

# Network and Server Statistics using Cacti

PacNOG 6  
November 2009

Hervey Allen  
Carlos Vicente



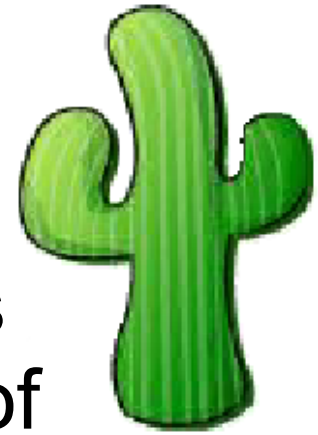
*nsrc@PacNOG 6  
Nadi, Fiji*

# Introduction

- A tool to monitor, store and present network and system/server statistics
- Designed around RRDTool with a special emphasis on the graphical interface
- Almost all of Cacti's functionality can be configured via the Web.

<http://www.cacti.net/>

# Introduction



**Cacti:** Uses RRDtool, PHP and stores data in MySQL. It supports the use of SNMP and graphics with MRTG.

*“Cacti is a complete frontend to RRDTool, it stores all of the necessary information to create graphs and populate them with data in a MySQL database. The frontend is completely PHP driven. Along with being able to maintain Graphs, Data Sources, and Round Robin Archives in a database, cacti handles the data gathering. There is also SNMP support for those used to creating traffic graphs with MRTG.”*

# General Description of Cacti

1. Cacti is written as a group of PHP scripts.
2. The key script is “poller.php”, which runs every 5 minutes (by default). It resides in /usr/share/cacti/site.
3. To work poller.php needs to be in /etc/cron.d/cacti like this:

```
MAILTO=root
```

```
*/5 * * * * www-data php /usr/share/cacti/site/poller.php >/dev/null 2>/var/log/cacti/poller-error.log
```

4. Cacti uses RRDtool to create graphs for each device and data that is collected about that device. You can adjust all of this from within the Cacti web interface.
5. The RRD data is stored in a MySQL database along with descriptions of each device that is monitored.
6. The RRD files are located in /var/lib/cacti/rra.

# Advantages

**You can measure Availability, Load, Errors and more all with history.**

- Cacti can view your router and switch interfaces and their traffic, including all error traffic as well.
- Cacti can measure drive capacity, CPU load (network h/w and servers) and much more. It can react to conditions and send notifications based on specified ranges.

## Graphics

- Allows you to use all the functionality of rrdgraph to define graphics and automate how they are displayed.
- Allows you to organize information in hierarchical tree structures.

## Data Sources

- Permits you to utilize all the functions of rrdcreate and rrdupdate including defining several sources of information for each RRD file.

# Advantages cont.

## Data Collection

- Supports SNMP including the use of *php-snmp* or *net-snmp*
- Data sources can be updated via SNMP or by defining scripts to do this.
- An optional component, *cactid*, implements SNMP routines in C with multi-threading. Important for very large installations, but not tested formally.

## Templates

- You can create templates to reuse graphics definitions, data and device sources

## User Management

- You can manage users locally or via LDAP and you can assign granular levels of authorization by user or groups of users.

# Disadvantages

## Configuration of Interfaces is Tedious

- The first time you add an interfaces, add graphics for each interface and place these graphics correctly on a hierarchical menu requires considerable time and effort.
- It's very important that you keep your Cacti configuration up-to-date with your network. You must either assign someone to do this, or create appropriate scripts and data shares for this purpose.
- If you make a configuration error it can be tedious to correct it.

But, in reality, for continuous use or large installations it is likely that you will be using scripts and tools to automate the configuration of Cacti.

