How to Guide:
Using Logwatch To Monitor dmeventd

Author: Brian Wood

Revision 0.71
02/25/08

Disclaimers

Intel Open Source License

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INTEL OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Table of Contents

Overview

From the man page for Logwatch:

> *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.*

The Logwatch utility is installed by default on Fedora & RedHat Enterprise Linux systems. Most other distributions provide the utility on their installation media or on their website update repositories.

Files Changed In This Document

- /etc/logwatch/conf/logwatch.conf

Files To Be Created

- /etc/logwatch/conf/services/dmeventd.conf
- /etc/logwatch/scripts/services/dmeventd
- dmeventd_cronjob.txt

*(***Note**: Samples of the files modified and created in this document can be found in the package "Logwatch_Dmeventd_Setup_Files.tgz".)*

Setup

1. Verify Logwatch Installation

Verify that the Logwatch utility is installed on your system.
(***Note:** On Fedora & Redhat the utility will be a symbolic link located in /usr/sbin/
called "logwatch". The actual location of the Perl script on is /usr/share/logwatch/scripts/logwatch.pl)

2. Modify logwatch.conf Configuration File

Add the following lines to the file: *etc/logwatch/conf/logwatch.conf*

```
MailTo = guest@anonymous.com
mailer = "/usr/sbin/sendmail -t"
LogDir = /var/log
MailFrom = Logwatch
Range = Today
Detail = Med
```

Explanation of variables:

MailTo	Change this to the person you would like administrative log reports emailed to.
mailer	Set this to the path of the program used to distribute email. (For Fedora & RedHat Enterprise Linux this is "usr/sbin/sendmail -t".)
LogDir	Set this to the base path of where your log files reside (For Fedora & RedHat Enterprise Linux this is /var/log.)
MailFrom	Set this to whom you would like the "From:" field filled in by. (In this example we use the default of "Logwatch".)
Range	Set this value to what you would like parsed in the logs by date: Yesterday, Today, All, etc... *To see all of the available options run the logwatch command with --range help* (For this example we use the "Today" option.)
Detail	This is for the level of detail Logwatch will put into its report. *According to the man page most system administrators would use the "Med" option.* (We use the "Med" option in this example.)

3. Create dmeventd.conf Configuration File

Create the following file: *dmeventd.conf*
and store it into the directory: */etc/logwatch/conf/services/*

Add the following lines to this newly created file:

```
Title = "Dmeventd Errors"
LogFile = messages
*OnlyService = dmeventd
```

Explanation of variables:

Title	This is where we set the title for our section within the Logwatch email. *Setting a distinctive name is helpful if you process more than one service in a given run (i.e. httpd, sendmail, cron, sshd, etc...).* (Here we are using the title of "Dmeventd Errors".)
LogFile	This is the particular file in the syslog repository we want to process event messages from. (For this example we want to process the "messages" directory in "/var/log" which was set in the previous configuration file, logwatch.conf.)
*OnlyService	Including that parameter means that we will only check for the daemon logs matching this text string. (In this case we only want to process log entries dealing with dmeventd.)

4. Create dmeventd Perl Script

Create the following file: *dmeventd*
and store it into the directory: */etc/logwatch/scripts/services/*

