

LOGWATCH NOTES

(Adapted From The Logwatch Documentation)
Mark E. Donaldson

LOGWATCH MAN PAGE

NAME

logwatch - system log analyzer and reporter

SYNOPSIS

logwatch [--detail *level*] [--logfile *log-file-group*] [--service *ser-vice-name*] [--print] [--mailto *address*] [--archives] [--range *range*] [--debug *level*] [--save *file-name*] [--logdir *directory*] [--hostname *hostname*] [--splithosts] [--multiemail] [--output *output-type*] [--numeric] [--version] [-help|--usage]

DESCRIPTION

LogWatch is a customizable, pluggable log-monitoring system. It will go through your logs for a given period of time and make a report in the areas that you wish with the detail that you wish. Logwatch is being used for Linux and many types of UNIX.

OPTIONS

--detail *level*

This is the detail level of the report. *level* can be a positive integer, or high, med, low, which correspond to the integers 10, 5, and 0, respectively.

--logfile *log-file-group*

This will force LogWatch to process only the set of logfiles defined by *log-file-group* (i.e. messages, xferlog, ...). Log-Watch will therefore process all services that use those log-files. This option can be specified more than once to specify multiple logfile-groups.

--service *service-name*

This will force LogWatch to process only the service specified in *service-name* (i.e. login, pam, identd, ...). LogWatch will therefore also process any log-file-groups necessary to process these Services. This option can be specified more than once to specify multiple services to process. A useful *service-name* is *All* which will process all services (and logfile-groups) for which you have filters installed.

--print

Print the results to stdout (i.e. the screen).

--mailto *address*

Mail the results to the email address or user specified in *address*.

--range *range*

You can specify a date-range to process. Common ranges are *Yesterday*, *Today*, *All*, and *Help*. Additional options are listed when invoked with the *Help* parameter.

--archives

Each log-file-group has basic logfiles (i.e. /var/log/messages) as well as archives (i.e. /var/log/messages.? or /var/log/messages.?.gz). When used with "--range all", this option will make

LOGWATCH NOTES

(Adapted From The Logwatch Documentation)
Mark E. Donaldson

LogWatch search through the archives in addition to the regular logfiles. For other values of `--range`, LogWatch will search the appropriate archived logs.

--debug level

For debugging purposes, *level* can range from 0 to 100. This will *really* clutter up your output. You probably don't want to use this.

--save file-name

Save the output to *file-name* instead of displaying or mailing it.

--logdir directory

Look in *directory* for log files instead of the default directory.

--hostname hostname

Use *hostname* for the reports instead of this system's hostname. In addition, if `HostLimit` is set in the `logwatch.conf` configuration file (see **MORE INFORMATION**, below), then only logs from this hostname will be processed (where appropriate).

--numeric

Inhibits additional name lookups, displaying IP addresses numerically.

--usage

Displays usage information

--help same as `--usage`.

FILES

`/usr/share/logwatch/`

This directory contains all the perl executables and configuration files shipped with the logwatch distribution.

`/etc/logwatch`

This directory contains local configuration files that override the default configuration. See **MORE INFORMATION** below for more information.

EXAMPLES

logwatch --service ftpd-xferlog --range all --detail high --print --archives

This will print out all FTP transfers that are stored in all current and archived xferlogs.

logwatch --service pam_pwdb --range yesterday --detail high --print

This will print out login information for the previous day...

INTRODUCTION

LogWatch is a system log analyzer and reporter. Usage information about LogWatch can be obtained through the man page: `man logwatch` The section titled "MORE INFORMATION" in the man page lists additional documentation files available with the distribution. A summary of the command-line switches described in the man page can be obtained with the `'--help'` option: `logwatch --help` The rest

LOGWATCH NOTES

(Adapted From The Logwatch Documentation)
Mark E. Donaldson

of this document is intended for those that wish to customize or enhance LogWatch beyond the capabilities provided with the command-line switches.

DIRECTORY STRUCTURE

This section describes the subdirectories and files shipped with the LogWatch distribution, using the names and locations used by default.

