

SAMSB

Autor: Luiz Cezar Quaquio

Tutorial para instalação do SAMSB (Snort + Apache2 + MySQL + Snorby + BarnYard2) no Debian.
Convenções utilizadas neste documento:

- O símbolo "#" significa que o comando deve ser executado como root;
- O símbolo "\$" significa que o comando deve ser realizado como usuário;
- O símbolo ">" é o prompt do MySQL.

Instalando o MySQL

Instalando o MySQL:

```
# apt-get install mysql-server-5.1 mysql-client libmysqlclient-dev libpcap-dev
```

Durante a instalação será pedido a senha para o usuário root do MySQL, mas se houver algum problema configure a senha do root do MySQL com o comando:

```
# mysqladmin -u root password senharootdomysql
```

Vamos criar o banco de dados e o usuário para acessá-lo:

```
# mysql -u root -p
```

Já no shell do MySQL:

```
> CREATE DATABASE snort;
```

```
> GRANT all privileges ON snort.* TO snort@localhost IDENTIFIED BY 'senha_snort_mysql';
```

```
> flush privileges;
```

```
> quit
```

Instalando o SNORT

Instalando o SNORT:

```
# apt-get install snort-mysql snort-rules-default
```

Criando as tabelas para o snort:

```
# zcat /usr/share/doc/snort-mysql/create_mysql.gz | mysql -u snort -h localhost -p snort
```

Agora vamos configurar o SNORT editando o arquivo /etc/snort/snort.debian.conf.

Primeiro, configure a rede interna no parâmetro já existente:

```
DEBIAN_SNORT_HOME_NET="127.0.0.0/16,192.168.0.0/24"
```

Interface usada:

```
DEBIAN_SNORT_INTERFACE="eth0"
```

Vamos dizer para o SNORT usar o banco de dados do MySQL, editando o arquivo /etc/snort/database.conf e coloque :

```
output database: log, mysql, user=snort password=snort_pwd dbname=snort host=localhost
```

Remova o arquivo de "pendência" do banco de dados para o snort:

```
# rm -rf /etc/snort/db-pending-config
```

Reinicie o SNORT e pronto, já está funcionando e registrando no MySQL:

```
# /etc/init.d/snort restart
```

Instalando o BarnYard2

Baixe o banyard2 e instale:

```
# tar xvf barnyard2-1.8.tar.gz
```

```
# cd barnyard2-1.8
```

```
# ./configure --with-mysql
```

```
# make
# make install
# cd etc
# cp barnyard2.conf /etc/snort
```

Vamos configurar o arquivo `/etc/snort/database.conf`. Comente com “#” a linha que diz respeito ao MySQL:

```
#output database: log, mysql, user=snort password=senha_snort_mysql dbname=snort
host=localhost
```

Adicione a linha:

```
output unified2: filename snort.out, limit 128
```

Agora vamos editar o arquivo `/etc/snort/barnyard2.conf`:

```
config daemon
config hostname: localhost
config interface: eth0
config logdir: /var/log/barnyard2/
config waldo_file: /var/barnyard2/waldo
```

Acrescente no final a linha:

```
output database: log, mysql, user=snort password=senha_snort_mysql dbname=snort host=localhost
```

Vamos criar as pastas para o uso do barnyard2:

```
# mkdir /var/log/barnyard2
# mkdir /var/barnyard2
```

Vamos reiniciar o SNORT:

```
# /etc/init.d/snort restart
```

Iniciando o BarnYard2:

```
# cd /etc/snort
# barnyard2 -d /var/log/snort -f snort.out
```

*** Os comandos acima para inicialização do BarnYard2 deverão ser executados sempre na inicialização do sistema, sugiro colocá-los no `/etc/rc.local` ***

Instalando o Apache2

Instale o apache e alguns requisitos:

```
# aptitude install apache2 apache2-prefork-dev libapr1-dev libaprutil1-dev libopenssl-ruby libcurl4-openssl-dev gem
```

Instalando o SNORBY

Vamos instalar os requisitos para o Snorby:

```
# aptitude install gcc g++ build-essential libssl-dev libreadline5-dev zlib1g-dev linux-headers-generic libsqlite3-dev libxslt-dev libxml2-dev imagemagick git-core libmagickwand-dev default-jre
```

