Micron

version 1.3.93, 3 January 2024

Sergey Poznyakoff.

Table of Contents

1 Overview

'Micron' is an implementation of the UNIX cron daemon, a program that executes periodically various tasks. It provides a flexible job scheduler that offers complete control over execution of the scheduled commands as well as additional organizational features.

Cron daemons have a long history, dating back to Seventh Edition Unix. The most widespread cron implementations today are *Vixie cron* and its derivatives, originally written by Paul Vixie, and *Dillon's cron*, written by Matt Dillon.

'Micron' is a clean, multi-threaded implementation, written from scratch, which does not share any code with its predecessors. Some of the new features provided by 'micron' are: flexible control over cronjob's standard output and error streams, which can be redirected to syslog, mailed to arbitrary list of emails, appended to a file, or simply ignored, control over the number of cronjob instances that are allowed to run simultaneously, and user-group crontabs. For a complete list, see Chapter 5 [Features], page 15.

While providing new features, 'micron' tries to preserve backward compatibility with the two above-mentioned implementations. In particular, existing crontabs from *Vixie* cron can be used with 'micron' without changes. The same holds true for per-user crontabs from *Dillon's* cron. System-wide crontabs from *Dillon's* cron will require a minor editing: the user name must be added between the schedule and command.

The implementation consists of two binaries: the main daemon micrond and the crontab utility.

2 Crontabs

A set of *crontab* files contains instructions specifying what commands to run and defining schedules for them. `micrond` reads its crontabs at startup and loads them to memory. When running, it keeps track of crontab modifications and updates its in-memory tables as soon as such are detected.

2.1 Crontab Groups

Crontabs are stored in several locations, collectively known as *crontab groups* or *crongroups*, for short. These are:

master crontab
: The file `/etc/crontab`. It is used for site-wide cronjobs, i.e. tasks necessary for the normal functioning of the system.

system crontabs
: A collection of crontab files in the `/etc/cron.d` directory. These are used for per-package cronjobs.

user crontabs
: Crontab files located in `/var/spool/cron/crontabs` contain per-user cronjobs and are editable by users.

user group crontabs
: A special crontab group intended for use with pseudo-accounts, such as '`apache`' or '`bind`'. Crontabs of this group are located in subdirectories of `/var/spool/cron/crongroups` named by the corresponding account. This crontab group is described in detail in Section 5.1 [User Group Crontabs], page 15.

Each *active* (i.e. non-empty and non-comment) line in a crontab specifies a schedule and a command line to be run according to that schedule. Active lines in master and system crontabs specify also login name of the user on behalf of whom the command must be run.

Both master and system crontabs are writable only by the super-user.

User and user group crontabs belong to particular users, and instructions they contain are executed on behalf of their owners. To enable users to manipulate their crontabs, the `crontab` command is provided (see Chapter 4 [crontab], page 13).

2.2 Crontab Format

Crontabs used by `micrond` are mostly compatible with *Vixie* crontabs. They are plain text, line-oriented files. When parsing a crontab, the leading and trailing whitespace on each line is ignored. Comments are introduced by a hash sign (#) appearing as the first non-whitespace character in a line. Comments and empty lines are ignored. Very long lines can be split across several physical lines using backslash as continuation character. Total length of a valid crontab line after removing continuation characters cannot exceed 1024 characters.

Each crontab line is either a cronjob definition or a variable setting.

2.2.1 Cronjob Definition

A *cronjob definition* is a line in crontab that defines a running schedule, a user on whose behalf the job will be run (in system crontabs), and a command to be run.

The fields are delimited by arbitrary amount of whitespace.

Cronjob definition begins with a *cron expression*, which defines schedule for running the command. It consists of five fields in this order:

field	**allowed values**
minute	0-59
hour	0-23
day of month	1-31
month	1-12, or abbreviated month names
day of week	0-7 (0 or 7 is Sunday), or abbreviated day names

A field may contain a single asterisk '`*`', which stands for "each time unit". For example, it means "each minute" when used in the first field, "each hour" in second one, and so on. Asterisk can be followed by a slash and a decimal number, which defines a step within the interval. E.g. '`*/2`' in second field stands for "each second hour".