The directory **/usr/share/logwatch** contains both the configuration and (perl) executable files. The contents of this directory are the following subdirectories:

- **default.conf:** Contains the default configuration files shipped with the LogWatch distribution.
- **dist.conf:** Contains the configuration files shipped with your specific Operating Systems distribution.
- **lib:** Contains perl library files.
- **scripts:** Contains the perl executables.

The **/etc/logwatch** directory contains the following subdirectories:

- **conf:** Contains the configuration files specific to the system.
- **scripts:** Contains the executable scripts specific to the system.

CONFIGURATION STRUCTURE

The contents of the three directories **/usr/share/logwatch/default.conf**, **/usr/share/logwatch/dist.conf**, and **/etc/logwatch/conf**, all have the same structure:

- **services:** This subdirectory contains the configuration files specific to each service. LogWatch determines which services are available by examining the contents of this directory. Each service configuration file is named by its service name with the ".conf" suffix.
- **logfiles:** This subdirectory contains the logfile group configuration files. Each logfile group configuration file contains information about one or more log files with the same format. Several services may use the same logfile group configuration file. Each of these configuration files are named by the group name with the ".conf" suffix. Many of the group names are taken from the name of a system log file (such as messages, maillog, secure, etc.), but not always.
- **logwatch.conf:** This file contains the defaults for the overall execution of LogWatch, and affect all of its services. Many of its parameters can be overridden by command-line switches when invoking the LogWatch executable, as described in the man page for LogWatch.
- **ignore.conf:** This file specifies regular expressions that, when matched by a log entry, will be ignored, regardless of which service is being executed.

The **/etc/logwatch/conf** directory may also contain the file 'override.conf',

EXECUTABLE STRUCTURE

LOGWATCH NOTES

(Adapted From The Logwatch Documentation)
Mark E. Donaldson

The contents of the two directories `/usr/share/logwatch/scripts` and `/etc/logwatch/scripts` have the same structure:

- **services:** This subdirectory contains the executable for each service. Unless otherwise specified in the configuration service file, the executables are written in the perl language.
- **shared:** This subdirectory contains executables that may be invoked by more than one configuration service file.
- **logfiles:** This subdirectory may contain subdirectories with logfile group names. The executables under each of these subdirectories are automatically invoked when running a service that uses the corresponding logfile group name.

CUSTOMIZING THE CONFIGURATION

LogWatch can be, and has been, used on many variants of the Linux and UNIX systems. Some distributions that include LogWatch modify the default configuration to comply with the settings of said distributions. Therefore, most people will not need to make any modifications to LogWatch.

However, LogWatch, starting with version 7.0, implements a mechanism to allow modifying the local system easier. These modifications may be needed either because the configuration of the service that writes to the system log has been altered from its default, or because the LogWatch user prefers what is reported or how it is reported by LogWatch to be different. You can customize the output of logwatch by modifying variables in the `/etc/logwatch/conf` directory.

Default values are specified in the `/usr/share/logwatch/default.conf` directory. Your distribution may have set additional defaults in the `/usr/share/logwatch/dist.conf` directory. All the variables available are declared in the files under these directories. You can change the default values to modify how or what is displayed with logwatch.

There are two mechanisms for customizing the variables:

1. The `/etc/logwatch/conf` directory is first searched for files with the same name and relative location as the `/usr/share/logwatch/default.conf` directory. Variables declared in these files override the defaults.

For example, if file `/etc/logwatch/conf/services/sendmail.conf` has the single entry: `$sendmail_unknownusersthreshold = 5` then the threshold for unknown users is set to five instead of the default of one. All other parameters are not modified.

The configuration files have four different types of declarations, determined by the first character in each line:

- `#`: Rest of line is a comment, and is ignored.
- `$`: Rest of first field is a variable
- `*`: Denotes the name of an executable script

Other than blank lines, the only other declarations are reserved variable names, such as `LogFile`, `Archive`, etc.

LOGWATCH NOTES

(Adapted From The Logwatch Documentation)
Mark E. Donaldson

In general, setting a variable overrides any value previously set. However, the following variables are cumulative:

- In `logwatch.conf`: `LogFile`, `Service`
- In `services/service_name.conf`: `LogFile`
- In `services/service_name.conf`: `LogFile`, `Archive`

To remove all previous declarations of that variable, set the variable to the empty string. Duplicate values in the cumulative variables are deleted. If an executable script is declared in an `/etc/logwatch/conf` file, all of the executable script declarations in the corresponding file in `/usr/share/logwatch/conf` are ignored.

