NAME

cvtsudoers - convert between sudoers file formats

SYNOPSIS

cvtsudoers [**-ehMpV**] [**-b** *dn*] [**-c** *conf_file*] [**-d** *deftypes*] [**-f** *output_format*] [**-i** *input_format*] [**-I** *increment*] [**-m** *filter*] [**-o** *output_file*] [**-O** *start_point*] [**-P** *padding*] [**-s** *sections*] [*input_file*]

DESCRIPTION

cvtsudoers can be used to convert between *sudoers* security policy file formats. The default input format is sudoers. The default output format is LDIF. It is only possible to convert a *sudoers* file that is syntactically correct.

If no *input_file* is specified, or if it is '-', the policy is read from the standard input. By default, the result is written to the standard output.

The options are as follows:

-b *dn*, **--base**=*dn*
: The base DN (distinguished name) that will be used when performing LDAP queries. Typically this is of the form ou=SUDOers,dc=my-domain,dc=com for the domain my-domain.com. If this option is not specified, the value of the SUDOERS_BASE environment variable will be used instead. Only necessary when converting to LDIF format.

-c *conf_file*, **--config**=*conf_file*
: Specify the path to the configuration file. Defaults to */etc/cvtsudoers.conf*.

-d *deftypes*, **--defaults**=*deftypes*
: Only convert Defaults entries of the specified types. One or more Defaults types may be specified, separated by a comma (','). The supported types are:

  all — All Defaults entries.

  global — Global Defaults entries that are applied regardless of user, runas, host or command.

  user — Per-user Defaults entries.

  runas — Per-runas user Defaults entries.

host Per-host Defaults entries.

command Per-command Defaults entries.

See the **Defaults** section in sudoers(5) for more information.

If the **-d** option is not specified, all Defaults entries will be converted.

-e, **--expand-aliases**
Expand aliases in *input_file*. Aliases are preserved by default when the output *format* is JSON or sudoers.

-f *output_format*, **--output-format**=*output_format*
Specify the output format (case-insensitive). The following formats are supported:

JSON JSON (JavaScript Object Notation) files are usually easier for third-party applications to consume than the traditional *sudoers* format. The various values have explicit types which removes much of the ambiguity of the *sudoers* format.

LDIF LDIF (LDAP Data Interchange Format) files can be imported into an LDAP server for use with sudoers.ldap(5).

Conversion to LDIF has the following limitations:

- Command, host, runas and user-specific Defaults lines cannot be translated as they don't have an equivalent in the sudoers LDAP schema.

- Command, host, runas and user aliases are not supported by the sudoers LDAP schema so they are expanded during the conversion.

sudoers Traditional sudoers format. A new sudoers file will be reconstructed from the parsed input file. Comments are not preserved and data from any include files will be output inline.

-h, **--help** Display a short help message to the standard output and exit.

-i *input_format*, **--input-format**=*input_format*
Specify the input format. The following formats are supported:

LDIF LDIF (LDAP Data Interchange Format) files can be exported from an LDAP server to convert security policies used by sudoers.ldap(5). If a base DN (distinguished name) is specified, only sudoRole objects that match the base DN will be processed. Not all sudoOptions specified in a sudoRole can be translated from LDIF to sudoers format.

sudoers Traditional sudoers format. This is the default input format.

-I *increment*, **--increment**=*increment*

When generating LDIF output, increment each sudoOrder attribute by the specified number. Defaults to an increment of 1.

-m *filter*, **--match**=*filter*

Only output rules that match the specified *filter*. A *filter* expression is made up of one or more **key =** *value* pairs, separated by a comma (‘,’). The **key** may be "user", "group" or "host". For example, **user** = *operator* or **host** = *www*. An upper-case User_Alias or Host_Alias may be specified as the "user" or "host".

A matching *sudoers* rule may also include users, groups and hosts that are not part of the *filter*. This can happen when a rule includes multiple users, groups or hosts. To prune out any non-matching user, group or host from the rules, the **-p** option may be used.

By default, the password and group databases are not consulted when matching against the filter so the users and groups do not need to be present on the local system (see the **-M** option). Only aliases that are referenced by the filtered policy rules will be displayed.

-M, **--match-local**

When the **-m** option is also specified, use password and group database information when matching users and groups in the filter. Only users and groups in the filter that exist on the local system will match, and a user’s groups will automatically be added to the filter. If the **-M** is *not* specified, users and groups in the filter do not need to exist on the local system, but all groups used for matching must be explicitly listed in the filter.