The field may also be a comma-delimited list of the following constructs:

n A single number. E.g. '`2`' in first field means "second minute of an hour".

i-j Two numbers delimited by a dash define a range. E.g '`2-5`' in first field is "minutes 2,3,4, and 5 (inclusive)". The order is important. For example, the following schedule:

```
55-5 * * * *
```

means "minutes 55 through 59 and 0 through 5 of the next hour".

i-j/n Same as above, but with step of N units, e.g.:

```
10-25/5 * * * *
```

which stands for "minutes 10,15,20,25".

Names can be used in "month" and "day of week" fields. Day and week names are abbreviated to their first three characters (case-insensitive). Ranges or lists of names are allowed as well.

The day of command's execution is determined by two fields: day of month, and day of week. If both are supplied (i.e. are not '`*`'), the result depends on the selected *day field semantics*. There are three cases:

Strict semantics

The command will be run only when both fields match the current time. For example,

```
30 4 1,15 * 5
```

means "run the command at 4:30 am each Friday between the 1st and 15th of each month".

This semantics is the default for `micrond`.

Vixie semantics

The command will be run when either field matches the current time. Thus, the previous example would cause a command to be run at 4:30 am on the 1st and 15th of each month, plus every Friday.

Dillon semantics

If both day of the month and day of week are supplied, the former must be an integer number in range 1-5. Its meaning is Nth such weekday in the month. For example, to run the date command at 11 am on the second and third Monday, Tuesday and Wednesday of each month:

```
0 11 2,3 * mon-wed date
```

To request the last Monday, etc. in a month, use '5'. This will always match the last Monday, etc., even if there are only four Mondays in the month:

```
0 11 1,5 * mon-wed date
```

When the fourth Monday in a month is the last, it will match against both 4 and 5, but will only run once if both are specified.

The semantics to use is selected by setting the `_MICRON_DAY_SEMANTICS` (see [_MICRON_DAY_SEMANTICS], page 8) variable to one of the following values: `strict`, `vixie`, `dillon` (case-insensitive).

Instead of the first five fields, the following macro-definitions can be used:

`@reboot` Run once, at `micrond` startup.

`@yearly`
`@annually`
Run once a year, i.e. '`0 0 1 1 *`'.

`@monthly` Run once a month, '`0 0 1 * *`'.

`@weekly` Run once a week, '`0 0 * * 0`'.

`@daily`
`@midnight`
Run once a day, '`0 0 * * *`'.

`@hourly` Run once an hour, '`0 * * * *`'.

In master crontab (`/etc/crontab`) and system crontabs (located in `/etc/cron.d`), the cron expression is followed by *user name* field, which holds a login name of the user on whose behalf the command will be run.

This field is absent in user personal crontabs, since these are always run on behalf of the user that owns them,

The remainder of line is taken as a shell command to be run when the time matches the schedule. The percent sign in the command line is special: the first occurrence of unescaped and unquoted '`%`' delimits the command line proper and the text that will be piped to its standard input. Any unescaped occurrences of '`%`' to the right of it will be replaced with newlines. To use a literal percent sign in the job command line, either precede it with a backslash or quote it using single or double quotes.

For example, the following command line:

```
mail hostmaster%Hello,%%This is a daily notification.%--%Regards from cron.%
```

will cause mirond to start the mail command and to supply the following text to its standard input:

```
Hello,

This is a daily notification.
--
Regards from cron.
```

Before running the command, micrond changes the directory to the home directory of the user on whose behalf the command is run (as defined by the HOME environment variable, see [HOME], page 8), modifies the environment as requested by variable settings (see Section 2.2.2 [Variable Settings], page 6) and runs the command as

```
$SHELL -c "command"
```

Default SHELL is '/bin/sh' (see [SHELL], page 8).

2.2.2 Variable Settings

Variable settings modify execution environment for subsequent cron commands. Some variables are internal to micrond and modify its behavior. A variable setting has the form

```
name = value
```

where *name* is the variable name and *value* is the value to be assigned to it. Notice several important differences from the similar construct in shell. First of all, optional whitespace is allowed on either side of the equals sign. Secondly, the value is assigned verbatim, after removing any trailing and leading whitespace. No expansions or substitutions take place. If you need this kind of functionality, move environment manipulations to a shell script and invoke it from the cron command.