Because of the way variables and executable scripts are declared, the files in `/etc/logwatch/conf/` can be created in one of two ways:

- you can create a file with only the modified variables (and new executable script declarations, if needed), as described above.
- you can copy an entire configuration file from `/usr/share/logwatch/conf` to its corresponding location in `/etc/logwatch/conf`, and then modify those lines that require it. Because duplicates are removed from cumulative variables, and new executable script groups override the old ones, the output should be correct.

2. The `/etc/logwatch/conf/override.conf` file is then searched. The first field in each line may be one of the following:

- `#` This character indicates that the rest of the line is a comment, and is ignored.
- `logwatch`: This string indicates that the rest of the line is a global configuration option, and uses the same syntax as the `/usr/share/logwatch/default.conf/logwatch.conf` file.
- `services/service_name`: (Where `service_name` is the name of a service.) This string indicates that the rest of the line is a configuration option for the specified service, and uses the same syntax as the `/usr/share/logwatch/default.conf/services` files.
- `logfiles/service_name`: (Where `service_name` is the name of a service.) This string indicates that the rest of the line is a configuration option for the specified service, and uses the same syntax as the `/usr/share/logwatch/default.conf/logfiles` files.

For example, if the file `/etc/logwatch/conf/override.conf` has the single entry:

`logwatch: Detail = High` then the default detail level for all services will be set to High.

And, in file `override.conf`, the following declaration:

`logfiles/messages: LogFile = syslog` will analyze the `syslog` file (in addition to the default `messages` file) for certain services.

LOGWATCH NOTES

(Adapted From The Logwatch Documentation)
Mark E. Donaldson

But the following two declarations combined: `logfiles/messages:LogFile = logfiles/messages:LogFile = syslog` will cause the messages file to be ignored for those same services, and only the syslog file will be used.

Customizing the Scripts

Similarly to the way you can customize the configuration, as specified in section 5, you can override the default executable scripts. This is accomplished by placing an executable file with the same name and relative path (with respect to `/usr/share/logwatch/scripts`) under the `/etc/logwatch/scripts` directory. If such a file is found in the `/etc/logwatch/scripts` directory, the corresponding file under `/usr/share/logwatch/scripts` will be ignored.

CREATING NEW SERVICE FILTERS

New services may be created by creating new configuration and executable files, described above, and placing them in the `/etc/logwatch` directory. This section provides additional details and examples for creating new service filters, but it might be easier to base the new files on the existing configuration and script files under the `/usr/share/logwatch` directory.

1. Logfile Groups

There is only one required line in the logfile group config file. This command is called 'LogFile'.

```
# This will be the logfile named 'messages' in the default logfile
# directory (probably /var/log).
LogFile = messages
```

```
# You can also give this command with an absolute path, like this:
LogFile = /var/log/messages
```

You can have as many LogFile entries as you wish. All the files specified will be merged into one input stream for any filters that use this logfile group. You can also use standard wildcards when you specify the filename.

Another command that is optional is called 'Archive'. You can specify a file to also include in the data stream if the '--archives' option is used. If these files do not exist it is okay. For example:

```
# These 2 'Archive' entries will allow users of most Red Hat Linux
# systems to access their archives of the 'messages' logfile:
Archive = messages.?
# If they configure Compression to be on in /etc/logrotate.conf:
Archive = messages.?.gz
# It is best just to include both of these so that the logfile group
# will work for most systems.
```

Now, the general theory is that the LogFile Group should apply the date range requested. If the logfile is in the standard syslog format, you can use the shared script 'ApplyStdDate' to filter out only the appropriate log entries. The way to call shared scripts (located under `/usr/share/logwatch/scripts/shared`) is:

```
*ApplyStdDate =
```

LOGWATCH NOTES

(Adapted From The Logwatch Documentation)
Mark E. Donaldson

Anything following the equal sign will be passed to the program as arguments (the equal sign can be eliminated if no arguments are needed). You should look at the current logfile group config files for examples.