-o *output_file*, **--output**=*output_file*

Write the converted output to *output_file*. If no *output_file* is specified, or if it is ‘-’, the converted *sudoers* policy will be written to the standard output.

-O *start_point*, **--order-start**=*start_point*

When generating LDIF output, use the number specified by *start_point* in the sudoOrder attribute of the first sudoRole object. Subsequent sudoRole object use a sudoOrder value

generated by adding an *increment*, see the **-I** option for details. Defaults to a starting point of 1. A starting point of 0 will disable the generation of sudoOrder attributes in the resulting LDIF file.

-p, **--prune-matches**

When the **-m** option is also specified, **cvtsudoers** will prune out non-matching users, groups and hosts from matching entries.

-P *padding*, **--padding**=*padding*

When generating LDIF output, construct the initial sudoOrder value by concatenating *order_start* and *increment*, padding the *increment* with zeros until it consists of *padding* digits. For example, if *order_start* is 1027, *padding* is 3, and *increment* is 1, the value of sudoOrder for the first entry will be 1027000, followed by 1027001, 1027002, etc. If the number of sudoRole entries is larger than the padding would allow, **cvtsudoers** will exit with an error. By default, no padding is performed.

-s *sections*, **--suppress**=*sections*

Suppress the output of specific *sections* of the security policy. One or more section names may be specified, separated by a comma (‘,’). The supported section name are: **defaults**, **aliases** and **privileges** (which may be shortened to **privs**).

-V, **--version**

Print the **cvtsudoers** and *sudoers* grammar versions and exit.

Options in the form "keyword = value" may also be specified in a configuration file, */etc/cvtsudoers.conf* by default. The following keywords are recognized:

defaults = *deftypes*

See the description of the **-d** command line option.

expand_aliases = *yes* | *no*

See the description of the **-e** command line option.

input_format = *ldif* | *sudoers*

See the description of the **-i** command line option.

match = *filter*

See the description of the **-m** command line option.

order_increment = *increment*

See the description of the **-I** command line option.

order_start = *start_point*
See the description of the **-O** command line option.

output_format = *json* | *ldif* | *sudoers*
See the description of the **-f** command line option.

padding = *padding*
See the description of the **-P** command line option.

prune_matches = *yes* | *no*
See the description of the **-p** command line option.

sudoers_base = *dn*
See the description of the **-b** command line option.

suppress = *sections*
See the description of the **-s** command line option.

Options on the command line will override values from the configuration file.

FILES

/etc/cvtsudoers.conf default configuration for cvtsudoers

EXAMPLES

Convert */etc/sudoers* to LDIF (LDAP Data Interchange Format) where the *ldap.conf* file uses a *sudoers_base* of my-domain,dc=com, storing the result in *sudoers.ldif*:

```
$ cvtsudoers -b ou=SUDOers,dc=my-domain,dc=com -o sudoers.ldif \
    /etc/sudoers
```

Convert */etc/sudoers* to JSON format, storing the result in *sudoers.json*:

```
$ cvtsudoers -f json -o sudoers.json /etc/sudoers
```

Parse */etc/sudoers* and display only rules that match user *ambrose* on host *hastur*:

```
$ cvtsudoers -f sudoers -m user=ambrose,host=hastur /etc/sudoers
```

Same as above, but expand aliases and prune out any non-matching users and hosts from the expanded entries.

```
$ cvtsudoers -ep -f sudoers -m user=ambrose,host=hastur /etc/sudoers
```

Convert *sudoers.ldif* from LDIF to traditional *sudoers* format:

```
$ cvtsudoers -i ldif -f sudoers -o sudoers.new sudoers.ldif
```

SEE ALSO

sudoers(5), sudoers.ldap(5), sudo(8)

AUTHORS

Many people have worked on **sudo** over the years; this version consists of code written primarily by:

Todd C. Miller

See the CONTRIBUTORS file in the **sudo** distribution (https://www.sudo.ws/contributors.html) for an exhaustive list of people who have contributed to **sudo**.

BUGS

If you feel you have found a bug in **cvtsudoers**, please submit a bug report at https://bugzilla.sudo.ws/

SUPPORT

Limited free support is available via the sudo-users mailing list, see https://www.sudo.ws/mailman/listinfo/sudo-users to subscribe or search the archives.

DISCLAIMER