You don't need to quote *value* even if it contains embedded whitespace. There are only two cases when quoting becomes necessary: to define an empty variable or to preserve leading or trailing whitespace. Both single and double quotes can be used, the only requirement being that both quotes match. Inside a quoted string, a backslash character can be used to escape a quote character or another backslash. To define an empty variable, use a pair of quotes.

The special construct

```
name =
```

unsets the variable *name*.

Each variable setting applies to all cron commands that follow it, until another setting overwrites it or the end of file is encountered.

The following are *built-in variables*. They modify the handling of cron commands that follow them, but are not copied to the command environment. Variables prefixed with '_JOB_' affect only the first cron command that appears after them, whereas variables prefixed with '_MICRON_' affect all commands that follow them, until another assignment of the same variable is encountered or the end of the file is reached. A built-in variable can be assigned a default value in the command line using the -v option (see [initial value], page 12).

`_MICRON_OUTFILE_FACILITY` [Built-in variable]
`_JOB_OUTFILE_FACILITY` [Built-in variable]

The value of this variable is the name of a file. If the variable is set, any output the cronjob produces will be appended to that file.

See Section 2.3.3 [Sending output to file], page 10.

`_MICRON_SYSLOG_FACILITY` [Built-in variable]
`_JOB_SYSLOG_FACILITY` [Built-in variable]

If this variable is set to a meaningful syslog facility, the cronjob output will be logged to that facility (priority '`INFO`'), instead of mailing it the usual way. See Section 2.3.2 [Logging output to syslog], page 9.

Allowed values for this variable are `auth`, `authpriv`, `cron`, `daemon`, `ftp`, `lpr`, `mail`, `news`, `syslog`, `user`, `uucp`, and `local0` through `local7`.

The value `default` means to use the default syslog facility (`cron`), and `off` or `none` disables syslog and reverts to mailing the program output as directed by the `MAILTO` variable (see [MAILTO], page 8).

`_JOB_SYSLOG_TAG` [Built-in variable]

Sets the syslog tag for the next cron command. The tag is used when logging cronjob output to syslog (see Section 2.3.2 [Logging output to syslog], page 9), or appending it to a file (see Section 2.3.3 [Sending output to file], page 10). If not specified, the tag will be constructed as

```
file:line(prog)
```

where *file* and *line* are the file name and line number of the line where the cron command appeared and *prog* is the first word of the command.

The global counterpart of this variable, `_MICRON_SYSLOG_TAG`, is provided for completeness. It is probably of little use, since it sets the same tag for all crontab entries.

`_MICRON_MAXINSTANCES` [Built-in variable]
`_JOB_MAXINSTANCES` [Built-in variable]

An integer value defining how many instances of the same cron job can be running simultaneously. Default is 1, i.e. `micrond` will refuse to start a job if its previous run has not yet terminated.

`_JOB_MAILTO` [Built-in variable]

This variable temporarily masks the `MAILTO` environment variable and disables the variables `_MICRON_SYSLOG_FACILITY` and `_MICRON_OUTFILE` for the next crontab entry. After that, the previous values are restored. Use it to redirect output of a single crontab entry to a particular email address.

The global version of this variable, `_MICRON_MAILTO`, is mostly equivalent to the traditional `MAILTO` variable, except that any existing value of `MAILTO` is retained in the environment.

See Section 2.3.1 [Mailing the cronjob output], page 8, for the detailed discussion of how the output is mailed.

See Section 2.3 [Cronjob Output], page 8,

_MICRON_DAY_SEMANTICS [Built-in variable]
_JOB_DAY_SEMANTICS [Built-in variable]

Defines the day semantics. Allowed values are: `strict`, `vixie`, and `dillon` (case-insensitive). See Section 5.4 [Day Field Semantics], page 16.

The following are environment variables that modify the behavior of `micrond` itself.

