

Munin_Statistics

Similar to MRTG graphs showing Network Usage, Munin is monitoring system used to produce graphs of usage of key areas in the running of a server. Munin can produce graphs of:

- Disk Usage
- Mysql
- Network (similar to mrtg)
- Postfix Mail
- Processes
- Squid Proxy
- System - CPU, Load, Memory etc.

Updated for Debian Squeeze

Contents

- [1 Installation](#)
- [2 Temperature Monitoring with Munin](#)
- [3 SMART Monitoring with Munin](#)
 - ◆ [3.1 Potential Problems](#)
 - ◆ [3.2 Specific Problem with /etc/munin/plugins/postfix_mailstats on Ubuntu Feisty and Debian Etch](#)
 - ◆ [3.3 Problem with apache accesses and munin-node](#)
- [4 Disk Usage \(DU\) Monitoring of home and mail Directories plugin for Munin](#)
- [5 Munin Monitoring of Xen CPU and Network Traffic](#)
- [6 Email Notifications from Munin](#)
 - ◆ [6.1 Summary of Checks](#)
 - ◆ [6.2 Subject Title of Email](#)
- [7 Munin via CGI \(Reduce CPU Load\)](#)
- [8 Monitor Memory Usage of Processes](#)
- [9 Extra Links](#)
- [10 Update: munin-node on Lenny](#)
- [11 Split up Munin multi-graphs to separate graphs](#)
 - ◆ [11.1 Before](#)
 - ◆ [11.2 After](#)
- [12 Apache Mod Watch & Munin Graphs showing vhost accesses](#)
 - ◆ [12.1 Compile](#)
 - ◆ [12.2 Add and config mod_watch apache module](#)
 - ◆ [12.3 Checking mod_watch for errors](#)
 - ◆ [12.4 mod_watch Errors received](#)
 - ◆ [12.5 resolve mod_watch errors](#)
 - ◆ [12.6 Munin Plugin for Mod Watch](#)
- [13 Migrate Munin RRD Graphs between x32 and x64](#)

Installation

(munin=graphing server, munin-node=server sending stats):

```
apt-get install munin munin-node
```

Munin_Statistics

It should work out of the box. Default web is located in ~~/var/www/munin~~ /var/cache/munin/www with config files in /etc/munin with a config for munin and munin-node. You can go: ln -s /var/cache/munin/www /var/www/munin and then visit <http://server/munin> to look at the graphs **or** look at /etc/munin/apache2.conf

Remember to open firewall to allow munin-node to connect to the graphing server! It uses port 4949.

Temperature Monitoring with Munin

This is with a 2.6 kernel and udev!

```
apt-get install lm-sensors
```

```
sensors-detect
```

```
// The above should edit /etc/modules and include extra lines. The next line adds the sensors in man  
modprobe -a i2c-viapro i2c-isa eeprom w83627hf
```

```
//A reboot should sort everything. (Cant remember if a reboot is required or not.)
```

Onto config of Munin. All munin templates are kept in /usr/share/munin/plugins.

```
ln -s /usr/share/munin/plugins/sensors_ /etc/munin/plugins/sensors_fan  
ln -s /usr/share/munin/plugins/sensors_ /etc/munin/plugins/sensors_temp  
ln -s /usr/share/munin/plugins/sensors_ /etc/munin/plugins/sensors_volt
```

The above lines should pick up all required - and should not need editing - once the sensor modules are included.

Test Sensors Test the sensors by simply going:

```
sensors
```

<http://www.debian-administration.org/articles/229>

<http://munin.projects.linpro.no/wiki/Documentation>

<http://www.debianhelp.co.uk/munin.htm>

<http://www.debian-administration.org/articles/327>

<http://munin.projects.linpro.no/wiki/PluginCat>

<http://www.valerie-et-guillaume.com/categorie/geek/>

SMART Monitoring with Munin

Note that smartmontools must be installed and smartd running before munin will pick up and use smart. Configure smart using the way mentioned on this wiki here: [SMART](#)

After SMART is running:

```
ln -s /usr/share/munin/plugins/smart_ /etc/munin/plugins/smart_sda
```

Installation

Munin_Statistics

```
ln -s /usr/share/munin/plugins/smart_ /etc/munin/plugins/smart_sdb

//edit the following only for Debian Etch or older
vi /etc/munin/plugin-conf.d/munin-node
#uncomment
#[smart_*]
#user root
#include
[smart_sda]
user root
env.smartargs -d ata -a

[smart_sdb]
user root
env.smartargs -d ata -a
```

That should do it. Apologies for the lack of a reference.

