cituser@logger2:~$ sudo apt-get update
cituser@logger2:~$ sudo apt-get install syslog-ng syslog-ng-core
note the exact same steps above are done regardless of if you are on the logserver or the client machine

note the exact same steps above are done regardless of if you are on the logserver or the client machine

cituser@logger2:~$
Now for specific server configuration
cituser@logger2:/etc/syslog-ng$ echo "global settings and some good information is in the syslog-ng.conf file"
global settings and some good information is in the syslog-ng.conf file
cituser@logger2:/etc/syslog-ng$
cituser@logger2:/etc/syslog-ng/conf.d$ echo "We will put our config files here"
We will put our config files here
cituser@logger2:/etc/syslog-ng/conf.d$
cituser@logger2:~/etc/syslog-ng/conf.d$ sudo vi testhost.conf
# testhost.conf file
# These options below will override the generic options for this one log
options {
    create_dirs(yes);
    owner(root);
    group(root);
    perm(0664);
    dir_owner(root);
    dir_group(root);
    dir_perm(0755);
};

# Here we are creating a source for our logs that will
# come from udp port 514
# This source should only show up once
# It might be better to put in the global conf file
source is a reserved word s_udp is not
source s_udp {
    udp(port(514));
};
udp(port(514));

#we can specify filters
#this one will make sure that we are only dealing with logs from
#this one host
#filter is a reserved word f_web is not
filter f_web {
    host("144.38.220.211");
};

#where should we write the logs to?
#note the special variables available to us
destination d_web {
    file("/var/log/web/$YEAR/$MONTH/$YEAR-$MONTH-$DAY.web.log");
};

#these lines put them all together
log {
    source(s_udp);
    filter(f_web);
    destination(d_web);
}
cituser@logger2:/etc/syslog-ng/conf.d$ sudo service syslog-ng restart
Jan 14 06:44:39 logger2 syslog-ng[2213]: Configuration reload request received, reloading configuration;
Jan 14 06:44:39 logger2 syslog-ng[2213]: EOF on control channel, closing connection;
Jan 14 06:44:39 logger2 syslog-ng[2213]: Configuration reload request received, reloading configuration;
Jan 14 06:44:39 logger2 syslog-ng[2213]: EOF on control channel, closing connection;
Jan 14 07:17:02 logger2 CRON[2541]: (root) CMD ( cd / && run-parts --report /etc/cron.hourly)
Jan 14 08:17:01 logger2 CRON[2545]: (root) CMD ( cd / && run-parts --report /etc/cron.hourly)
Jan 14 08:58:31 logger2 syslog-ng[2213]: syslog-ng shutting down; version='3.5.3'
Jan 14 08:58:31 logger2 syslog-ng[2585]: syslog-ng starting up; version='3.5.3'
Jan 14 08:58:32 logger2 syslog-ng[2585]: EOF on control channel, closing connection;
cituser@ubuntu:~$ echo "now we are on the client machine"
now we are on the client machine
cituser@ubuntu:~$
$ sudo apt-get install syslog-ng syslog-ng-core
[sudo] password for cituser:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libdbi1 libesmtp6 libeventlog0 libhiredis0.10 libibv0kis0 libmongoclient0
  libnet1 syslog-ng-mod-amqp syslog-ng-mod-geoip syslog-ng-mod-json
  syslog-ng-mod-mongodb syslog-ng-mod-redis syslog-ng-mod-smtp
  syslog-ng-mod-sql syslog-ng-mod-stomp
Suggested packages:
  rabbitmq-server mongodb-server libdbd-mysql libdbd-pgsql libdbd-sqlite3
  activemq
The following packages will be REMOVED:
  rsyslog ubuntu-minimal
The following NEW packages will be installed:
  libdbi1 libesmtp6 libeventlog0 libhiredis0.10 libibv0kis0 libmongoclient0
  libnet1 syslog-ng syslog-ng-core syslog-ng-mod-amqp syslog-ng-mod-geoip
  syslog-ng-mod-json syslog-ng-mod-mongodb syslog-ng-mod-redis
  syslog-ng-mod-smtp syslog-ng-mod-sql syslog-ng-mod-stomp
0 upgraded, 17 newly installed, 2 to remove and 3 not upgraded.
Need to get 893 kB of archives.
After this operation, 1,828 kB of additional disk space will be used.
Do you want to continue? [Y/n]
cituser@ubuntu:/etc/syslog-ng/conf.d$ sudo vi myhost.conf
Define a new source that essentially 'tails' the apache logs