MAILTO [Environment]

Defines an email address or a comma-delimited list of email addresses. If any of the cron commands that follow this setting produces anything on its standard output or standard error, the output will be collected and mailed to the emails listed in the `MAILTO` variable. If the variable is undefined, the output will be sent to the owner of the crontab. If `MAILTO` is set to an empty string, no mail will be sent.

See Section 2.3.1 [Mailing the cronjob output], page 8, for the detailed discussion of how the output is mailed.

HOME [Environment]

Defines the working directory from which the subsequent commands will be run. Defaults to home directory of the user on whose behalf the command is run.

SHELL [Environment]

Defines the shell used to start commands. Defaults to '`/bin/sh`'.

2.3 Cronjob Output

When run, a cronjob can print something on its standard output or error streams. The output it produces can be sent via email to a given list of addresses, appended to a file, or reported via syslog. This *output redirection* can be configured either globally, or individually for each job. It is controlled by the following variables:

MAILTO
_MICRON_MAILTO
_JOB_MAILTO

Send the output via email.

_MICRON_SYSLOG_FACILITY
_JOB_SYSLOG_FACILITY

Report each output line via syslog.

_MICRON_OUTFILE
_JOB_OUTFILE

Append the output to a file.

These settings are discussed in detail in the following subsections.

2.3.1 Mailing the cronjob output

The default behavior is to mail the cronjob output to the user on whose behalf the cronjob is run. The message `From:` header is constructed as follows:

```
From: (Cron daemon) <owner@host>
```

where *owner* is the login name of the job owner, and *host* is the name of the host where it was run.

The `Subject:` header contains the owner login name, hostname of the server where the command was run and the command itself, in the following format:

```
    Cron <user@host> command
```

A copy of execution environment is included in the message in the form of additional `X-Cron-Env:` headers, each containing a single environment variable.

The recipient of the message can be altered by setting the `MAILTO`, `_JOB_MAILTO`, and `_MICRON_MAILTO` variables in the crontab.

The `MAILTO` variable is the traditional way of defining the recipients for the job output. It affects all cronjobs in crontab that appear after it, until next definition of any of the three variables discussed, or end of the crontab file, whichever occurs first. It is also reflected in the environment of the job itself. See [MAILTO], page 8.

If `MAILTO` is unset, the default behavior (mailing to the owner) is restored. If it is set to an empty string, cronjob output is discarded.

Notice the difference:

`MAILTO =` Restores default behavior.

`MAILTO = ""`
Discards the output.

The two built-in variables `_JOB_MAILTO` and `_MICRON_MAILTO` behave as `MAILTO`, with the following differences:

The `_JOB_MAILTO` variable, if set, affects only the cronjob that immediately follows it. It is not reflected in the environment. See [_JOB_MAILTO], page 7.

The `_MICRON_MAILTO` variable is rarely used. It works exactly as `MAILTO`, except that the actual value of the latter remains unchanged. It is not reflected in the environment, either.

2.3.2 Logging output to syslog

To send the output of all the cronjobs to the syslog, assign the name of the syslog facility to the built-in variable `_MICRON_SYSLOG_FACILITY`, e.g.:

```
    _MICRON_SYSLOG_FACILITY = local0
```

Valid facility names are `auth`, `authpriv`, `cron`, `daemon`, `ftp`, `lpr`, `mail`, `news`, `syslog`, `user`, `uucp`, and `local0` through `local7`.

Similarly, to log only a single cronjob output, define the variable `_JOB_SYSLOG_FACILITY` immediately before that cronjob.

In the produced log, each line of a cronjob output will be tagged with a fully-qualified name of the cronjob, constructed as:

```
    file:line(prog)
```

You can alter this using the `_JOB_SYSLOG_TAG` variable. The following example illustrates the use of the two per-job syslog variables:

```
    _JOB_SYSLOG_FACILITY = daemon
    _JOB_SYSLOG_TAG = daily_cleanup
    0 1 * * *  root  /usr/local/bin/cleanup
```

2.3.3 Sending output to file

To redirect cronjob output to a file, use the `_MICRON_OUTFILE` or `_JOB_OUTFILE` built-in variables. The value of these variables is the name of file to append the output to. For example, to redirect output of all cronjobs in a crontab to file `/var/log/cronout`, place the following at the beginning of the crontab:

```
_MICRON_OUTFILE = /var/log/cronout
```

For example, when running `micrond` in a docker container, you can redirect the output of all cronjobs to the container log as follows:

```
_MICRON_OUTFILE = /proc/1/fd/1
```

The `_JOB_OUTFILE` variable affects only the cronjob that follows it.