Potential Problems

On Ubuntu edgy, I apt-get install munin munin-node. After a couple of minutes, munin would run - however NO graphs would be generated in /var/www/munin. The template html page would be there with the localhost.localdomain, however NO info or NO graphs would appear. After looking in:

```
/var/lib/munin/
```

I saw that all the rrd files were there!!! Its just that they were not been parsed and made into PNG's etc. ALL rrd and munin packages, recommended pages were all installed.

I decided to use the munin-graph tool to see if it would look at the rrd files and whether it would throw an error relating to a version of perl etc. Although - running this as root is not recommended as it will mess up ALL permissions I needed to try it. So I ran:

```
/usr/share/munin/munin-graph --force-root
/usr/share/munin/munin-html --force-root
```

I got errors!! Something about choke etc. After running the above for another few times - it seemed to iron itself out. I reset back all the permissions, and let cron run it as munin, and THANKFULLY it worked. That solved it for me. Normally munin will run out of the box (which is great), however it needed a bit of persuasion on Ubuntu edgy.

Specific Problem with /etc/munin/plugins/postfix_mailstats on Ubuntu Feisty and Debian Etch

The error was:

```
Use of uninitialized value in pattern match (m//) at /etc/munin/plugins/postfix_mailstats line 239,
Modification of a read-only value attempted at /etc/munin/plugins/postfix_mailstats line 239, <LOGFI
2007/05/18-20:30:05 Plugin "postfix_mailstats" exited with status 2304. ----
2007/05/18-20:30:15 Server closing!
```

The known fix was:

Munin_Statistics

```
vi /etc/munin/plugins/postfix_mailstats
#on line 239 change from:
$cause = 'OTHER' unless $1 =~/^d+$/;
#to:
$cause = 'OTHER' unless $1 =~/^d+$/;
```

That should solve that little error. Restart munin-node and then see. Reference:

http://www.nabble.com/Bug-403341:-munin-node:-missing-~-in-postfix_mailstats:239-breaks-plugin-t2831828.html

The exact same problem was found on Debian Etch. The above fixed it also. Another reference to this problem can be found at:

<http://www.bluequartz.us/phpBB2/viewtopic.php?t=22694&sid=585e9869ebc13d17503c2fa307ab606a>

Problem with apache_accesses and munin-node

So I saw that apache graphs from munin were not updating. Taking a quick look in: /var/log/munin/munin-node.log showed:

```
Can't locate object method "new" via package "LWP::UserAgent" at /etc/munin/plugins/apache_processes
```

Solution:

```
apt-get install libwww-perl
```

Refs: <http://osdir.com/ml/db.rddtool.munin.general/2005-08/msg00046.html>

Disk Usage (DU) Monitoring of home and mail Directories plugin for Munin

The following file is placed in: /usr/share/munin/plugins/du

```
#!/usr/bin/perl -w
#
# Copyright (C) 2006 Philipp Gruber <pg@flupps.net>
#
# This program is free software; you can redistribute it and/or
# modify it under the terms of the GNU General Public License
# as published by the Free Software Foundation; either version 2
# of the License, or (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.
#
#
#
#
```