Add the following lines to this newly created file
*(***Note**: A sample of this file can be found in the package "Logwatch_Dmeventd_Setup_Files.tgz")*:

```
########################################################
#
# dmeventd
########################################################
#

########################################################
#
# This was written and is maintained by:
#    Brian Wood <brian.j.wood@intel.com>
#
# Please send all comments, suggestions, bug reports,
#    etc, to <brian.j.wood@intel.com>.
########################################################
#

# Set the location of the folder to store last time stamp
# (This is used to record the last log sent out so repeats
#  are not mailed in error.)
$Storage_file = "/etc/logwatch/scripts/services/timestamp.txt";
$count = 0;
$Detail = $ENV{'LOGWATCH_DETAIL_LEVEL'} || 0;

if (-e $Storage_file) {
        open(FD, "+<", "$Storage_file") or die $!;
        seek(FD, 0, 0);
        read(FD, $prev_time, 8);
}
else {
        open(FD, ">", "$Storage_file") or die $!;
        $prev_time = "";
}

while (defined($ThisLine = <STDIN>)) {
        #SAMPLE LOG DATA: Oct 15 01:14:33 dmraid-devhost dmeventd[24857]:
Processing device "isw_febiihjha_Volume0" for events
        # All of the elements of the 'split()' aren't used, but could be if custom formating is
desired.
        ($month, $day, $time, $hostname, $program, $message) = split(' ',$ThisLine, 6);
        chop($program); # Chop off the colon
        if ($prev_time eq "" || $time gt $prev_time) { # If this is the first run or the time
is newer than that stored print log entry
                #print "$ThisLine";
                if($ThisLine =~ /Processing Raid|End of|Monitoring device|No longer/ ) {
                        $entries{$count} = "$month $day $time:  $message";
                }
                else {
                        $entries{$count} = "$month $day $time:    $message";
                }
                $count++; #Keep a count of the number of new logs
        }
}
```

```
if ($count != 0) {
        print ("There were a total of $count new log entries\n\n");
        print ("Date            Message\n");
        print ("------------------------\n");
        $num = 0;
        while ($num < $count) {
                print ("$entries{$num}");
                $num++;
        }
}

seek(FD, 0, 0);
printf FD $time;
close(FD);
exit(0);

# vi: shiftwidth=3 tabstop=3 syntax=perl et
```

The full details of this file are beyond the scope of this document, but in essence:

- Logwatch passes as input all the lines from the syslog file containing the keyword "dmeventd"
- It checks these lines against the stored time stamp of the last log entry reported and stores the new entries
- Finally sending the new entries out in an email alert to the specified configuration file recipient

If the entries are found to have already been processed in an earlier run of Logwatch no email will be sent, this cuts down on unnecessary email traffic to the network administrator.

***Note:** The only value that needs to be set before use is the variable *$Storage_file*.

So, on the line:
$Storage_file = "/home/user1/timestamp.txt";
change the string to a location & filename that you would like to store the most recently processed log entry time stamp.

5. Configure Cron Job

Finally, configure a "cron" job to automatically run the Logwatch utility at specific intervals:

I. Create a file called "*dmeventd_cronjob.txt*" (this can be anywhere; so long as you remember its location).
II. Inside this file store the following text
*(***Note**: A sample of this file can be found in the package "Logwatch_Dmeventd_Setup_Files.tgz")*:

```
# Runs the logwatch script looking for dmeventd syslog messages.
# The setup will run on the following interval:
#        field          allowed values
#      -----         --------------
#      minute          0-59
#      hour            0-23
#      day of month  1-31
#      month           1-12 (or names, see below)
#      day of week   0-7 (0 or 7 is Sun, or use names)
#
# Some samples:
# */5 * * * * /usr/sbin/logwatch --service dmeventd
# This will run every five minutes, every hour, every day of the month,
# every month of the year, every day of the week.
#
# 0-59 * * * * /usr/sbin/logwatch --service dmeventd
# This will run every minute, every hour, every day of the month,
# every month of the year, every day of the week.
# (Note: this call also be accomplished with a */1 instead of 0-59)

 0-59 * * * * /usr/sbin/logwatch --service dmeventd
```

***Note:** Make sure there is a new line at the end of the file before saving. If you perform the next step with a crontab file that does not include an empty new line at the end you will get the following error:*

"dmeventd_cronjob.txt":21: premature EOF
errors in crontab file, can't install.

III. From the command-line run the following command as the "root" user:

[root@dmraid-devhost test]# **crontab dmeventd_cronjob.txt**

(This loads the new cronjob into /var/spool/cron/root on a Fedora/RedHat Enterprise Linux system.)