Output from each job is appended to the output file in a single chunk, which appears between the following two delimiter lines:

```
ts_start: tag output begins
```

and

```
ts_end: tag output ends
```

where *ts_start* and *ts_end* are job start and termination timestamps in ISO 8601 format, and *tag* is the cronjob tag, formatted as described in [cronjob tag], page 9. If the `_SYSLOG_TAG` variable is defined, its value will be used instead.

For example:

```
2021-10-23T03:00:00: /etc/cron.d/backup:16(daily_backup) output begins
.
.
.
2021-10-23T03:07:15: /etc/cron.d/backup:16(daily_backup) output ends
```

3 micrond

`micrond` executes commands periodically as directed by one or more *crontabs*. See Chapter 2 [Crontabs], page 3, for a discussion of crontab locations and format.

Normally, a cron job is not run if its previous instance is still running. For instance, if a cron job scheduled to run each minute takes three minutes to finish, it will be actually run once in three minutes. This behavior can be altered by setting the `_MICRON_MAXINSTANCES` variable to the desired number of cron job instances that can be run simultaneously (use `_JOB_MAXINSTANCES` to change the setting for one job only, see [_MICRON_MAXINSTANCES], page 7).

Depending on cronjob settings, any data the running job produces on its standard output and standard error can be either logged via syslog in real time, or captured and, upon termination of the job, mailed to the job owner, or appended to a file (see Section 2.3 [Cronjob Output], page 8).

On GNU/Linux systems, `micrond` monitors each opened crontab for modifications and re-reads it as soon as it is written to disk. On other systems, it checks crontab modification times each minute and re-reads those crontabs for which this value has changed.

3.1 micrond Invocation

The `micrond` daemon takes no command line arguments. Its behavior is controlled by command line options.

Unless instructed otherwise, upon startup `micrond` disconnects from the controlling terminal and remains in the background. During normal execution it logs its diagnostic messages (if any) via `syslog` facility '`cron`'. Eventual output and error diagnostics from the invoked cron jobs are mailed to the job owner, as described in Section 2.3.1 [Mailing the cronjob output], page 8.

The following options modify the `micrond` behavior:

`-f` Remain in foreground.

`-h` Display a short help summary and exit.

`-g` *`group`*`=`*`dir`*
Define the directory or file name for crontab group *group*. Valid group names are: `master`, `system`, `user`, and `group`. See Section 2.1 [Crongroups], page 3.

`-g` *`group`*
`-g no`*`group`*
Enable or disable crontab group *group*. Note, that the `group` crontab group is disabled by default.

`-l` *`pri`* Log only messages with syslog priority *pri* or higher. Valid arguments are, in order of increasing priority: `debug`, `info`, `notice`, `warning`, `err`, `crit`, `alert`, and `emerg`.

`-m` *`mailer`* Set the mailer command. Default is `/usr/sbin/sendmail -oi -t`.

`-P` *`file`* Write PID of the program to *file*.

`-p` *socket* Send messages to syslog via this socket. The argument is either an absolute file name of a UNIX socket, or a host name or IPv4 address optionally followed by a colon and port number or service name.

`-S` When running in foreground (see the `-f` option), log messages from `micrond` to the syslog facility '`cron`', instead of printing them to the standard error. Not to be confused with the `-s` option, described below.

`-s` Log output from cronjobs to syslog. By default, the '`cron`' facility is used. Use the `-v syslog_facility=`*f* option to change it to facility *f*. See below, for the discussion of the `-v` option.

`-T` *N* In foreground mode, when logs go to stderr, precede each line with a timestamp in ISO 8601 format (`-T0`). *N* can be 1 to skip seconds and 2 to skip both minutes and seconds from the timestamp.