Munin_Statistics

```
# This munin-plugin shows the size of every users home directory in a
# munin-diagram.
#
# Because calculating the size every 5 minutes would take very long, it reads
# the size from the file /tmp/munin-homedirs.cache
#
# You need to add the following line to the crontab of root, to generate the
# file every hour. This has to be done by root, because other users probably
# don't have read access to all home directories.
#
# 0 * * * * du -B 1 --max-depth=1 /home | grep -Ev 'home$' > /tmp/munin-homedirs.cache
#
# You must also make sure, that munin is allowed to read this file.
#
#
## family=contrib

#open (HOME,"ls /home|");
open (HOME, "cat /tmp/munin-du.cache | cut -f 2|");
@users = <HOME>;
close (HOME);

if ($ARGV[0] and $ARGV[0] eq "config") {
    print "graph_title Size of selected directories\n";
    print "graph_args --base 1024 -l 1\n";
    print "graph_vlabel Bytes\n";
    foreach $user (@users) {
        $user =~ s![^a-zA-Z0-9_ \/\n]!_!g;
        $user =~ s/\n//gsm;
        print "$user.label $user\n";
    }
    exit 0;
}

open (WHO,"cat /tmp/munin-du.cache |");
@who = <WHO>;
close (WHO);

my %logins;
foreach $user (@users) {
    $user =~ s/\n//gsm;
    $logins{$user} = 0;
}

my $free = `df -P -B 1 / | tail -n1 | awk '{ print \$4 }'`;
foreach $user (@who) {
    $user =~ s![^a-zA-Z0-9_ \/\n]!_!g;
    $user =~ s/^[([0-9]+)\w(.*)/$2.value $1/g;
    print "$user";
}

# vim:syntax=perl
```

Note: The above file is modified from the original which can be obtained at:

<http://munin.projects.linpro.no/wiki/plugin-homedirs> <http://www.mega2000.de/~philipp/homedirs>

The above munin plugin reads the information from a temp file in /tmp/munin-du.cache. A cronjob running once daily refreshes this cache file. The following command is used for the cronjob:

```
#!/bin/bash
du -B 1 --max-depth=1 /home /srv /var /tmp /usr | sort -nr | grep -Ev '/home$' | grep -Ev '/var$' |
```

Munin_Statistics

#Its not very tidy, I know, but it does the trick.

Munin Monitoring of Xen CPU and Network Traffic

The following websites provide the original information:

http://munin.projects.linpro.no/attachment/wiki/PluginCat/xen_traffic_all

http://www.skullkrusher.net/linux/scripts/xen_percent (Original source:

<http://lists.xen-source.com/archives/html/xen-users/2006-01/msg00323.html>)

```
cd /usr/share/munin/plugins/
wget http://wiki.kartbuilding.net/xen_traffic_all
wget http://wiki.kartbuilding.net/xen_cpu_percent
chmod 755 xen_traffic_all xen_cpu_percent
ln -s /usr/share/munin/plugins/xen_traffic_all /etc/munin/plugins/
ln -s /usr/share/munin/plugins/xen_cpu_percent /etc/munin/plugins/
vi /etc/munin/plugin-conf.d/munin-node
#add the following:
[xen_traffic_all]
user root
[xen_percent]
user root

/etc/init.d/munin-node restart
```

That should do it.

Information obtained from:

<http://www.debian-administration.org/articles/229>

Email Notifications from Munin

Typically I use nagios to monitor services, while munin monitors individual systems, with each system having munin-node installed. Instead of having nice pretty graphs to monitor performance on an hourly, daily or weekly basis, munin can easily be set to send warning and critical email notifications.

The information on notifications using munin is quite scarce. It took quite a bit of trial and error to get everything fully corrected. Below is the most simple config. It sends one email when a service is exceeded. It sends a final email when the service is below the threshold set.

```
vi /etc/munin/munin.conf
#include the following:
contact.me.command mail -s "Munin Notification ${var:host} ${var:graph_title}" user@emailaddress.com

# a simple host tree
[server.domain.com]
address 127.0.0.1
use_node_name yes
```

Munin_Statistics

```
df._dev_sda1.warning 90
df._dev_sda1.critical 95
```

Its that simple. With a minimum of two entries all can be up and running. Now, lets explain a little:

The values/variable which will be monitored can be obtained on the client/host running munin-node using munin-run. For example, we want to monitor postfix mail usage with Munin and to send email notifications if queues are exceeded.

```
ssh root@mail
cd /etc/munin/plugins
ls
munin-run postfix_mailqueue
#If there is no postfix in the plugins folder, try:
#munin-run df
#Take the values which were returned and keep note.
exit #exit ssh on the mail host.
ssh root@muninserver
vi /etc/munin/munin.conf
[mail.domain.com]
    address 192.168.1.10
    use_node_name yes
    postfix_mailqueue.active.warning 90
```

See the following for allowing contacts for particular servers and/or for particular services:

<http://munin.projects.linpro.no/wiki/HowToContactNagios>

```
contact.nagios.command /usr/local/nagios/bin/send_nsca nagioshost.example.com -c /usr/local/nagios/e

contacts no                                # Disables warning on a system-wide basis.

[example.com;]
contacts nagios                            # Enables warning through the "nagios" contact for the group example.c

[foo.example.com]
address localhost
contacts no                                # Disables warning for all plugins on the host foo.example.com.

[example.com;bar.example.com]
address bar.example.com
df.contacts no                             # Disables warning on the df plugin only.
df.notify_alias Disk usage                 # Uses the title "Disk usage" when sending warnings through munin-limi
# Useful if the receiving end does not accept all kinds of characters
# NB: Only available in Munin-1.2.5 or with the patch described in tic
```

Summary of Checks

```
pluginname.service.warning 90
pluginname.service.critical 95
```

```
pluginname = ls /etc/munin/plugins/
servicename = munin-run pluginname
```

Subject Title of Email

```
contact.me.command mail -s "Munin Notification ${var:host} ${var:graph_title}" user@emailaddress.com
```

A list of the `${var:host}` etc. can be found at: <http://munin.projects.linpro.no/wiki/MuninAlertVariables>

Number of Alerts (max messages). I didnt bother setting this. More info can be found at:
http://munin.projects.linpro.no/wiki/contact.contact.max_messages

Munin via CGI (Reduce CPU Load)

As the cpu usage of Munin creating graphs every 5 minutes takes quite a while, munin ships with a cgi script which will render the graphs with all the data. By default the `munin-cgi-graph` (cgi) is installed with munin. To enable, do the following:

```
vi /etc/munin/munin.conf
graph_strategy cgi

apt-get install libdate-manip-perl
vi /etc/apache2/sites-enabled/monitor
ScriptAlias /cgi-bin/munin-cgi-graph /usr/lib/cgi-bin/munin-cgi-graph
/etc/init.d/apache2 reload

chgrp www-data /var/log/munin /var/log/munin/munin-graph.log
chmod g+w /var/log/munin /var/log/munin/munin-graph.log
chgrp -R www-data /var/www/munin/yourdomain
chmod -R g+w /var/www/munin/yourdomain

vi /etc/logrotate.d/munin
#under /var/log/munin/munin-graph.log { change:
#create 640 munin adm
create 660 munin www-data
```

Wait 5 minutes for munin to rebuild the html pages. Take a look at the source and you will see something like:

Another option for munin (to reduce cpu load) is to edit `/etc/cron.d/munin` and change the graphing only to per hour.

Monitor Memory Usage of Processes

I came across a useful article showing how munin can monitor and show the memory usage of a process (apache2 for example) over time. Seeing as I was doing tweaks with MySQL and apache and caching websites, I wanted to see how progress was going. Source website:

<http://somethingemporium.com/2009/06/monitoring-per-process-memory-usage-with-munin-memory-mod-php>

On the above website was listed a munin plugin called "multimemory" allowing you to monitor the memory usage of multiple processes in one nice graph.