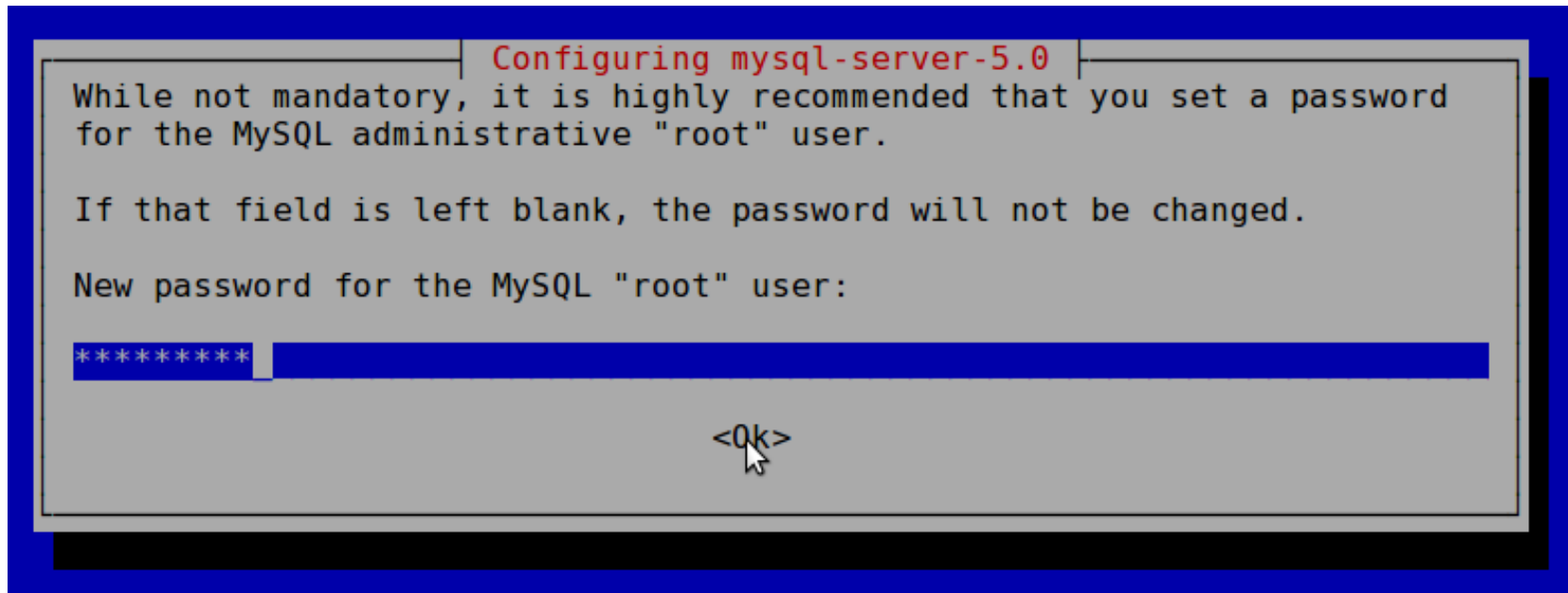
# Installation: Ubuntu Server 9.04

- Available in RPM form and packages for Gentoo, Red Hat, Fedora, SuSE, FreeBSD, etc.
- It is necessary to install *cactid* separately if you wish to use this for larger installations. Again, this code has not been formally measured for improved performance.
- In Ubuntu/Debian...

```
# apt-get install cacti
```

