-VERSion

Print the version of the *nis-util-services* program being executed.

All other options will produce a diagnostic error.

All options may be abbreviated; the abbreviation is documented as the upper case letters, all lower case letters and underscores (_) are optional. You must use consecutive sequences of optional letters.

All options are case insensitive, you may type them in upper case or lower case or a combination of both, case is not important.

For example: the arguments "-help", "-HEL" and "-h" are all interpreted to mean the **-Help** option. The argument "-hlp" will not be understood, because consecutive optional characters were not supplied.

Options and other command line arguments may be mixed arbitrarily on the command line.

The GNU long option names are understood. Since all option names for *nis-util-services* are long, this means ignoring the extra leading '-'. The "--option=value" convention is also understood.

EXIT STATUS

The *nis-util-services* command will exit with a status of 1 on any error. The *nis-util-services* command will only exit with a status of 0 if there are no errors.

SEE ALSO

services(5)

description of the file format

ypserv(8)

NIS server

makedbm(1)

create or dump a *ypserv(8)* database file

<http://www.iana.org/assignments/service-names>

Towards the top there is a paragraph outlining the rules for protocol names and service names.

<http://www.iana.org/assignments/port-numbers>

This contains the official list of port names and numbers.

COPYRIGHT

nis-util-services version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-services* program comes with ABSOLUTELY NO WARRANTY; for details use the '*nis-util-services -VERSion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*nis-util-services -VERSion License*' command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
 ^^* WWW: http://miller.emu.id.au/pmiller/

NAME

/etc/nis-util.conf – nios-util configuration file

DESCRIPTION

The nis-util collection of programs may be configured using a variety of files and environment variables.

Sources

in the following precedence order from lowest to highest:

built-in A configuration item's hard coded default. Usually conservative.

/etc/nis-util.conf

The system wide default values, if present. Not usually required, you should be conservative with setting you place here.

\$HOME/.nis-util.rc

The per-user default values, if present. Not usually required.

NIS_UTIL

The NIS_UTIL environment variable, if present. This is formatted identically to the file format, including newlines.

NIS_UTIL_name

The utility specific environment variable, if present. This is formatted identically to the file format, including newlines. the name of the environment variable depends on the program being executed, for example the *nis-util-passwd(1)* command would look for the NIS_UTIL_PASSWD environment variable.

-configuration=filename

A configuration file specified on the command line.

A specific command line option, as it relates to a specific item,

usually in the [global] section. For example, the **-no-warning** option set the “[global] warning” configuration item.

If there is an environment variable “NIS_UTIL_TEST_MODE=true” setting, all configuration sources except for built-in and command line variants are ignored.

FILE FORMAT

The file is formatted similarly to a classic .ini file. It has multiple sections. Each section starts a line with a section name in [square brackets], and is followed with lines for the form “name = value”.

Blank lines are ignored. Comments start with a hash (“#”) character, preceded by white space or the start of the line, and extend to the end of the line. Note that text such as “a@b#c” does not contain a comment.

Section Names

The global section name is [global], and contains configuration items of interest to many of the nis-util programs and NIS maps.

Each NIS map has its own section, controlling what is and isn't checked. For example: the “passwd” map, corresponding to the “/etc/passwd” file, is in section “[passwd]”.

BUILTIN DEFAULTS

The default values of the various configuration items are as follows:

```
[global]
warning = true
pedantic = true
```

```
[passwd]
name-length-maximum = 8
```

```
[group]
name-length-maximum = 8
```

```
[hosts]
name-length-maximum = 63
```

The default defaults are: false, for boolean values; zero, for integer values; and the empty string, for string values.

COPYRIGHT

nis-util-conf version 1.0

Copyright © 2001, 2002, 2003, 2007, 2008, 2009, 2011, 2012 Peter Miller

The *nis-util-conf* program comes with ABSOLUTELY NO WARRANTY; for details use the '*nis-util-conf -VERsion License*' command. This is free software and you are welcome to redistribute it under certain conditions; for details use the '*nis-util-conf -VERsion License*' command.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
^/* WWW: http://miller.emu.id.au/pmiller/