Finally, if the directory `/usr/share/logwatch/scripts/logfiles//` exists, any scripts in that directory will be executed. All of these scripts take the contents of all the specified logfiles in through STDIN and output the modified logfile through STDOUT. B.

2. Service Filter Configuration File

Once you have defined one or more logfile groups (or decided on one or more existing logfile groups), you need to define your service filter. This file needs to be in `/etc/logwatch/conf/services/` and it needs to be named `service_name.conf`, where `service_name` is the name of the service. You should probably copy an existing config for another service to create a new one.

There is only one required line. This is the command 'LogFile'. The LogFile command allows you to specify one or more *LogFile Groups* (as described above) that this filter will process. Remember, any filter can process any number of LogFile Groups, and any LogFile Group may contain the data from any number of logfiles (and archives).

For a service filter that needs messages from `/var/log/messages` you would add this line:

```
LogFile = messages
```

NOTE: This is *not* because the name of the logfile is 'messages', but it is because the name of the LogFile Group that has been defined is 'messages'.

You can have commands in the form of:

```
*SharedScriptName = Arguments
```

that will execute a script found in the `/usr/share/logwatch/scripts/shared/` directory named 'SharedScriptName' with arguments 'Arguments'. This filter will modify the input to the service's filter.

You can also have commands in the form:

```
$EnvironmentVariable = Value
```

This command will set the 'EnvironmentVariable' environment variable to the value 'Value'. This environment variable will be accessible by your filter program.

You will also usually want to specify a title for your script (new in Logwatch 4.0). If specified, then a start and stop delimiter will be added by Logwatch for your specific service (with your script's output between those delimiters). This will *only* happen if you produce output. If you produce no output, the headers will not be created. Here is how you define your title:

```
Title = "My Service Title"
```

LOGWATCH NOTES

(Adapted From The Logwatch Documentation)
Mark E. Donaldson

3. Service Filter Executable

Once everything above has been done, you are ready to actually write your filter. This can be done in any language as all it does is:

- Read logfile entries from STDIN
- Access some environment variables
- Generate a report on STDOUT

Before you try to write a filter, you should create the filter and make its contents the test script given below. The filter needs to be located in `/etc/logwatch/scripts/services/` and named `service_name` (because you named the config file `service_name.conf`).

```
##### Cut Here #####
#!/bin/bash
# This is a nice script that will show you the lines you will
# be processing and reporting on. It will first display the
# standard environment variables and then it takes STDIN and
# dump it right back out to STDOUT.
# These are the standard environment variables. You can define
# more in your service config file (see above). echo "Date Range: $LOGWATCH_DATE_RANGE"

echo "Detail Level: $LOGWATCH_DETAIL_LEVEL"
echo "Temp Dir: $LOGWATCH_TEMP_DIR"
echo "Debug Level: $LOGWATCH_DEBUG"

# Now take STDIN and dump it to STDOUT
cat
##### Cut Here #####
```

If you temporarily replace a script such as 'pam' with the above, you will notice that much has been cut out of `/var/log/messages` before it gets to this filter. The value of the environment variable `LOGWATCH_DETAIL_LEVEL` can be any integer. In reality, it is usually 0 (for low), 5 (for medium), and 10 (for high).

Your script should only produce output as appropriate. If there are no relevant log entries, no output should be produced. Likewise, if you are reporting two things, such as "Good Logins" and "Bad Logins", you should only produce even the headers when appropriate. For example:

Bad Logins:
 amber (2 time(s))
 kirk (3 time(s))

Good Logins:
 amber (5 time(s))
 kirk (10 time(s))

But, if no failed logins occur, you should only output:

LOGWATCH NOTES

(Adapted From The Logwatch Documentation)
Mark E. Donaldson

Good Logins:
 amber (5 time(s))
 kirk (10 time(s))

Note that there is no "Bad Logins:" header as there were no bad logins. You should also use the detail environment variable when deciding what to output. Bad logins might always be displayed, but good logins might only be displayed at higher detail levels. Here is a guide on how you should use the detail setting:

- 0 (Low): Display only errors and security-related issues
- 5 (Med): Display anything that a typical administrator would be interested in
- 10 (High): Display anything that a paranoid administrator would want to see

In some cases, you can use a security setting higher than 10. This would be reserved for information so trivial that it would not even interest the US Government.