`-t` *n* Before exiting, `micrond` checks if any of the cronjobs are still running. If so, it sends them the `SIGTERM` signal and waits *n* seconds for them to terminate. The cronjobs that fail to exit within that amount of time are terminated forcibly by sending them the `SIGKILL` signal.

The default timeout is 60 seconds.

`-V` Print program version, licensing information and exit.

`-v` *name*`=`*value*

Assigns initial value for the `micrond` internal variable *name* (see [built-in variables], page 6). The variable name must be used without prefix. Comparison is case-insensitive. For example:

```
-v syslog_facility=daemon
```

is equivalent to

```
_MICRON_SYSLOG_FACILITY = daemon
```

`-W` *opt* Set internal `micrond` option. As of version 1.3.93, only one such option is implemented: `paranoid_memfree`. When `-Wparanoid_memfree` is given, `micrond` will meticulously free all allocated memory before terminating. Normally this isn't needed, as the operating system will reclaim that memory anyway. Use this option if you run `micrond` under `valgrind` or another memory leak detector, in order to avoid spurious warnings.

4 The crontab Utility

The crontab utility allows the user to list, edit or remove personal or group crontabs.

The microndcron daemon reads crontabs from several crontab groups, two of which contain crontabs for particular system users. The *user crongroup* contains per-user crontabs, and the *group crongroup* contains user crontabs editable by a group of users (see Section 2.1 [Crongroups], page 3).

By default, crontab operates on per-user crontabs. To edit group crontabs, the -g option must be specified.

When run without arguments, crontab enters *copy mode*, in which it copies the content of the supplied file to the user personal crontab, overwriting its prior content. For example

```
crontab newfile
```

overwrites the current crontab schedule of the user with the content of the file newfile.

To list the content of the crontab, use the -l option. It will be displayed on standard output.

To edit it, run crontab -e. A temporary copy of the crontab will be created and loaded to the editor specified by the VISUAL environment variable. If it is unset, the EDITOR variable is consulted. If this variable is unset as well, the built-in default 'vi' will be used.

Once you quit the editor, the edited crontab will be atomically moved to your personal crontab and re-read by micrond.

The -r option removes the crontab. Use it with caution. No backup copies are preserved.

When using destructive operations (such as copying or removal), it is safer to use the -i option which instructs the program to ask for the user consent before undertaking the modification.

The super-user can address the crontab of a particular user by supplying the user login name with the -u option. The use of this option is restricted for super-user, except if used together with the -g option.

User crontab groups contain multiple files for each system user. They are useful for certain pseudo-accounts. For example, a site running multiple web services may need to install separate crontabs for each of them and to allow users who run these services to edit their crontabs. This is done using the -g option. The name of the account for which the crontab is edited is supplied with the -u option. For example, to edit a crontab 'portal' in account 'www-data', one would use:

```
crontab -g -u www-data -e portal
```

The use of group crontabs for account *x* is allowed only for users who are members of the primary group of *x*.

4.1 crontab Invocation

The crontab usage is:

crontab [-u *name*] *file*

Copy mode. Replace the crontab of the user who invoked the command with the content of *file*.

The `-u` is allowed only for super-user. With this option, the crontab of the user *name* is replaced.

`crontab [-eilr] [-u name]`
: List (`-l`), edit (`-e`) or remove (`-r`) the user's crontab.

The `-u` is allowed only for super-user. With this option, the command operates on the crontab of the user *name*.

`crontab -g [-u name] [-eilr] file`
: List (`-l`), edit (`-e`) or remove (`-r`) the crontab *file* in the crongroup of the current user (or user *name*). (see Section 5.1 [User Group Crontabs], page 15)

The `-u` option is allowed for super-user and any user whose primary group matches that of the user *name*.

The following table summarizes command line options of the `crontab` utility:

`-e` Edit crontab.

`-g` Operate on user cron group files.

`-h` Print a short command line usage summary and exit.

`-i` Interactively ask before removing or replacing.

`-l` List crontab content. When used with the `-g` option, this option lists the content of the crontab *file* in the group. If *file* is not supplied, displays the list of available crontabs in the group, along with their owners.