Source: <http://github.com/dominics/munin-plugins/blob/master/multimemory>

```
cd /usr/share/munin/plugins
wget http://wiki.kartbuilding.net/multimemory
chmod 755 multimemory
ln -s /usr/share/munin/plugins/multimemory /etc/munin/plugins/

vi /etc/munin/plugin-conf.d/munin-node
#add the following:
[multimemory]
env.names apache2 named mysqld

/etc/init.d/munin-node restart
#That should do it. Wait and watch for the graphs.

ln -s /usr/share/munin/plugins/multips_memory /etc/munin/plugins/
vi /etc/munin/plugin-conf.d/munin-node
#add in the following
[multips_memory]
env.names apache2 nagios3
:wq
/etc/init.d/munin-node restart
```

Related links:

<http://github.com/dominics/munin-plugins/blob/master/multimemory>

<http://somethingemporium.com/2009/06/monitoring-per-process-memory-usage-with-munin-memory-mod-php>

<http://www.linuxjournal.com/article/10248>

<http://jazstudios.blogspot.com/2009/01/show-ram-memory-usage-linux.html>

http://elinux.org/Runtime_Memory_Measurement

Extra Links

<http://edseek.com/archives/2006/07/13/munin-alert-email-notification/>

<http://munin.projects.linpro.no/wiki/HowToContact>

<http://munin.projects.linpro.no/wiki/munin-man>

<http://munin.projects.linpro.no/wiki/munin.conf>

Update: munin-node on Lenny

So I installed munin-node on debian lenny, and it took me a few minutes to see why there were no graphs been generated. Problem: it appears that by default no plugins are enabled by default when installing munin-node. Thats unforunate!

Munin_Statistics

Anyways, google led me to: <http://articles.slicehost.com/2010/4/9/enabling-munin-node-plugin-ins-on-debian>

To enable the most common plugins go:

```
/usr/sbin/munin-node-configure --shell | sh
ls /etc/munin/plugins
/etc/init.d/munin-node restart
```

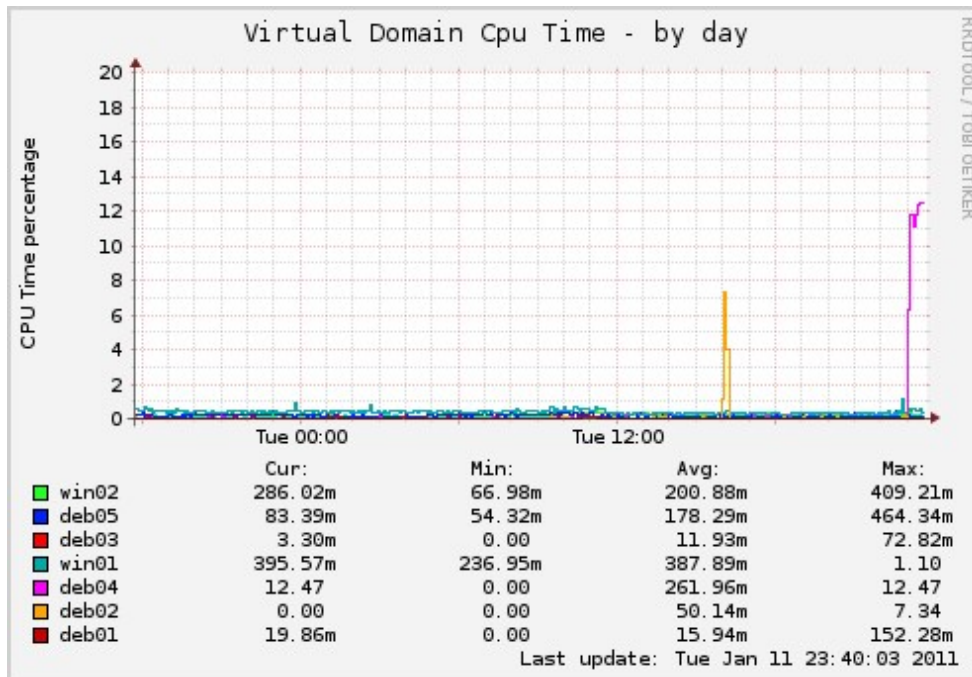
Split up Munin multi-graphs to separate graphs

I use munin to monitor my KVM VMs as outlined on

http://wiki.kartbuilding.net/index.php/KVM_Setup_on_Debian_Lenny#Munin_Plugins_for_Libvirt_.2F_KVM

This creates graphs like the following below. It shows all VMs listed together. I wanted to be able to separate out the VMs into separate munin graphs which I could then script out and make available to people running the VM. In the example below I wanted to have the VM called **deb04** have its own CPU Usage Graph.