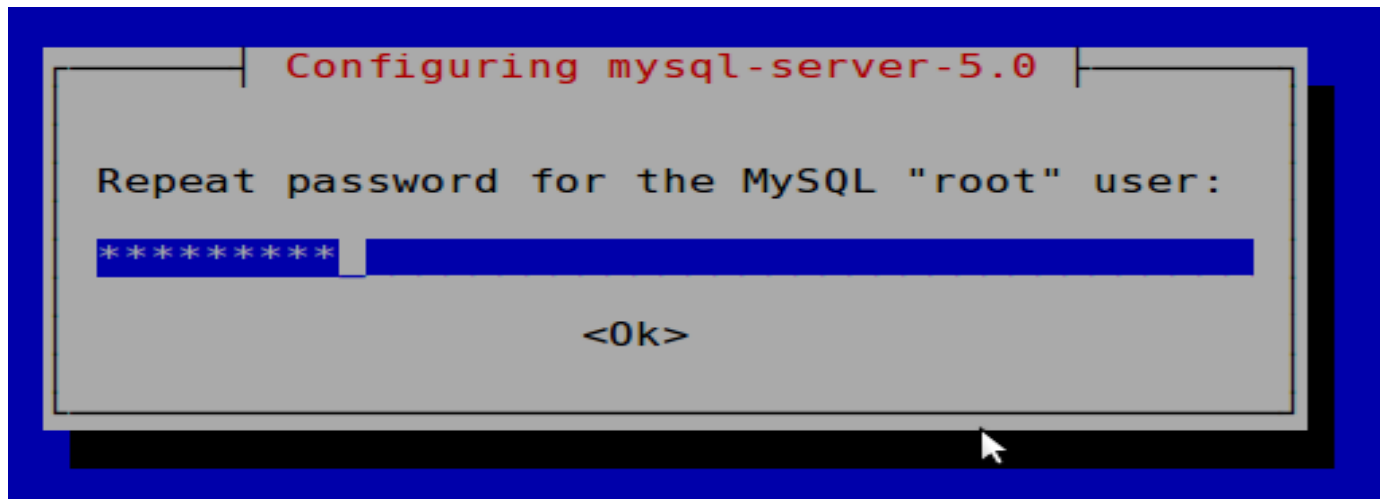


# Installation: 2



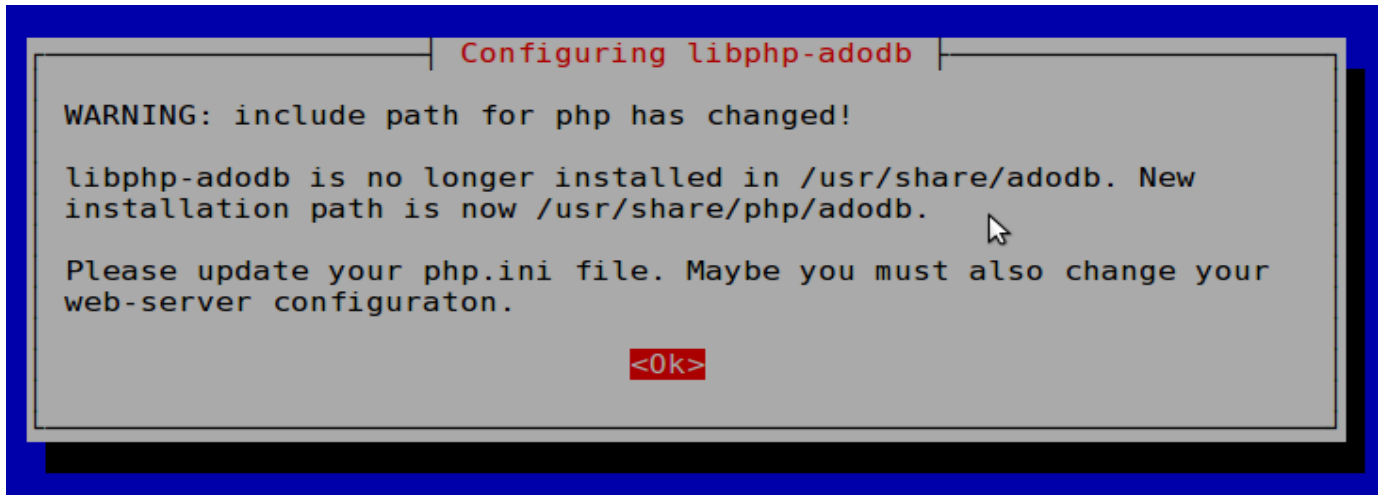
Use the workshop password

# Installation: 3



Again, use the workshop password

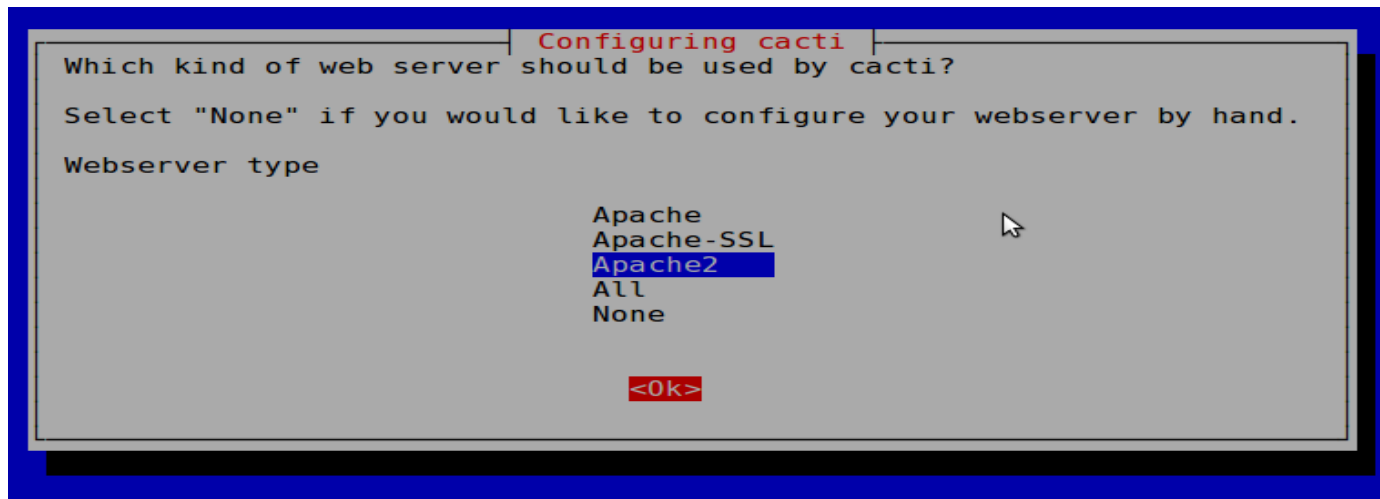
# Installation: 4



```
Configuring libphp-adodb  
WARNING: include path for php has changed!  
  
libphp-adodb is no longer installed in /usr/share/adodb. New  
installation path is now /usr/share/php/adodb.  
  
Please update your php.ini file. Maybe you must also change your  
web-server configuraton.  
  
<Ok>
```

Informational message. Is not normally an issue.

# Installation:5



We are using Apache2. Be sure this is chosen, then highlight <Ok> and press <ENTER> to continue.

# Installation: 6

```
Configuring cacti

cacti must have a database installed and configured before it can be
used.  If you like, this can be handled with dbconfig-common.

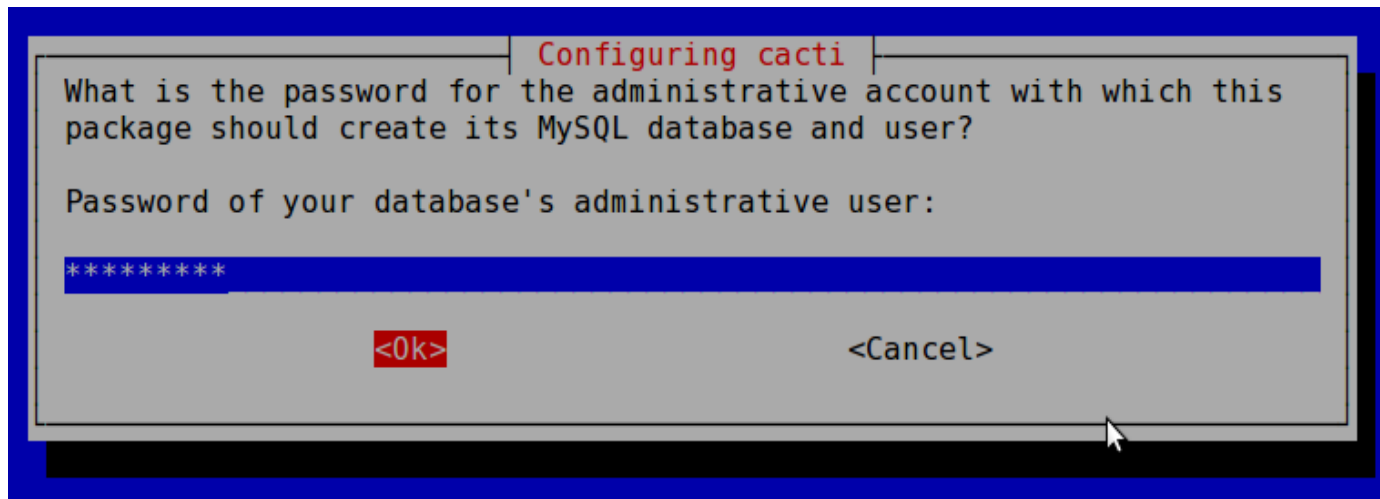
If you are an advanced database administrator and know that you want to
perform this configuration manually, or if your database has already
been installed and configured, you should refuse this option.  Details
on what needs to be done should most likely be provided in
/usr/share/doc/cacti.

Otherwise, you should probably choose this option.

Configure database for cacti with dbconfig-common?
<Yes> <No>
```

