## <http://www.slsmk.com/setup-syslog-with-loganalyzer-on-ubuntu-server/>

## Setup Syslog on Ubuntu Server

This article will cover:  
1) Setting up a syslog server to log messages from local and remote sources.  
2) Setup a GUI front end showing syslog items.

I’ve done this on both Ubuntu Server 10.04 and 12.04.

1. Get some items loaded from the repository.

# *apt-get install build-essential apache2 php5 php5-gd libapache2-mod-php5 mysql-server php5-mysql rsyslog*

1. Edit */etc/rsyslog.conf* and uncomment or add the following. This will set the server to accept inbound syslog messages on UDP port 514.

### rsyslog: Enabling remote logging service in Ubuntu

Newer versions of Ubuntu (since 9.10 according to rsyslog wiki: <http://wiki.rsyslog.com/index.php/Ubuntu>) comes with *rsyslog* instead of *sysklogd*. That obviously didn't work, while trying to enable it via the sysklogd way by adding the "-r" option in the startup script.   
What you'll need to do is just uncomment 2 lines in the /etc/rsyslog.conf file. This will set the server to accept inbound syslog messages on UDP port 514

*# provides UDP syslog reception  
$ModLoad imudp  
$UDPServerRun 514*

That's if you want to provide UDP syslog service. Uncomment the following 2 lines if you want to provide TCP syslog service:

*# provides TCP syslog reception  
$ModLoad imtcp  
$InputTCPServerRun 514*

Once you've made the changes, either reload or restart the rsyslogd service:

myhost:/etc# */etc/init.d/rsyslog reload*  
Reloading enhanced syslogd: rsyslogd.

I've chose to enable UDP for my server.  We'll use netstat to check if rsyslogd is listening to the specified port:

myhost:/etc*# netstat -tlnup | grep 514*  
udp        0      0 0.0.0.0:514             0.0.0.0:\*                           13282/rsyslogd  
udp6       0      0 :::514                  :::\*                                13282/rsyslogd

*rsyslogd* is indeed listening to the proper port and protocol.

1. Since the log analyzer runs on php, we need to tell apache how to handle php pages. Edit */etc/apache2/apache2.conf* and add in the following item underneath “DefaultType None”

*DefaultType text/plain*

*Addtype application/x-httpd-php .php*

Note: If this step is not done properly, you will get a message when loading the syslog web page prompting you to save the file instead of Apache displaying the file.

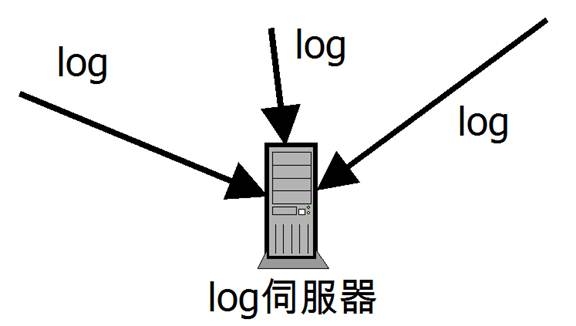
Note: Restart the web server after you made configuration change like: # */etc/init.d/apache2 restart*.

**Direct syslog event to MySQL database**

<http://download.ithome.com.tw/article/index/id/1368>

△注意：本文是配合 Ubuntu Server 12.04 LTS 安裝

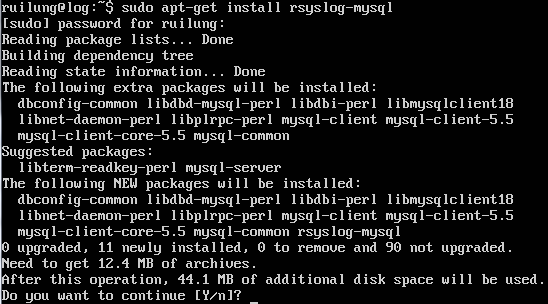
▲ 圖一、LOG伺服器收容其它電腦的LOG。



使用rsyslog架設LOG伺服器。rsyslog在Ubuntu中預設就會安裝，預設只收取本機的LOG資料，LOG以純文字方式存在/var/log目錄中，用這方式收其它伺服器送來的LOG，最後應該會錯亂，所以要安裝rsyslog的mysql支援，如此可將收進來的LOG放到資料庫中，如果要當LOG伺服器的電腦沒有安裝AMP，可參考「[動態網頁伺服器 -- apache + mysql+ php](http://download.ithome.com.tw/article/index/id/1109)」，如果有，那往下來安裝rsyslog mysql支援，安裝的指令則如下所示：