Before



```
vi /etc/munin/munin.conf
```

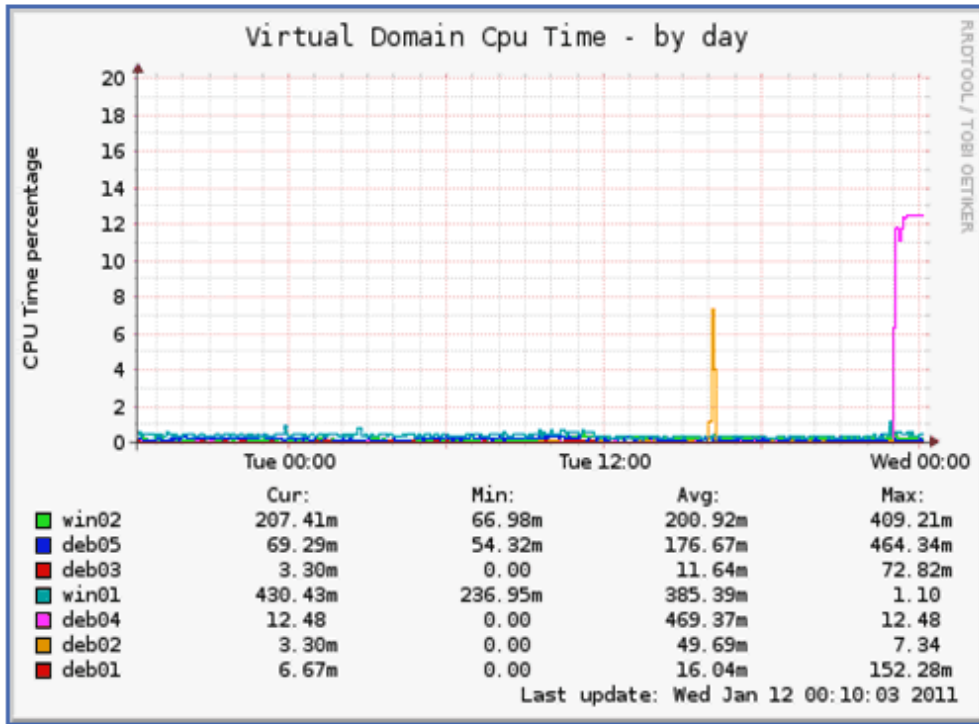
```
[server.domain.com]
```

```
address ip.address.of.server
use_node_name yes
```

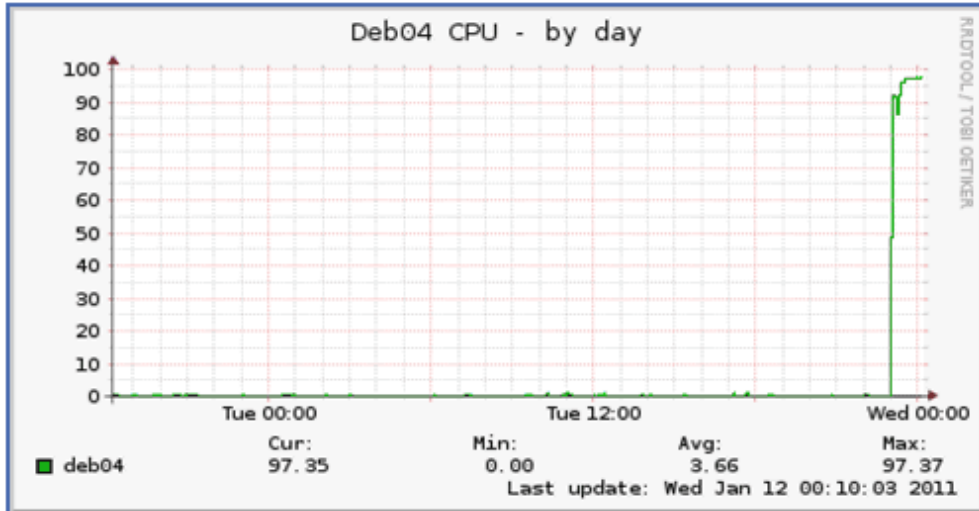
```
#libvirt_cputime.deb04_cputime.graph no //Uncomment this so the main Graph d
libvirt_cputime_deb04.graph_title Deb04 CPU
libvirt_cputime_deb04.update no
libvirt_cputime_deb04.graph_args -u 100 //Set the Upper limit to 100%
libvirt_cputime_deb04.deb04.cdef deb04,8,* //Multiplier for number of cores (Co
libvirt_cputime_deb04.graph_category Virtual Machines
libvirt_cputime_deb04.graph_order deb04=server.domain.com:libvirt_cputime.deb04_cputime
```