IV. To verify this new cron job is loaded and is set to run execute the following command:

```
[root@dmraid-devhost test]# crontab -l
# Runs the logwatch script looking for dmeventd syslog
messages.
# The setup will run on the following interval:
#       field          allowed values
#      -----         --------------
#      minute          0-59
#      hour            0-23
#      day of month  1-31
#      month           1-12 (or names, see below)
#      day of week   0-7 (0 or 7 is Sun, or use names)
#
# Some samples:
# */5 * * * * /usr/sbin/logwatch --service dmeventd
# This will run every five minutes, every hour, every day
of the month,
# every month of the year, every day of the week.
#
# 0-59 * * * * /usr/sbin/logwatch --service dmeventd
# This will run every minute, every hour, every day of the
month,
# every month of the year, every day of the week.
# (Note: this call also be accomplished with a */1 instead
of 0-59)

 0-59 * * * * /usr/sbin/logwatch --service dmeventd
```

As you can see in this example our logwatch job is now loaded and set to run every minute.

6. Cycle Time

That's it, if a new dmeventd event is stored into the syslog file Logwatch will catch it the next time it runs (which by this setup will be the next minute clock cycle).

7. Sample Output

Sample output that will be generated & mailed using this setup to the user specified in the *logwatch.conf* file:

```
################### Logwatch 7.3.6 (05/19/07)
####################
      Processing Initiated: Tue Jan  1 12:32:01 2008
      Date Range Processed: today
                          ( 2008-Jan-01 )
                          Period is day.
    Detail Level of Output: 5
          Type of Output: unformatted
        Logfiles for Host: testhost.blah.com
 ##################################################################
############

 ---------- Dmeventd Errors Begin ----------

 There were a total of 7 new log entries

 Date          Message
 -----------------------
 Jan 1 12:31:36:  Processing Raid Volume "isw_beggfdcfcf_raid00" for Events
 Jan 1 12:31:36:    Stripe device, 8:48 (/dev/sdd) has reported an I/O error.
 Jan 1 12:31:36:    The kernel has recorded 5 event(s) against this device.
 Jan 1 12:31:36:    Associated Userspace Names: /dev/sdd=Disabled /dev/sde=Active
 Jan 1 12:31:36:    Associated SATA Port Mapping: /dev/sdd=3 /dev/sde=4
 Jan 1 12:31:36:    Associated UUID: DMRAID-isw_beggfdcfcf_raid00
 Jan 1 12:31:36:  End of event processing for Raid Volume "isw_beggfdcfcf_raid00"

 ---------- Dmeventd Errors End ----------


 ######################## Logwatch End
#########################
```

Appendix

Notes About Example Used In This Document

- All examples in this document assume the user has basic Linux knowledge.

- The Author takes no responsibility for what happens to your system by using this guide. All instructions and testing should be on a non-production system before deploying to general service.

- As of the time of this writing both Fedora 7 and RedHat Enterprise Linux5.x were able to build all of the items described in this document; your experiences may vary in future operating system releases.

- The syslog log file needs to be rotated daily for the */etc/logwatch/scripts/services/dmeventd* Perl script to function as described in this document. Since this script only stores a simple text string representation of the last time a dmeventd log entry was caught it would gather & email old log file data if used on a syslog that contained multiple days worth of data. By default Fedora and Redhat use "Logrotate" to cycle this daily, so this will not be a problem.

- As mentioned in the previous point, the example described in this document will only work properly if either the user or cron runs the Logwatch utility at least once a day. Since this is meant to be used as part of the critical data storage hardware monitoring infrastructure it should be ran as described in this document to be most effective.

- If you would like to run this script on a weekly basis the basic logic of the */etc/logwatch/scripts/services/dmeventd* script along with the setup of the utility "Logrotate" would need to be changed in order to store/process full date strings for use in multiple log file comparisons. (*This is beyond the scope of this document.)*

Other Reading Material On Logwatch

- In the directory /usr/share/doc/logwatch-* you'll find a text file named "HOWTO-Customize-Logwatch" which goes in-depth into Logwatch customization.

- A wealth of information is also available at the maintainer's (Kirk Bauer <kirk@kaybee.org>) website: http://www.logwatch.org/
  Here you'll find the latest version along with a forum to post questions about Logwatch.