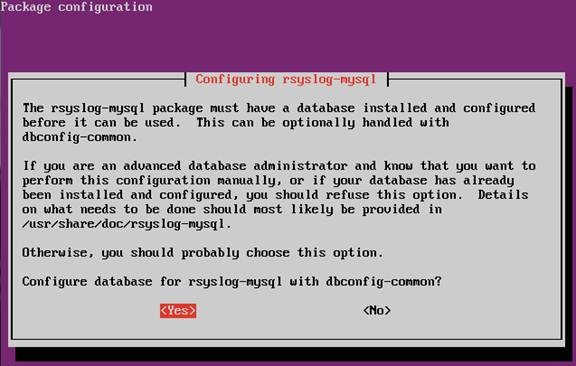
*● sudo apt-get install rsyslog-mysql*

▲ 圖二、安裝rsyslog-mysql的示意圖。

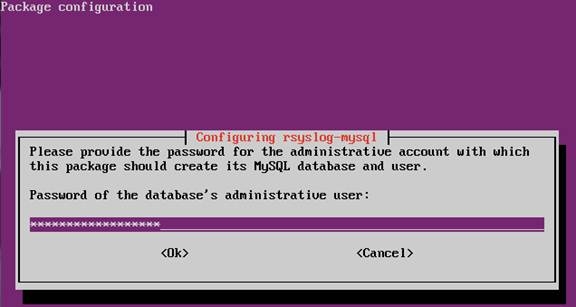


透過rsyslog-mysql可將收進來的資料轉存到資料庫中，那是存到那個資料庫的裡面呢？答案在<<圖六>>，新資料庫的名字叫Syslog(Linux下大小寫是有差)，Syslog是在rsyslog-mysql安裝的過程中進行設定，設定過程中需要輸入mysql的root密碼<<圖四>>，以新增使用者(預設會新增一個rsyslog的資料庫使用者)與資料庫Syslog的動作。

▲ 圖三、安裝rsyslog-mysql過程，提示將設定資料庫。



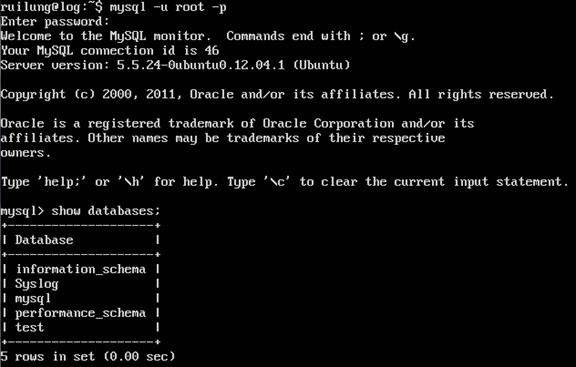
▲ 圖四、輸入資料庫root密碼，以便有權限進行下一步。



▲ 圖五、輸入rsyslog-mysql的資料庫使用者密碼。



▲ 圖六、以show database指令查看新增的資料庫為Syslog。



rsyslog的設定檔放在/etc/rsyslog.conf中，要做下述的設定修改，’#’後是說明，不需要輸入。

● sudo vi /etc/rsyslog.conf

█ $ModLoad imudp

讓rsyslog服務，除了做本機的LOG收集外，也要收網路的資料，這個功能需要載入udp的模組，將預設行前的’#’移除，如果想要透過tcp收網路LOG也可以，只要將設定檔中，其它段的imtcp的Mod載入就可。  
█ $UDPServerRun 514

將預設行前的’#’移除，一般LOG伺服器會聽514 port，如果想要修改listen的port也可以，只是LOG伺服器改了，送LOG出來的電腦/伺服器/網通設備，也要能配合修改才能正確的傳送LOG。

● sudo vi /etc/rsyslog.d/mysql.conf

█ $ModLoad ommysql

這一行及下一行要手動新增，這一行是說要載入mysql模組，以讓LOG可寫入資料庫中。  
█ \*.\*       :ommysql:127.0.0.1,Syslog,rsyslog,topsecret

資料庫連線設定，前面的\*.\*是說將所有的資料寫入資料庫中，查看本機的/etc/rsyslog.d/50-default.conf，還可以看到更多的LOG等級寫法，例如只傳kernel的LOG是kern.\*；127.0.0.1是mysql所在的IP，Syslog是使用的資料庫名稱，rsyslog是資料庫使用者名稱，topsecret是使用者密碼(記得改成<<圖五>>中設定的密碼)。

