CENTRAL LOGS - SYSLOG-NG

Recently I did an article on using the *syslog* program to create a centralized logging infrastructure (see here). It works, but it does have some drawbacks; all the entries sit in one file, it can only use UDP, and some others. But there is a program called *syslog-ng* (homepage here), which aims to address many of these problems. Lets look at how we can adapt a current centralized *syslog* infrastructure, to make use of *syslog-ng* as the centralized node.

What do I need?
You will of course need to download *syslog-ng*..

```
wget http://www.balabit.com/downloads/syslog-ng/2.0/src/syslog-ng-2.0rc1.tar.gz
```

You will also need two dependencies (specific to the version I am using above)..

```
wget http://www.balabit.com/downloads/syslog-ng/2.0/src/eventlog-0.2.5.tar.gz
wget ftp://ftp.gtk.org/pub/glib/2.12/glib-2.12.3.tar.gz
```

Then we do the installations..

```
tar -xvvf glib-2.12.3.tar.gz
cd ./glib-2.12.3
./configure
make
make install
cd ..

tar -xzvf eventlog-0.2.5.tar.gz
cd ./eventlog-0.2.5
./configure
make
make install
cd ..

tar -xzvf syslog-ng-2.0rc1.tar.gz
cd syslog-ng-2.0
export PKG_CONFIG_PATH=/usr/local/lib/pkgconfig
./configure
make
make install
```

Setup the server
Firstly, remember that the default install places the *syslog-ng.conf* file in the */usr/local/etc* folder, so we will be editing that. Next, bear in mind that we are setting this up to replace the *syslog* server as the central node..

```
###This is to setup global options
options {
  create_dirs(yes);
  sync(0);
  time_reopen (10);
  log_fifo_size (1000);
  long_hostnames(off);
  use_dns (yes);
  use_fqdn (no);
  keep_hostname (yes);
        };

###This setups logs from the syslog-ng daemon
source l_int_src { internal(); };
###This setups the local syslog logs
source l_sys_src { unix-stream("/dev/log"); };
###This setups the local kernel logs
source l_krn_src { file("/proc/kmsg"); };
###This setups the server is listen for any normal syslog messages
source r_udp_src { udp( ip (x.x.x.x) port (514)); };

###This splits the logs recieved into hostname and date
destination r_messages {
```

```
 file("/var/log/hosts/$HOST/$YEAR.$MONTH.$DAY/messages");
};

###This just puts all processed logs into a "messages" file
log { source(l_krn_src); source(l_sys_src); source(r_udp_src); destination(r_messages);
};
```

Starting up

Ok, so we have installed the software and setup the configuration file. Now lets get *syslog-ng* to startup when the machine starts, go to the folder where you extracted *syslog-ng*, and go into the *contrib* folder. Inside there you will find a bunch of files named *init.d.<something>*, take the appropriate one and copy it to your startup script folder. Once thats done then just remove *syslog* from starting up when your server starts.

Final Words

Syslog-ng is a lot more powerful then the above setup, it can filter, read from all sorts of files/pipes/etc and many other nice things. It is well worth having a look around and playing with it. But nonetheless, we have accomplished the goal of replacing our *syslog* central node with a *syslog-ng* central node. As always, have fun and learn.