# GNU/Linux Administration - Feature #269

## Adding Snorby Frontend for Snort IDS

12/24/2013 02:03 AM - Daniel Curtis

| | | | |
|---|---|---|---|
| **Status:** | Closed | **Start date:** | 12/23/2013 |
| **Priority:** | Normal | **Due date:** | |
| **Assignee:** | Daniel Curtis | **% Done:** | 100% |
| **Category:** | Intrusion Detection/Prevention | **Estimated time:** | 2.00 hours |
| **Target version:** | | **Spent time:** | 9.00 hours |

**Description**

I have setup Snort on my router, however due to the flash based media for its OS there are constraints on log files size. The snort package on pfSense support Barnyard2, which is a MySQL interface to allow logs or alerts to be stored on a MySQL database, which is where Snorby comes in.

> Snorby brings your existing and new network securits monitoring data to life with a suite of beautiful, relevant, and, most importantly, actionable metrics. Share data like sensor activity comparisons or your most active signatures directly with your constituents with daily, weekly, monthly, and ad-hoc PDF reports.

Snorby requires a LAMP stack with Ruby and Passenger installed, I have the LAMP stack already installed, however I will include the the packages in this tutorial to be comprehensive.

As a first step we're going to install Snort. Luckily it's up in the repos, so we're just going to apt-get it. I'm going to go with the snort-mysql package, as it gives a mysql DB support to snort which is a good thing.

First let's get a mysql server up and running:

```
apt-get update
apt-get upgrade
apt-get install mysql-server mysql-client
```

Then we can get snorby up:

```
apt-get install snort-mysql
```

This is needed to for a SQL schema file

This will ask a few questions and it doesn't matter what you answer as we'll have to reconfigure it after Snorby has been installed anyway.

Moving on to installing the Snorby prerequisites:

```
apt-get install libyaml-dev git-core default-jre imagemagick libmagickwand-dev wkhtmltopdf build-e
ssential libssl-dev libreadline-gplv2-dev zlib1g-dev <linux-headers-686-pae> libsqlite3-dev libxsl
t1-dev libxml2-dev libmysqlclient-dev libmysql++-dev apache2-prefork-dev libcurl4-openssl-dev ruby
 ruby-dev
```

Don't forget to use the linux headers for your kernel's architecture…

Instal a few prerequisite gems:

```
gem install bundler rails
gem install rake --version=0.9.2
```

Switch to the web directory:

```
cd /var/www/
```

Download the source for the application.

```
git clone http://github.com/Snorby/snorby.git
```

Change to the Snorby config directory:

```
cd /var/www/snorby/config/
```

Set up configuration files:

```
cp database.yml.example database.yml
cp snorby_config.yml.example snorby_config.yml
sed -i s/"\/usr\/local\/bin\/wkhtmltopdf"/"\/usr\/bin\/wkhtmltopdf"/g snorby_config.yml
```

Create snort database and user:

```
mysql -u root -p

CREATE DATABASE snort;
GRANT ALL PRIVILEGES TO 'snort'@'localhost' IDENTIFIED BY 'SuperSecretPassword';
EXIT
```

Tell snorby the mysql database name, user and password that it should use.

```
nano database.yml
```

At this point you should also create the user and the database.

Change into the Snorby directory:

```
cd /var/www/snorby/
```

Let's install it:

```
bundle install --deployment
bundle exec rake snorby:setup
```

I encountered an error during this part, after googling a bit I found that a stale Gemfile.lock was the culprit and to solve it I needed to remove the .bundle directory then bundle install:

```
cd /var/www/snorby
rm -rf .bundle
bundle install
bundle exec rake snorby:setup
```

I also encountered a problem where I had to set the time in the config/snorby_config.yml

A third error was encountered where there was a dependency problem where bundler needed a version of activesupport that was not installed, to fix this I ran:

```
bundle update activesupport railties rails
gem install arel
gem install ezprint
bundle install
bundle exec rake snorby:setup
```

Yet another problem was encuntered during the installation:

rake aborted!
uninitialized constant Syck::Syck

Tasks: TOP => snorby:setup => environment

The fix was actually simple as well. Make sure the database in the database.yml file matches the database created earlier.

At this point Snorby should start when you type:

```
bundle exec rails server -e production -b 127.0.0.1
```

If you point your browser to
http://localhost:3000/
the Snorby WebUI should pop up.
You can access it with the default credentials:

- snorby@snorby.org
- snorby

Don't be stupid, c+hange the **email** and the **password** after logging in+.

Now if you look around the site you'll notice that Snorby isn't getting any data just yet. So we'll have to configure Snort!

### ALT VPS Method

Since ALT uses a pfSense firewall, snort is installed there (at a cost to performance). And as such snort and barnyard must be configured on the pfSense firewall by going to Service -> Snort -> Edit Interface -> barnyard2

### Original method

Now if you look around the site you'll notice that Snorby isn't getting any data just yet. So we'll have to configure Snort:

```
dpkg-reconfigure snort-mysql
```

Answer the questions, set up all the interface you need for sniffing network traffic and enter Snorby's mysql database and the username and password for it when prompted. Now that the database is configured we'll just need to move away a lock file, so that Snort can start up.

```
mv /etc/snort/db-pending-config /etc/snort/db-pending-config_no_more
```

At this point we're ready to launch snort:

```
service snort start
```

## Let's test it!

Snort should alert for nmap scans so on another box just type:

```
nmap -A -T5 yourhost.org
```

Let it run, then check Snorby. Now there's really only one thing left before we're done.

## Make Snorby autostart rails web server on boot

```
cd /etc/init.d/
nano snorby
```

A simple script like this should do the trick:

```
#!/bin/bash

cd /var/www/snorby && bundle exec rails server -e production &
```

Let's put it to start in runlevel 2:

```
chmod +x snorby
update-rc.d -f snorby start 2
```

And now Snorby will start whenever the system enters runlevel 2 and we're done.

### History

#### #1 - 12/24/2013 04:17 PM - Daniel Curtis

*- Description updated*

#### #2 - 12/25/2013 01:53 AM - Daniel Curtis

*- Description updated*

*- Status changed from New to In Progress*

*- % Done changed from 100 to 80*

#### #3 - 12/25/2013 07:10 AM - Daniel Curtis

*- Description updated*

*- % Done changed from 80 to 90*

#### #4 - 12/27/2013 10:44 AM - Daniel Curtis

*- Status changed from In Progress to Closed*

*- % Done changed from 90 to 100*

#### #5 - 12/30/2013 02:25 PM - Daniel Curtis

I have finally got Snorby along side Puppet Master, I needed to upgrade to Phusion Passenger 4, then specify the version of Ruby used on the Puppet Master. However I came across a problem trying to get Snorby to run on Apache with Passenger, and the above method will not automatically start Snorby at boot. The problem was that Passenger could not find the necessary gems to run the application, according to the Passenger documentation this can be fixed by adding a system user and changing the ownership of the snorby folder to the snorby system user:

```
sudo adduser --system snorby
sudo adduser snorby www-data
chown -R /var/www/snorby
```

From there I had to locally install the gems needed to the snorby folder:

```
cd /var/www/snorby
bundle install --path vendor/bundle
```

Now that the snorby folder is owned and built for the snorby user it works.

#### #6 - 02/16/2015 02:14 PM - Daniel Curtis

*- Project changed from 57 to GNU/Linux Administration*

*- Category set to Intrusion Detection/Prevention*