▲ 圖七、rsyslog.conf設定示意圖。



 設定全部做 就要把rsyslog服務重啟，重啟的指令是

● sudo service rsyslog restart

現在有一個可以收遠端LOG的伺服器了，網通設備應該直接指定丟到LOG伺服器的IP與PORT就可以傳送LOG資料，在Ubuntu的伺服器中，我們則需在/etc/rsyslog.d/50-default.conf新增LOG伺服器設定，再重啟rsyslog服務，以傳LOG到遠端伺服器中，新增的設定如下所示。

● sudo vi /etc/rsyslog.d/50-default.conf

     █ \*.\*   @192.168.1.14 #將此台電腦的LOG全部(\*.\*)傳到192.168.1.14的伺服器上(剛才設定伺服器的IP)。

To direct the syslog events to a log file, add the command in */etc/rsyslog.conf* like:

*auth.\* /var/log/auth.log*

All syslog events from auth facility are written into auth.log file.

Type *tail -f /var/log/auth.log* to verify the content.

**On to the LogAnalyzer**.  
1. Download the latest log analyzer from the adiscon web site at *http://loganalyzer.adiscon.com/downloads*

*cd /opt*

*wget http://download.adiscon.com/loganalyzer/loganalyzer-3.6.2.tar.gz*

2. Unzip and extract the file.

*gunzip loganalyzer-3.6.2.tar.gz*

*tar -xvf loganalyzer-3.6.2.tar*

1. Copy the LogAnalyzer /src/ folder to the Apache www root or subfolder, copy the install script, make executable, and run the install scripts. It’s important to run the configure script from the same directory that will hold the syslog php files.

*mkdir /var/www/syslog*

*cp -r /opt/loganalyzer-3.6.2/src/\* /var/www/syslog*

*cp -r /opt/loganalyzer-3.6.2/contrib/\*.sh /var/www/syslog*

*chmod +x /var/www/syslog/\*.sh*

*cd /var/www/syslog/*

*./configure.sh*

1. Grant Apache access to syslog.

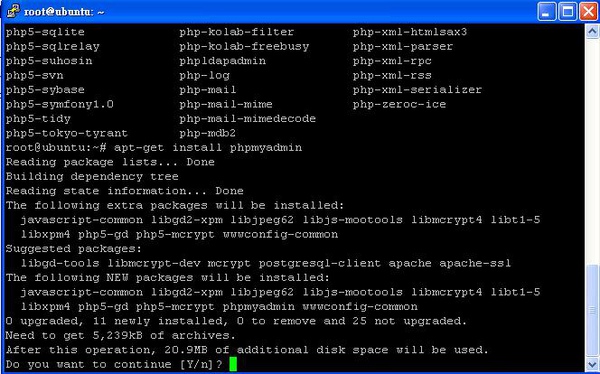
*usermod -G adm www-data*

Use a web browser to hit the new web service at [*http://SERVERNAME/syslog/index.php*](http://SERVERNAME/syslog/index.php) (Ex: [*http://localhost/syslog/index.php*](http://localhost/syslog/index.php)). The page will show a message stating the service is not configured. Follow the steps to setup your syslog front end. Go to <http://140.111.74.38/blog/post/2/632> to see the detailed information about how to initialize LogAnalyzer.

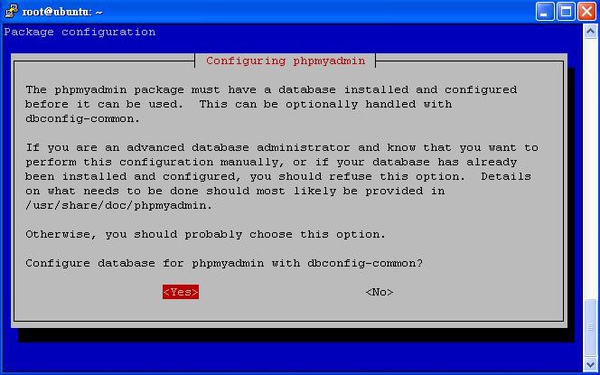
## <http://linadonis.pixnet.net/blog/post/27585552-ubuntu-server-%E5%AE%89%E8%A3%9D-phpmyadmin>

## [Ubuntu Server 安裝 Phpmyadmin](http://linadonis.pixnet.net/blog/post/27585552)

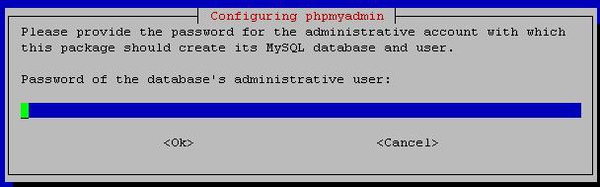
* 1.安裝套件
* *#sudo apt-get install phpmyadmin*