After

:: [Virtual Domain Cpu Time](#)



:: [Deb04 CPU](#)



Because I use cgi to render the images, I can now script the URL:

http://domain.com/cgi-bin/munin-cgi-graph/domain.com/server.domain.com/libvirt_cputime_deb04-day.png

Links:

No 1 source: <http://ingvar.blog.linpro.no/2008/04/07/todays-munin-tip-splitting-out-a-noisy-graph-from-munin/>

No 2 source: <http://munin-monitoring.org/wiki/protocol-config>

http://munin-monitoring.org/wiki/graph_args

<http://munin-monitoring.org/wiki/munin-man>

<http://munin-monitoring.org/wiki/plugin-bcp#Graphscaling>

<http://munin-monitoring.org/wiki/fieldname.cdef>

Sample <http://munin-monitoring.org/wiki/PercentGraphHowto>

Apache Mod_Watch & Munin Graphs showing vhost accesses

While this is more of an apache topic than a munin one, the main reason I wanted to compile the mod_watch module for apache is so I can graph vhost usage and access. Munin will then show graphs of how much a vhost is used. Also, while there are other how-to websites out there for this, they seem to drop off and disappear after a few years.

mod_watch itself is not supported and is actually quite difficult to get a hold of. I was using this on Debian Etch and Lenny on a x32 system. Copying the mod_watch.so and running on x64 failed so I needed to recompile on a x64 debian squeeze.

The main reference I used was: <http://stubbedtoe.co.nz/index.php/2009/09/apache-vhost-monitoring-with-munin/> This worked perfectly. There was an issue with debian squeeze which will be sorted below.

Compile

```
apt-get install apache2-dev libcrypt-ssleay-perl libwww-perl make
wget http://wiki.kartbuilding.net/mod\_watch-4.3\_apache22\_mod.tar.gz
#gunzip and tar -xvf
nano Makefile.dso
//change APXS location to:
APXS=/usr/bin/apxs2

make -f Makefile.dso build
make -f Makefile.dso install
```

Add and config mod_watch apache module

```
vi /etc/apache2/mods-available/watch.load
//add in
LoadModule watch_module /usr/lib/apache2/modules/mod_watch.so

a2enmod watch
vi /etc/apache2/sites-enabled/01-default (or whatever vhost you want)
<Location /watch-info>
    SetHandler watch-info
    Order deny,allow
    Deny from all
    Allow from 127.0.0.1 ::1
</Location>
<Location /watch-list>
    SetHandler watch-list
    Order deny,allow
    Deny from all
    Allow from 127.0.0.1 ::1
</Location>

/etc/init.d/apache2 restart
```

Checking mod_watch for errors

```
elinks http://localhost/watch-list
#check for output
```

mod_watch Errors received

in error.log with Debian Squeeze running 2.6.32-5-amd64

```
tail -f /var/log/apache2/error.log
//I noticed the following warns when I accessed any vhosts with mod_watch enabled
[Fri Mar 18 22:35:36 2011] [crit] (20014)Internal error: shGetLockedEntry(7fca84e5aae8, "SERVER") fa
[Fri Mar 18 22:35:36 2011] [error] shVerifyString(7fca84e5aae8, 7fca83c946f8) failed!
[Fri Mar 18 22:35:36 2011] [crit] (20014)Internal error: shGetLockedEntry(7fca84e5aae8, "monitor2.bu
[Fri Mar 18 22:35:36 2011] [error] shVerifyString(7fca84e5aae8, 7fca83c94718) failed!
[Fri Mar 18 22:35:36 2011] [crit] (20014)Internal error: shGetLockedEntry(7fca84e5aae8, "SERVER") fa
```