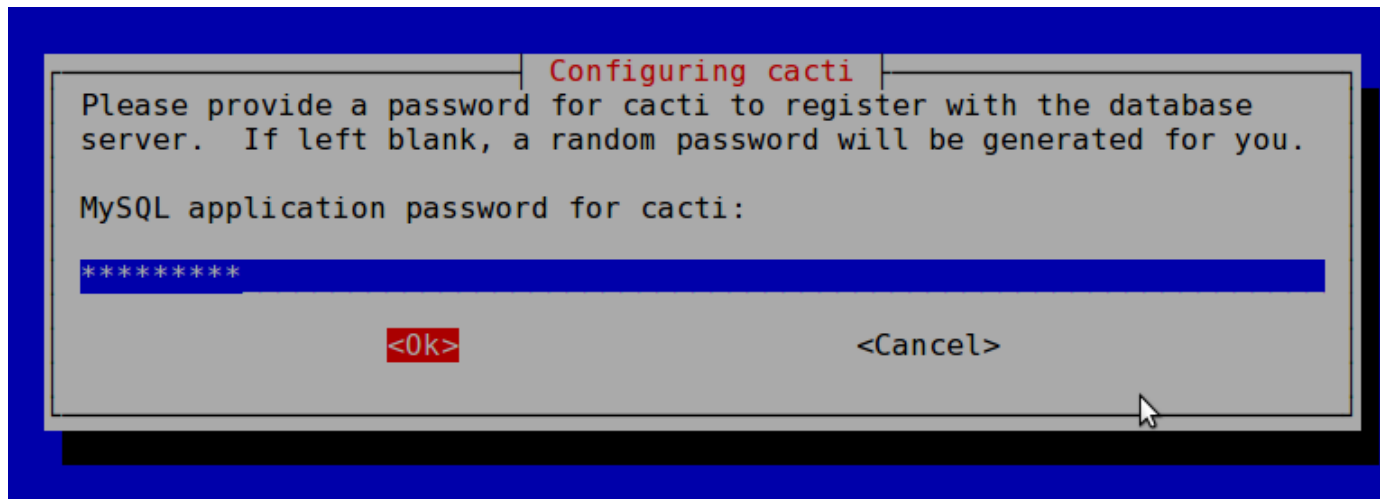
Choose <Yes>. If you choose <No> you will have to manually configure your database at a later time.

# Installation:7



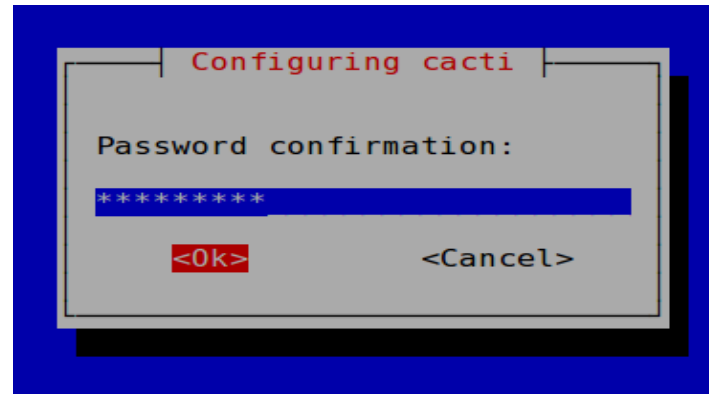
Use our workshop password.

# Installation:8



Again, use the workshop password.

# Installation:9



Finally, one last time, use the workshop password.



# cacti: Installation

**Now use a web browser and open the following address:**

**`http://localhost/cacti`**

**You will see the following...**

# cacti: Installation

## Cacti Installation Guide

Thanks for taking the time to download and install cacti, the complete graphing solution for your network. Before you can start making cool graphs, there are a few pieces of data that cacti needs to know.

Make sure you have read and followed the required steps needed to install cacti before continuing. Install information can be found for [Unix](#) and [Win32](#)-based operating systems.

Also, if this is an upgrade, be sure to reading the [Upgrade](#) information file.