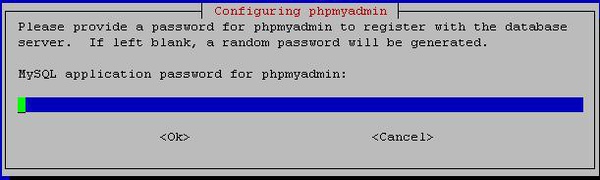
* 2.點選Yes



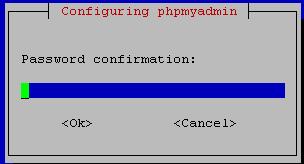
* 3.輸入管理者登入帳號



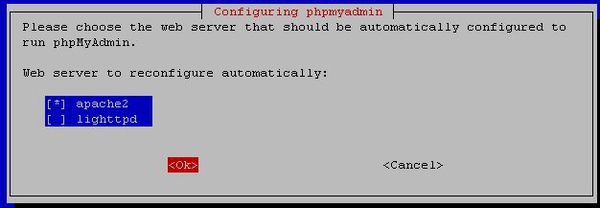
* 4.輸入管理者的密碼



* 5.在輸入一次密碼



* 6.選擇[apach2]



* 7.完成後開啟流覽器輸入[http://127.0.0.1/phpmyadmin/]



* PS:若在連結時出現

# Not Found

* The requested URL /phpmyadmin/ was not found on this server.  
  解決方法:  
  #cd /var/www  
  #ln -s /usr/share/phpmyadmin  
  完成

phpMyAdmin是MySQL最常用的管理工具，但MySQL本身就提供好用的管理程式：mysqladmin.exe，但因為是命令列互動式，大家還是會選擇有UI介面的。實際操作mysqladmin.exe。  
打開DOS視窗後，切換到存放MySQL應用程式的資料夾。以下粗體為實際指令，{}包起來的內容是說明。  
D:\WebServer\mysql\bin>**mysql -h 127.0.0.1 -u root -p** {本行指令：應用程式名稱、指定伺服器、伺服器名稱/網紙、指定使用者、使用者名稱、指定資料庫}  
Enter password:  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 1  
Server version: 5.5.14-log MySQL Community Server (GPL)  
…  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
(進入管理模式，下列所有指令後面需以;結尾)  
mysql> **show databases;** {察看目前擁有的資料庫清單}  
+--------------------+  
| Database |  
+--------------------+  
| information\_schema |  
| mysql |  
| performance\_schema |  
| test |  
+--------------------+  
4 rows in set (0.08 sec)  
  
mysql> **Select user();** {顯示目前擁有的使用者清單}  
+----------------+  
| user() |  
+----------------+  
| root@localhost |  
+----------------+  
1 row in set (0.03 sec)  
  
mysql> **use test;** {選擇要進行操作的資料庫}  
Database changed  
mysql> **show tables;** {顯示資料庫test的資料表清單}  
Empty set (0.06 sec)  
  
mysql> **create table UsrData (Id int, FirstName varchar(20), LastName varchar(20));** {建立一個資料表}  
Query OK, 0 rows affected (0.11 sec)  
  
mysql> **show tables;**  
+----------------+  
| Tables\_in\_test |  
+----------------+  
| usrdata |  
+----------------+  
1 row in set (0.00 sec)  
  
mysql> **desc usrdata;** {檢視資料表usrdata的欄位定義}  
+-----------+-------------+------+-----+---------+-------+  
| Field | Type | Null | Key | Default | Extra |  
+-----------+-------------+------+-----+---------+-------+  
| Id | int(11) | YES | | NULL | |  
| FirstName | varchar(20) | YES | | NULL | |  
| LastName | varchar(20) | YES | | NULL | |  
+-----------+-------------+------+-----+---------+-------+  
3 rows in set (0.06 sec)  
  
mysql> **insert into usrdata (Id, FirstName, LastName) values (0, 'Java', 'Sun');**  
Query OK, 1 row affected (0.03 sec)  
  
mysql> **select \* from usrdata;**  
+------+-----------+----------+  
| Id | FirstName | LastName |  
+------+-----------+----------+  
| 0 | Java | Sun |  
+------+-----------+----------+  
1 rows in set (0.03 sec)  
  
mysql> **select firstname, lastname from usrdata;**  
+-----------+----------+  
| firstname | lastname |  
+-----------+----------+  
| Java | Sun |  
+-----------+----------+  
1 rows in set (0.00 sec)  
  
mysql> **quit;** {離開}  
Bye

==========================================================

*Install MySQLDB so that Django can access MySQL.*

*1. sudo apt-get install libmysqld-dev*

*2. sudo apt-get install python-dev*

*3. Download MySQL-python-1.2.4b4.tar.gz*

*1) python setup.py build*

*2) sudo python setup.py install*

## Install Django and setup the environment path in the *.bashrc* file.

# Setting up Django and mod\_wsgi

## <http://ericholscher.com/blog/2008/jul/8/setting-django-and-mod_wsgi/>

This is assuming Ubuntu 8.04 Server Edition.