Instale o wkhtmltopdf:

```
# wget http://wkhtmltopdf.googlecode.com/files/wkhtmltopdf-0.10.0\_rc2-static-i386.tar.bz2
# tar xvjf wkhtmltopdf-0.10.0_rc2-static-i386.tar.bz2
# cp wkhtmltopdf-i386 /usr/sbin/wkhtmltopdf
```

Instalando o ruby:

```
# wget http://ftp.ruby-lang.org/pub/ruby/1.9/ruby-1.9.2-p290.tar.gz
# tar -xvzf ruby-1.9.2-p290.tar.gz
# cd ruby-1.9.2-p290
# ./configure
# make && make install
```

Vemos a versão instalada:

```
# ruby -v  
ruby 1.9.2p290 (2010-08-18 revision 29036) [i686-linux]
```

Instalamos los gems

```
# gem install thor i18n bundler  
# gem install tzinfo builder memcache-client rack rack-test erubis mail text-format sqlite3-ruby  
# gem install rack-mount --version=0.4.0  
# gem install rails --version=3.0.0
```

Instalando Snorby:

```
# git clone http://github.com/Snorby/snorby.git /var/www/snorby
```

Editamos o arquivo de configuração do banco de dados:

```
# vi /var/www/snorby/config/database.yml  
snorby: &snorby  
adapter: mysql  
username: snort  
password: senha_snort_mysql  
host: localhost
```

Editamos o arquivo de configuração do Snorby:

```
# vi /var/www/snorby/config/snorby_config.yml
```

development:

```
domain: localhost:3000  
wkhtmltopdf: /usr/sbin/wkhtmltopdf
```

test:

```
domain: localhost:3000  
wkhtmltopdf: /usr/sbin/wkhtmltopdf
```

production:

```
domain: localhost:3000  
wkhtmltopdf: /usr/sbin/wkhtmltopdf
```

Já no shell do MySQL:

```
> GRANT all privileges ON snorby.* TO snort@localhost;  
> flush privileges;  
> quit
```

Instalando o Snorby:

```
# cd /var/www/snorby  
# bundle install  
# rake snorby:setup RAILS_ENV=production
```

Vamos criar o acesso do usuário snort no banco de dados do Snorby:

```
# mysql -u root -p
```

Edite o arquivo /etc/snort/barnyard2.conf e acrescente a linha:

```
output database: log, mysql, user=snort password=senha_snort_mysql dbname=snorby  
host=localhost
```

Editando o Apache2:

Instalando o módulo passenger:

```
# gem install --no-ri --no-rdoc --version 3.0.3 passenger
```

```
# cd /usr/local/lib/ruby/gems/1.9.1/gems/passenger-3.0.3/bin
```

```
# ./passenger-install-apache2-module -a
```

```
The Apache 2 module was successfully installed.
```

Please edit your Apache configuration file, and add these lines:

```
LoadModule passenger_module /usr/local/lib/ruby/gems/1.9.1/gems/passenger-3.0.3/ext/apache2/mod_passenger.so
PassengerRoot /usr/local/lib/ruby/gems/1.9.1/gems/passenger-3.0.3
PassengerRuby /usr/local/bin/ruby
```

After you restart Apache, you are ready to deploy any number of Ruby on Rails applications on Apache, without any further Ruby on Rails-specific configuration!

Press ENTER to continue.

Deploying a Ruby on Rails application: an example

Suppose you have a Rails application in `/somewhere`. Add a virtual host to your Apache configuration file and set its `DocumentRoot` to `/somewhere/public`:

```
<VirtualHost *:80>
  ServerName www.yourhost.com
  DocumentRoot /somewhere/public # <-- be sure to point to 'public'!
  <Directory /somewhere/public>
    AllowOverride all # <-- relax Apache security settings
    Options -MultiViews # <-- MultiViews must be turned off
  </Directory>
</VirtualHost>
```

And that's it! You may also want to check the Users Guide for security and optimization tips, troubleshooting and other useful information:

```
/usr/local/lib/ruby/gems/1.9.1/gems/passenger-3.0.3/doc/Users\_guide\_Apache.html
```

Enjoy Phusion Passenger, a product of Phusion (www.phusion.nl) :-)
<http://www.modrails.com/>

Phusion Passenger is a trademark of Hongli Lai & Ninh Bui.