Cacti is licensed under the GNU General Public License, you must agree to its provisions before continuing:

```
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.
```

Next >>

Press “Next >>”

# cacti: Installation

**Cacti Installation Guide**

Please select the type of installation

The following information has been determined from Cacti's configuration file. If it is not correct, please edit 'include/config.php' before continuing.

Database User: cacti  
Database Hostname:  
Database: cacti  
Server Operating System Type: unix

**Next >>**

Choose "New Install" and press "Next >>" again.

# cacti: Installation

**Cacti Installation Guide**

Make sure all of these values are correct before continuing.

**[FOUND] RRDTool Binary Path:** The path to the rrdtool binary.  
/usr/bin/rrdtool

**[FOUND] PHP Binary Path:** The path to your PHP binary file (may require a php recompile to get this file).  
/usr/bin/php

**[FOUND] snmpwalk Binary Path:** The path to your snmpwalk binary.  
/usr/bin/snmpwalk

**[FOUND] snmpget Binary Path:** The path to your snmpget binary.  
/usr/bin/snmpget

**[FOUND] snmpbulkwalk Binary Path:** The path to your snmpbulkwalk binary.  
/usr/bin/snmpbulkwalk

**[FOUND] snmpgetnext Binary Path:** The path to your snmpgetnext binary.  
/usr/bin/snmpgetnext

**[FOUND] Cacti Log File Path:** The path to your Cacti log file.  
/usr/share/cacti/site/log/cacti.log

**SNMP Utility Version:** The type of SNMP you have installed. Required if you are using SNMP v2c or don't have embedded SNMP support in PHP.  
NET-SNMP 5.x

**RRDTool Utility Version:** The version of RRDTool that you have installed.  
RRDTool 1.2.x

**NOTE:** Once you click "Finish", all of your settings will be saved and your database will be upgraded if this is an upgrade. You can change any of the settings on this screen at a later time by going to "Cacti Settings" from within Cacti.

**Finish**

Your screen should look like this. If it does not ask your instructor for help.

Press "Finish"

## Note!

Be sure that "RRDTool 1.2.x" is chosen and *not* "1.0.x".

# cacti: First Login



## User Login

Please enter your Cacti user name and password below:

User Name:

Password:

Login

First time login use:

User Name: *admin*

Password: *admin*

# cacti: Password Change



## User Login

**\*\*\* Forced Password Change \*\*\***

Please enter a new password for cacti:

Password:

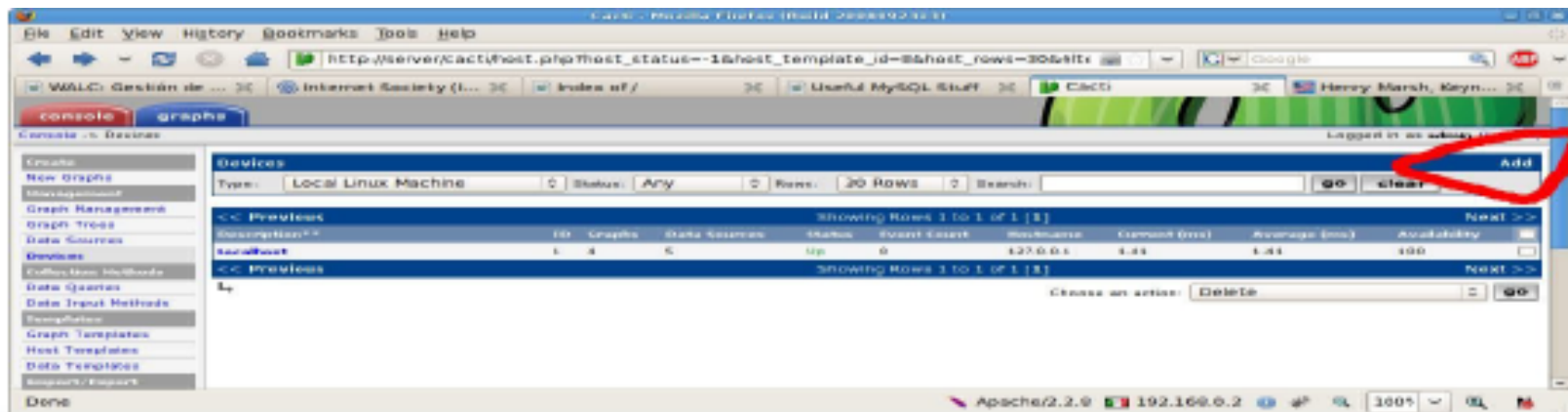
Confirm:

Save

Now you must change the *admin* password. Please use the workshop password.

# Add Devices: 1

- **Management -> Devices -> Add**
- Specify device attributes
  - Choose a device template and this will ask you for additional information about the device.
  - You can add additional templates when, or if, you want.



# Add Devices: 2

console graphs

Console -> Devices -> (Edit) Logged in as admin (Logout)

**Devices [new]**

**Description**  
Give this host a meaningful description.

**Hostname**  
Fully qualified hostname or IP address for this device.

**Host Template**  
Choose what type of host, host template this is. The host template will govern what kinds of data should be gathered from this type of host.

**Notes**  
Enter notes to this host.

**Disable Host**  
Check this box to disable all checks for this host.  Disable Host

**Availability/Reachability Options**

**Downed Device Detection**  
The method Cacti will use to determine if a host is available for polling. NOTE: It is recommended that, at a minimum, SNMP always be selected.