**Update**: Take note, this is installing mod\_wsgi 1.3. The latest version of the package is 2.3. If you want to get the latest version from apt, you should use the [Debian 2.3 package](http://packages.debian.org/unstable/python/libapache2-mod-wsgi)

Step 1: *apt-get install libapache2-mod-wsgi*

This should automatically install mod\_wsgi into your apache instance and install it.

Step 2: Create an apache directory on your filesystem, presumably inside of your Django project. I keep my code in ~/Python/Project, so I did:

*mkdir ~/rwqaauto/rat22mysql/apache*

*vim ~/ rwqaauto/rat22mysql /django.wsgi*

Then in that file you need to copy this code:

***import******os****,* ***sys***

*sys.path.append('/home/eric/Python/PROJECT')*

*os.environ['DJANGO\_SETTINGS\_MODULE'] = 'PROJECT.settings'*

***import******django.core.handlers.wsgi***

*application = django.core.handlers.wsgi.WSGIHandler()*

This creates an interface between Django and WSGI. If you start getting errors about not seeing your project or modules, try adjusting and/or adding some things to your sys.path.

Step 3:

Inside your */etc/apache2/*sites-available/ directory, this is where you are going to put your configuration for your server. In /etc/apache2/sites-available/default:

*<VirtualHost \*:80>*

*ServerAdmin eric@ericholscher.com*

*ServerName ericholscher.com*

*ServerAlias www.ericholscher.com*

*DocumentRoot /var/www/*

*LogLevel warn*

*WSGIDaemonProcess ericholscher processes=2 maximum-requests=500 threads=1*

*WSGIProcessGroup ericholscher*

*WSGIScriptAlias / /home/eric/Python/PROJECT/apache/django.wsgi*

*Alias /media /var/www/media/*

*</VirtualHost>*

The last 3 lines of WSGI stuff if what you want to pay attention to. You are pointing WSGIScriptAlias to the file we created in Step 2. The other two WSGI prompts aren’t necessary unless you are running multiple sites on your server. The Alias is so that the /media URLs on your site continue to work, it should point to where ever you have your media files stored.

Hopefully this will get you started along the way to setting up mod\_wsgi on Apache with Django.

## Add username/password (django/django) in MySQL

## *mysql> create user ‘django’@’localhost’ identified by ‘django’*

## *mysql> grant all on Syslog.\* to ‘django’@’localhost’*

在 MySQL 命令視窗下指令：

1. 顯示使用者清單：  
*mysql > select \* from mysql.user; (註：會顯示出一大張的列表，不容易看得清楚)*

*mysql > desc mysql.user; (顯示 mysql 這個資料庫中 user 資料表的所有欄位)*

*mysql > select Host, User from mysql.user; (只顯示mysql資料庫中user資料表的Host及User共2個欄位)*

2. 新增使用者：  
　　*mysql > create user****'UserName'@'Host'****identified by****'Password'****;*

3. 給予權限：  
　　*mysql > grant****OPTION****on****TARGET****to****USER****;*

4. 刪除使用者：  
　　mysql > *drop user***OPTION***'UserName'@'Host'*;

註：  
　1. UserName 為使用者名稱，例如 root , joey , jackey ...，請注意，要加單引號(')  
　2. Password 為密碼，請注意，要加單引號(')

Example:  
　1. 新增使用者：  
*mysql > create user 'joey'@'localhost' identified by 'abcd2345';  
　　　mysql > create user 'joey'@'%' identified by 'abcd2345';*

　2. 給予權限：  
*mysql > grant all on \*.\* to 'joey'@'localhost'; (注意：joey 對整個資料庫的所有資料表都擁有全部的權利)*

*mysql > grant select,insert on myFirstDB.\* to 'joey'@'localhost';*  
　　　　　(joey只擁有對 myFirstDB資料檔[所有表格]的查詢及新增權利)  
　　　　　(權利：SELECT,INSERT,UPDATE,DELETE,CREATE,DROP)

　3. 刪除使用者：  
*mysql > drop user 'joey'@'localhost';*