Send the logs off to a remote logging server (if required)
# make sure your ip and port match
destination loghost { udp("144.38.220.35" port(514)); };

#note that s_src is defined in the global options
log { source(s_src); destination(loghost); };
sudo service syslog-ng restart
cituser@ubuntu:/etc/syslog-ng/conf.d$ logger "This command sends a message to the log file"

cituser@ubuntu:/etc/syslog-ng/conf.d$
Jan 14 09:33:46 144.38.220.34 cituser: hello from over here
Jan 14 09:33:51 144.38.220.34 sudo:  cituser : TTY=pts/0 ; PWD=/etc/syslog-ng/conf.d ; USER=root ; COMMAND=/usr/sbin/service syslog-ng restart
Jan 14 09:33:51 144.38.220.34 sudo: pam_unix(sudo:session): session opened for user root by cituser(uid=0)
Jan 14 09:33:51 144.38.220.34 syslog-ng[3724]: syslog-ng shutting down; version= '3.5.3'
Jan 14 09:33:51 144.38.220.34 syslog-ng[3761]: syslog-ng starting up; version='3.5.3'
Jan 14 09:33:52 144.38.220.34 syslog-ng[3761]: EOF on control channel, closing connection;
Jan 14 09:33:52 144.38.220.34 sudo: pam_unix(sudo:session): session closed for user root
Jan 14 09:33:54 144.38.220.34 cituser: hello from over here
Jan 14 09:34:41 144.38.220.34 cituser: hello from over here
Jan 14 09:34:42 144.38.220.34 cituser: hello from over here
Jan 14 09:34:42 144.38.220.34 cituser: hello from over here
Jan 14 09:34:42 144.38.220.34 cituser: hello from over here
Jan 14 09:34:42 144.38.220.34 cituser: hello from over here
Jan 14 09:34:43 144.38.220.34 cituser: hello from over here
Jan 14 09:34:43 144.38.220.34 cituser: hello from over here
Jan 14 09:34:43 144.38.220.34 cituser: hello from over here
Jan 14 09:35:04 144.38.220.34 cituser: This command sends a message to the log file
Now let's parse those logs so that we only see cron messages in a file.
#just_cron.conf

These options below will override the generic options for this one log options {
    create_dirs(yes);
    owner(root);
    group(root);
    perm(0664);
    dir_owner(root);
    dir_group(root);
    dir_perm(0755);
};

filter f_crona {
    host("144.38.220.34") and match("CRON");
};

destination d_crona {
    file("/var/log/web/$YEAR/$MONTH/$YEAR-$MONTH-$DAY.cron.log");
};

log {
    source(s_udp);
    "just_cron.conf" 26L, 520C
}
create_dirs(yes);
owner(root);
group(root);
perm(0664);
dir_owner(root);
dir_group(root);
dir_perm(0755);
}

filter f_crona {
    host("144.38.220.34") and match("CRON");
};

destination d_crona {
    file("/var/log/web/$YEAR/$MONTH/$YEAR-$MONTH-$DAY.cron.log");
};

log {
    source(s_udp);
    filter(f_crona);
    destination(d_crona);
};
don't forget to restart the syslog daemon
From client create a simple cronjob that runs every minute
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line indicating with different fields when the task will be run and what command to run for the task
#
# To define the time you can provide concrete values for minute (m), hour (h), day of month (dom), month (mon), and day of week (dow) or use '*' in these fields (for 'any').
# Notice that tasks will be started based on the cron's system daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow command
* * * * * echo "this is a hello message that runs every minute" >> /dev/null

"/tmp/crontab.PLLk36/crontab" 23L, 965C
23,1 All
cituser@logger2: /etc/syslog-ng/conf.d$ echo "View logs in correct location"
View logs in correct location
cituser@logger2: /etc/syslog-ng/conf.d$ tail -f /var/log/web/2016/01/2016-01-14.cron.log