`-r` Remove the crontab.

`-u name` Operate on crontab of user *name*.

`-V` Print program version, licensing information and exit.

5 Key Features

The following sections describe features specific to micrond.

5.1 User Group Crontabs

User group crontabs are an experimental feature designed to facilitate maintenance of per-service crontabs. Consider, for example, a web server that runs multiple web sites maintained by various users who need to run periodic backend jobs on behalf of the account the httpd server runs as. User group crontabs make it possible without intervention of the system administrator. Let's assume httpd runs as the user 'apache'. The system administrator creates a directory /var/spool/cron/crongroups/apache, and sets 'apache' as its owner:

```
mkdir /var/spool/cron/crongroups/apache
chown apache: /var/spool/cron/crongroups/apache
```

Then, he adds login names of the users who are allowed to edit apache cronjobs to the primary group of the 'apache' user. Once done, these users become able to create and edit crontabs in this directory using the crontab command with the -g option (short for 'group'). For example, the command

```
crontab -u apache -g -e myproject
```

edits the file myproject in this directory.

User group crontabs are disabled by default. To enable them, run micrond with the -g group option.

5.2 Long Crontab Lines

Very long crontab lines can be split across several physical lines using the familiar backslash continuation technique: a backslash appearing immediately before the ending newline character is removed along with the newline and the content of the next line is appended in its place. Multiple line continuations are allowed, as long as total line length does not exceed 1024 characters.

5.3 Built-in Variables

A number of built-in variables control the interpretation of crontab entries and execution of commands. Each built-in variable has two name variants: setting the name prefixed with '_JOB_' affects only the cron job definition that immediately follows (with optional variable assignments in between), whereas setting the name prefixed with '_MICRON_' affects all commands that follow, until another assignment of the same variable (or its '_JOB_' counterpart) is encountered or the end of file is reached. For example, the following fragment instructs micrond to log all output produced by the command run-periodic to syslog facility 'daemon' using the tag 'hourly'. These two settings affect only this particular command:

```
_JOB_SYSLOG_FACILITY = daemon
_JOB_SYSLOG_TAG = hourly

15 * * * *  root  run-periodic
```

Built-in variables are described in detail in [built-in variables], page 6.

5.4 Day Field Semantics

In a crontab schedule, the day of a command's execution can be specified by two fields: day of month (field 3), and day of week (field 5). If both fields are restricted (i.e. are not '*'), their interpretation differs between various `cron` implementations. *Vixie* cron will run the command when either field matches the current time (the fields are thus joined by a logical OR). *Dillon's* cron interprets the 3rd field as an ordinal number of weekday in month (so that allowed numeric values of the 3rd field in this case are 1-5). Consider for example the following schedule:

```
0 11 1,4 * 1-3
```

For *Vixie* cron, this means "run the command on each 1st and 4th day of the month as well as on each Monday, Tuesday and Wednesday". The meaning of this schedule for *Dillon's* cron is: "run the command on each first and fourth Monday, Tuesday and Wednesday in the month".

The semantics used by `micron` is configurable. By default it assumes the two fields to be joined by a logical AND, i.e. the example above would mean "each first and fourth day of the month *if* the day of week is Monday, Tuesday or Wednesday". The use of *Vixie* or *Dillon* semantics can be requested by setting the `_MICRON_DAY_SEMANTICS` variable in the crontab. For example, the line:

```
_MICRON_DAY_SEMANTICS = Vixie
```

requests the semantics used by *Vixie* cron.

See [_MICRON_DAY_SEMANTICS], page 8, for a detailed description of this variable.

5.5 Variable Assignments in Crontabs

Variable assignments can appear anyplace in a crontab. The modified environment remains in effect for all subsequent commands until changed by another assignment or the end of file is reached, whichever happens first. For example, the output of the following two example entries is mailed to two different users:

```
MAILTO=one
* * * * * command one
MAILTO=two
* * * * * command two
```

5.6 Job Output Report

Output of a crontab job can be either mailed to its owner (a traditional behavior), or reported via `syslog` to arbitrary facility, or appended to a disk file.