Vamos criar o arquivo do módulo para o Apache2:

```
# vi /etc/apache2/mods-available/passenger.load
```

Adicione no arquivo:

```
LoadModule passenger_module /usr/local/lib/ruby/gems/1.9.1/gems/passenger-3.0.3/ext/apache2/mod_passenger.so
```

Vamos habilitar o módulo:

```
# vi /etc/apache2/mods-available/passenger.conf
```

Adicione no arquivo:

```
<IfModule mod_passenger.c>
  PassengerRoot /usr/local/lib/ruby/gems/1.9.1/gems/passenger-3.0.3
  PassengerRuby /usr/local/bin/ruby
</IfModule>
```

Instalando o módulo:

```
# a2enmod passenger
```

```
# a2enmod rewrite
```

```
# a2enmod ssl
```

Reinicie o apache2:

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

Edite o arquivo `/etc/apache2/sites-available/default`:

```
<Virtualhost *:80>
```

```
# ServerName infosegura
DocumentRoot /var/www/snorby/public/

<Directory /var/www/snorby/public>
  RailsBaseURI /
  Options +ExecCGI FollowSymLinks -MultiViews
  AllowOverride all
  Order allow,deny
  Allow from all
</Directory>
```

Reinicie o apache2:
/etc/init.d/apache2 restart

Vamos reiniciar o SNORT:
/etc/init.d/snort restart

Iniciando o Snorby:
cd /var/www/snorby
ruby script/delayed_job start -e production

*** Os comandos acima para inicialização do Snorby deverão ser executados sempre na inicialização do sistema, sugiro colocá-los no /etc/rc.local ***

Acesse pelo navegador o Snorby:
http://<ip_snorby>

Coloque o usuário e senha:
Usuário: **snorby@snorby.org**
Senha: **snorby**

Snorby
"All About Smplicity"

Please log in to continue...

Email
example@example.com

Password
P@ssw0rd

Welcome, Login! Forgot Password? Remember me

http://www.snorby.org © 2011 Dustin Willis Webber

Tela principal, onde o “Dashboard” é atualizado a cada 30 min e os eventos são atualizados imediatamente:

Verifique se o Snorby Worker, está funcionando clicando em “Administration > Administrator Menu > Worker & Job Queue”

**** DICA **:**

Crie o script abaixo para executar o Barnyard2 na inicialização do sistema:

```
#!/bin/sh
```

```
case $1 in  

  start)
```

```
  printf "starting $0..."
```

```
  cd /etc/snort
```

```
  barnyard2 -d /var/log/snort -f snort.out
```

```

    echo -e 'OK'
;;
stop)
    printf "stopping $0..."
    killall barnyard2
    echo -e 'OK'
;;
restart)
    $0 stop
    $0 start
;;
*)
    echo "usage: $0 (start|stop|restart)"
;;
esac

```

Para executar acrescente no /etc/rc.local:
/caminho/nome_script_barnyard2 start

Para executar o Snorby na inicialização do sistema, crie o script:
#!/bin/bash
cd /var/www/snorby
/usr/local/bin/ruby script/delayed_job start -e production

Para executar acrescente no /etc/rc.local:
/caminho/nome_script_snorby

Atualizar o Snorby

O projeto Snorby está em constante desenvolvimento, por isso se quiser atualizar o seu, é necessário o RAKE:

```
# gem install rake
```

Entre no diretório de instalação do Snorby:
cd /var/www/snorby

Vamos utilizar o GIT para a atualização. Diga a ele que seus arquivos estão nesse diretório...

```
# git add .  
# git commit -m 'Coloque seu comentário aqui'
```

Verifica as atualizações disponíveis:
git rm Gemfile.lock && git pull origin master

Atualize as dependências:
gem update --system
bundle install --path vendor/cache

Edite o arquivo com as credenciais de acesso ao banco de dados MySQL e coloque o usuário e senha de acesso ao banco do Snorby:
vi config/database.yml

Atualize o Snorby:
rake snorby:update

Acesse pelo navegador o Snorby:
http://<ip_snorby>

Referências:

<http://muchikon.blogspot.com/2011/03/snorby-225-en-debian-wheezy.html>

https://lswiki.byu.edu/index.php/Snort,_Barnyard2,_and_Snorby

<http://www.vivaolinux.com.br/artigo/Debian-Sarge-+-Snort-+-MySQL-+-Acidlab-+-Apache/>