**Ping Method**  
The type of ping packet to sent. NOTE: ICMP on Linux/UNIX requires root privileges.

**Ping Port**  
TCP or UDP port to attempt connection.

**Ping Timeout Value**  
The timeout value to use for host ICMP and UDP pinging. This host SNMP timeout value applies for SNMP pings.

**Ping Retry Count**  
The number of times Cacti will attempt to ping a host before failing.

**SNMP Options**

**SNMP Version**  
Choose the SNMP version for this device.

**SNMP Community**  
SNMP read community for this device.

**SNMP Port**  
Enter the UDP port number to use for SNMP (default is 161).

**SNMP Timeout**  
The maximum number of milliseconds Cacti will wait for an SNMP response (does not work with php-snmp support).

**Maximum OID's Per Get Request**  
Specified the number of OID's that can be obtained in a single SNMP Get request. NOTE: This feature only works when using Spine.



# Add Devices: 3

Choose SNMP version 2 for this workshop.

At your own location you can use SNMP version 3 if your devices support this.

SNMP access is a security issue:

- Version 2 is not encrypted
- Watch out for globally readable “public” communities
- Be careful about who can access r/w communities.

# Add Devices: 4

Note the “Associated Data Queries” menu:

- By default Cacti does not use snmp to query a device. You must be sure to add this.

### Associated Graph Templates

Graph Template Name	Status
1) Linux - Memory Usage	Is Being Graphed (Edit)
2) Unix - Load Average	Is Being Graphed (Edit)
3) Unix - Logged In Users	Is Being Graphed (Edit)
4) Unix - Processes	Is Being Graphed (Edit)

Add Graph Template:

### Associated Data Queries

Data Query Name	Debugging	Re-Index Method	Status
1) Unix - Get Mounted Partitions	(Verbose Query)	Uptime Goes Backwards	Success [2 Items, 1 Row]

Add Data Query:  Re-Index Method:

# Create Graphics

- Chose the “Create graphs for this host”
- Under Graph Templates generally check the top box that chooses *all* the available graphs to be displayed.
- Press Create.
- You can change the default colors, but the predefined definitions generally work well.

# Create Graphics: Step 1

The screenshot shows the Nagios Core 4 web interface. The top navigation bar has 'console' and 'graphs' tabs. The breadcrumb is 'Console -> Create New Graphs'. The user is logged in as 'admin'. The main content area is for host 'pc1 (pc1.mgmt.conference.apricot.net) Local Linux Machine'. There are two links: '\*Edit this Host' and '\*Create New Host'. Below this is a 'Graph Templates' section with a table of templates. The 'Create' column has a dropdown menu with '(Select a graph type to create)'. Below that is a 'Data Query [Unix - Get Mounted Partitions]' section with a table of device names and mount points. At the bottom right are 'cancel' and 'create' buttons.

console graphs

Console -> Create New Graphs Logged in as admin (Logout)

pc1 (pc1.mgmt.conference.apricot.net) Local Linux Machine

Host:  Graph Types:

[\\*Edit this Host](#)  
[\\*Create New Host](#)

Graph Template Name	
Create: Linux - Memory Usage	<input checked="" type="checkbox"/>
Create: Unix - Load Average	<input checked="" type="checkbox"/>
Create: Unix - Logged in Users	<input checked="" type="checkbox"/>
Create: Unix - Processes	<input checked="" type="checkbox"/>
Create: <input type="text" value="(Select a graph type to create)"/>	

Data Query [Unix - Get Mounted Partitions]		
Device Name	Mount Point	
/dev/sda1	/	<input checked="" type="checkbox"/>

# Create Graphics: Step 2

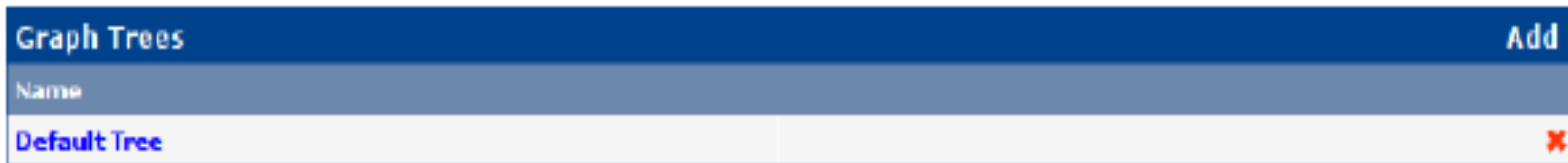
The screenshot shows a web application interface for creating graphs. The top navigation bar includes 'console' and 'graphs' tabs. Below the navigation bar, the breadcrumb trail reads 'Console -> Create New Graphs -> Create Graphs from Data Query'. The user is logged in as 'admin' with a 'Logout' link. The left sidebar contains a menu with categories: 'Create', 'New Graphs', 'Management', 'Graph Management', 'Graph Trees', 'Data Sources', 'Devices', 'Collection Methods', 'Data Queries', 'Data Input Methods', 'Templates', 'Graph Templates', 'Host Templates', 'Data Templates', 'Import/Export', 'Import Templates', 'Export Templates', 'Configuration', 'Settings', 'Utilities', 'System Utilities', 'User Management', and 'Logout User'. A green cactus icon is visible at the bottom of the sidebar. The main content area displays three graph creation options, each with a 'Legend Color' field. The 'Unix - Logged in Users' option has a color of '4668E4', and the 'Unix - Processes' option has a color of 'F51D30'. At the bottom right, there are 'cancel' and 'create' buttons.