resolve mod_watch errors

Googling led me to <http://forums.cacti.net/viewtopic.php?p=186639#p186639>

```
cd /root/mod_watch-4.3_apache22_mod/
vi SharedHash.c
//on line 102 change to look like:
shVerifyString(struct shTable *tp, char *str)
{
    /* if ((char *) tp->shared <= str && str < (char *) tp->eshared) { */
    if ((char *) apr_shm_baseaddr_get(tp->shared) <= str && str < (char *) tp->eshared) {
        if (str + strlen(str) < (char *) tp->eshared)
            return str;
    }
}
```

So now browsing various vhosts on the server showed no errors in error.log. Using elinks (apt-get install elinks) going to <http://localhost/watch-list> showed results.

Munin Plugin for Mod_Watch

```
cd /usr/share/munin/plugins
wget http://wiki.kartbuilding.net/apache\_watch
chmod 755 apache_watch
vi apache_watch
//change my $URL = exists $ENV{'url'} ? $ENV{'url'} : "http://monitor2.burkesys.com:%d/watch-list";
my $URL = exists $ENV{'url'} ? $ENV{'url'} : "http://server.com:%d/watch-list";

ln -s /usr/share/munin/plugins/apache_watch /etc/munin/plugins/apache_watch_requests
```

That should do it. Once mod_watch is working and accessible munin should pick it up ok. References:
<http://stubbytoe.co.nz/index.php/2009/09/apache-vhost-monitoring-with-munin/>
http://exchange.munin-monitoring.org/plugins/apache_watch/_version/1
<http://forums.cacti.net/about15475.html>

Munin_Statistics

This is a nice apache module. I'm surprised that there is no maintained similar apache module. I've seen another similar munin plugin that parses `/var/log/apache2/vhost%_access.log` to generate these graphs, however it requires separate log files for each vhost and parses the logs every so often. Any more info / queries / suggestions drop us a mail [sburke\[at\]burkesys.com](mailto:sburke[at]burkesys.com)

Migrate Munin RRD Graphs between x32 and x64

Although not critical, on setting up a new munin server on x64 debian, I wanted to keep the existing graphs which were made on a x32 debian server. Copying across the various folders in `/etc/munin/munin.conf` was easy. However when munin ran on the new x64 server errors appeared in `/var/log/munin/`. The reason for this is that RRD is x32 or x64 specific.

See: <http://ken.ipl31.net/2009/04/27/this-rrd-was-created-on-other-architecture/>

The above kind person created two scripts, a `dump.sh` and a `restore.sh` that goes through all rrd files munin created and exports them out as xml files. It worked nicely.

```
dump.sh:
#!/bin/bash
MUNIN_DIR="/var/lib/munin"

for f in `find ${MUNIN_DIR} -name '*.rrd' -print` ;
do
f_xml=`dirname ${f}`/`basename ${f} .rrd`.xml
rrdtool dump "$f" > "${f_xml}"
chown root:0 "${f_xml}"
done
```

Scp across `/var/lib/munin` onto the new server. (Might want to tar it up, as it'd be quite big in size).

```
restore.sh:
#!/bin/bash
MUNIN_DIR="/var/lib/munin"

for f in `find ${MUNIN_DIR} -name '*.xml' -print` ;
do
f_rrd=`dirname ${f}`/`basename ${f} .xml`.rrd
mv -f "${f_rrd}" "${f_rrd}.bak"
chown root:0 "${f_rrd}.bak"
rrdtool restore "$f" "${f_rrd}"
chown munin:munin "${f_rrd}"
done
```

Code above from: <http://ken.ipl31.net/2009/04/27/this-rrd-was-created-on-other-architecture/>