The default behavior is to send the output to the user on whose behalf the cronjob was run. This in discussed in Section 2.3.1 [Mailing the cronjob output], page 8.

When syslog is requested, each line of the cronjob output is sent to the syslog in real time. This setting can be enabled either globally by the use `-s` command line option, individually in a crontab (using the `_MICRON_SYSLOG_FACILITY` variable), or individually for a cronjob using the `_JOB_SYSLOG_FACILITY` variable.

Syslog tag can be supplied using the _JOB_SYSLOG_TAG variable (see [_JOB_SYSLOG_TAG], page 7). In its absence, syslog tag is constructed from the location of the job in the crontab file and the first word of the command.

For a detailed discussion of syslog redirection, See Section 2.3.2 [Logging output to syslog], page 9.

Finally, the output can be captured and appended to a given disk file upon termination of the cronjob. This can be useful, e.g. when running microndin a Docker container. See Section 2.3.2 [Logging output to syslog], page 9.

5.7 Simultaneous Job Execution

Micrond controls the number of simultaneously running instances of a cron job. It is set by the _MICRON_MAXINSTANCES variable. The default value is 1, which means that the job won't be started until its previous instance terminates.

See [_MICRON_MAXINSTANCES], page 7.

5.8 Detection of Crontab Modifications

On GNU/Linux systems, micrond uses inotify (http://man.gnu.org.ua/manpage/?7+inotify) to track crontab modifications, which means that any change to a crontab is noticed as soon as the crontab file is saved.

On other systems, micrond relies on checking the crontab modification times each minute, which is less effective.

The use of the 'kqueue' interface on *BSD systems is planned in future versions.

6 Downloads and Installation

The program can be downloaded from `https://download.gnu.org.ua/release/micron`. Before installation, create a group which will be used as owner for the user and user group crontab directories. The `crontab` binary will be installed as set-GID to that group. By default, the group is named '`crontab`'. Assuming this, the usual build sequence is

```
./configure
make
make install
```

If you chose another group name, supply it to `configure` using the `--with-crontab-gid` option.

The above commands will install the package under `/usr/local`. That is, the server will be installed as `/usr/local/sbin/micron`, the `crontab` utility as `/usr/local/bin/crontab`, etc. If that's not what you want, use the `--prefix` option to specify the installation prefix, e.g.

```
./configure --prefix=/usr
```

Please refer to the `INSTALL` document in the source directory for a discussion of available options to configure and their effect.

7 Bug Reports

If you think you found a bug in `micron` or in its documentation, please send a mail to `gray@gnu.org` (Sergey Poznyakoff) or use the bug tracker at `https://puszcza.gnu.org.ua/bugs/?group=micron` (requires authorization).

Appendix A GNU Free Documentation License

Version 1.3, 3 November 2008

0. PREAMBLE

   The purpose of this License is to make a manual, textbook, or other functional and useful document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

   This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

   We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

   This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

   A “Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

   A “Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

   The “Invariant Sections” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not “Transparent” is called “Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

The “publisher” means any person or entity that distributes copies of the Document to the public.

A section “Entitled XYZ” means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as “Acknowledgements”, “Dedications”, “Endorsements”, or “History”.) To “Preserve the Title” of such a section when you modify the Document means that it remains a section “Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

   You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

   The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

   In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

   You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

   You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

   A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

   If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

   Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

   If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

   You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

   However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

   Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

   Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

    The Free Software Foundation may publish new, revised versions of the GNU Free Documentation 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. See `https://www.gnu.org/licenses/`.

    Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

    "Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

    "CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

    "Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

    An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

    The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
  Copyright (C)  year  your name.
  Permission is granted to copy, distribute and/or modify this document
  under the terms of the GNU Free Documentation License, Version 1.3
  or any later version published by the Free Software Foundation;
  with no Invariant Sections, no Front-Cover Texts, and no Back-Cover
  Texts.  A copy of the license is included in the section entitled ``GNU
  Free Documentation License''.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with. . . Texts." line with this:

```
    with the Invariant Sections being list their titles, with
    the Front-Cover Texts being list, and with the Back-Cover Texts
    being list.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Index