# View the Graphics

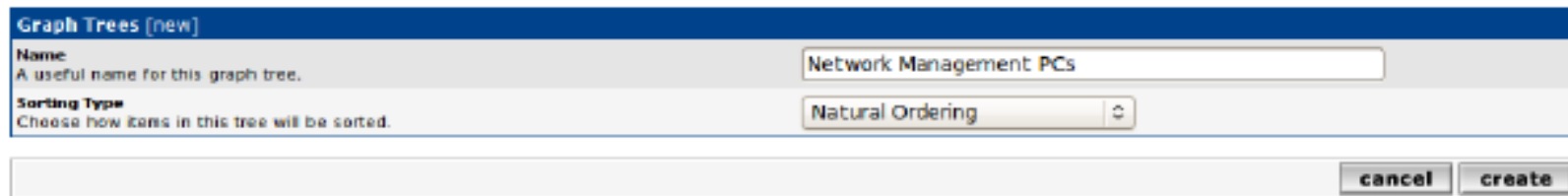
- Place the new device in its proper location in your tree hierarchy.
- Building your display hierarchy is your decision. It might make sense to try drawing this out on paper first.
  - Under Management → Graph Trees select the Default Tree hierarchy (or, create one of your own).

# Graphics Tree

First, press “Add” if you want a new graphing tree:



Second, name your tree, choose the sorting order (the author likes Natural Sorting and press “create”:



# Graphics Trees

Third, add devices to your new tree:

**Graph Trees** [edit: Network Management PCs]

**Name**  
A useful name for this graph tree.

**Sorting Type**  
Choose how items in this tree will be sorted.

**Tree Items** Add

Item	Value
No Graph Tree Items	

Once you click “Add” you can add “Headers” (separators), graphs or hosts. Now we'll add Hosts to our newly created graph tree:

**Tree Items**

**Parent Item**  
Choose the parent for this header/graph.

**Tree Item Type**  
Choose what type of tree item this is.

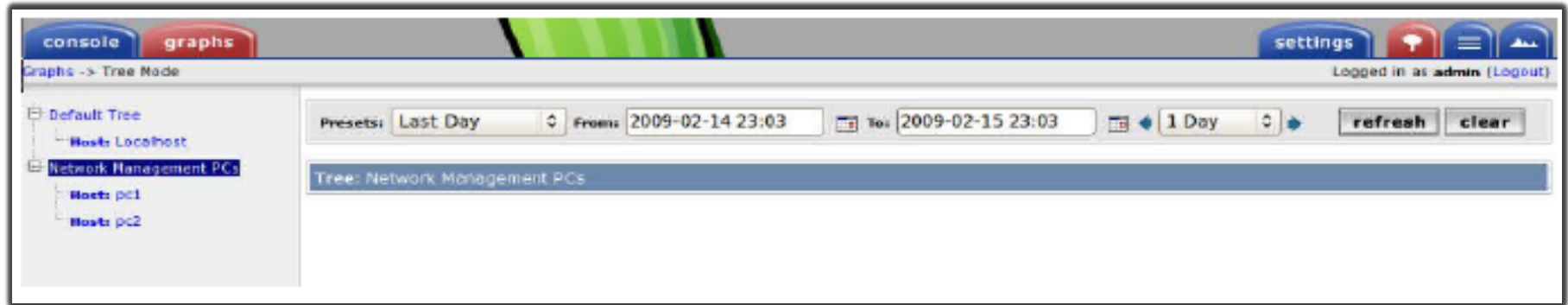
**Tree Item Value**

**Host**  
Choose a host here to add it to the tree.

**Graph Grouping Style**  
Choose how graphs are grouped when drawn for this particular host on the tree.



# Graphics Tree with 2 Devices

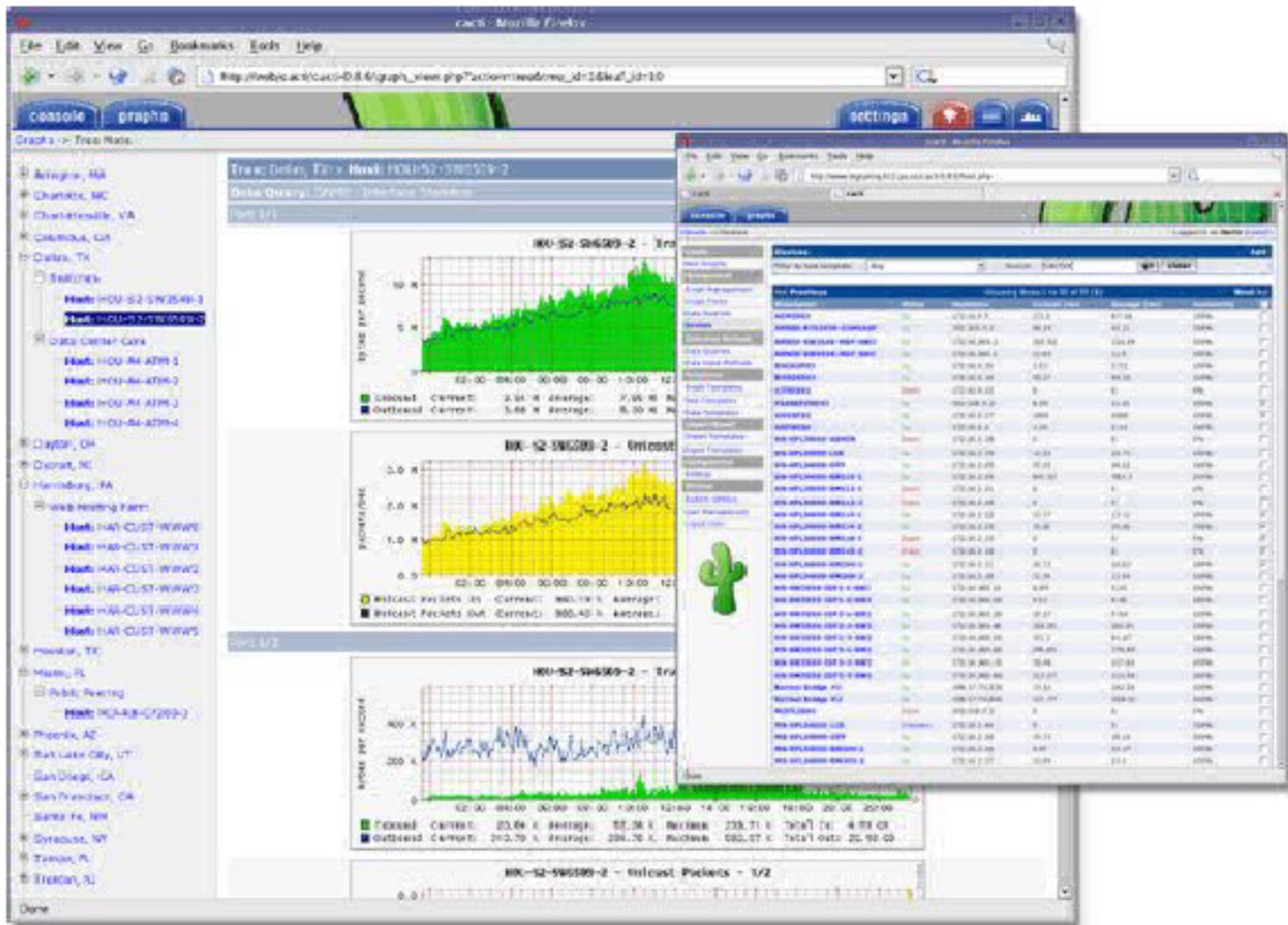


- Our graphics tree *just* after the first two devices were added.
- So far, no graphics are displayed – the first graphics can take up to 5 minutes to display.
- Cacti graphs are stored on disk and updated using RRDTool via the poller.php script, which, by default, is run every five minutes using `cron`.

A much larger example →

*nsrc@PacNOG 6  
Nadi, Fiji*

# An Example...



nsrc@PacNOG 6  
Nadi, Fiji

# Conclusions

- Cacti is very flexible due to its use of templates.
- Once you understand the concepts behind RRDTool, then how Cacti works should be (more or less) intuitive.
- The visualization hierarchy of devices helps to organize and discover new devices quickly.
- There are very few to no statistics available about the performance of *cactid* (volunteers are welcome!).
- It is not easy to do a rediscover of devices.
- To add lots of devices requires lots of time and effort. Software such as Netdot, Netdisco, IPPlan, TIPP can help – as well as local scripts that update the Cacti back-end MySQL database directly.

# References

- Cacti Web Site:  
<http://www.cacti.net/>
- Cacti Discussion Group:  
<http://forums.cacti.net/>

# Older Configuration Issues

- Cacti uses MySQL to store configurations. In older Ubuntu versions it was necessary to manually create the cacti MySQL database and set the permissions:

```
# mysqladmin --user=root create cacti
# mysql cacti < cacti.sql
# mysql --user=root mysql

mysql> GRANT ALL ON cacti.* TO cactiuser@localhost IDENTIFIED BY 'cacti_pass';
mysql> flush privileges;
```

- It was, also, sometimes necessary to manually specify the cacti connection parameters in `/etc/cacti/db.php`:

```
$database_type = "mysql";
$database_default = "cacti";
$database_hostname = "localhost";
$database_username = "cactiuser";
$database_password = "cacti_pass";
$database_port = "3306";
```

# Older Configuration Issues

- Make sure that there is a cron job that has been configured as well – Likely in `/etc/cron.d/cacti`.
- This will be something like:

```
*/5 * * * * www-data php /usr/share/cacti/site/poller.php >/dev/null \ 2>/  
var/log/cacti/poller-error.log
```

- This is not necessary with the Debian package in Ubuntu 8.10, and later.

# Using *cactid* Alternate Poller Code

```
# tar xvzf cacti-cactid-0.8.6.tar.gz
# cd cactid-0.8.6
# ./configure
# make
# make install
```

```
# vi /usr/local/cactid/bin/cactid.conf
DB_Host      localhost
DB_Database  cacti
DB_User      cactiuser
DB_Pass      cacti_pass
DB_Port      3306
```

In the Web interface go to:

- **Configuration -> Settings -> Paths -> Cactid Poller File Path** and specify the location of cactid.
- Go to **Poller** and in **Poller Type**, select **cactid**