Syslog

Specifics

- uses port 514 TCP and 514 UDP (default)

Installation

```
$ apt install rsyslog
```

Check that it is running

```
$ netstat -ano
```

Configure

```
vi /etc/rsyslog.conf
```

Disable/Enable by commenting out or uncommenting the modules imudp and intcp

```
# /etc/rsyslog.conf configuration file for rsyslog
#
# For more information install rsyslog-doc and see
# /usr/share/doc/rsyslog-doc/html/configuration/index.html
#
# Default logging rules can be found in /etc/rsyslog.d/50-default.conf


#################
#### MODULES ####
#################

module(load="imuxsock") # provides support for local system logging
#module(load="immark")  # provides --MARK-- message capability

# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
module(load="imtcp")
input(type="imtcp" port="514")
```

This configuration file also allows you to specify where log types go...

<MORE TO COME>

Forward logs to another service

```
*.* @127.0.0.1:514
```

Testing Rsyslog

To listen on a port:

Figure out your interface

```
$ ifconfig

docker0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        inet6 fe80::42:d5ff:fe84:1047  prefixlen 64  scopeid 0x20<link>
        ether 02:42:d5:84:10:47  txqueuelen 0  (Ethernet)
        RX packets 56436034  bytes 78842926492 (78.8 GB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 73681312  bytes 26355844162 (26.3 GB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enp0s25: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.50  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::6e3b:e5ff:fe41:582b  prefixlen 64  scopeid 0x20<link>
        ether 6c:3b:e5:41:58:2b  txqueuelen 1000  (Ethernet)
        RX packets 123047743  bytes 50439393846 (50.4 GB)
        RX errors 0  dropped 451010  overruns 0  frame 0
        TX packets 142608496  bytes 115546425420 (115.5 GB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 20  memory 0xf7f00000-f7f20000

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 2350821  bytes 433332792 (433.3 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2350821  bytes 433332792 (433.3 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

...
```

```
$ tcpdump -i enp0s25 port 514
```

If you have syslog running in a docker container on your would use the following command to see the data

```
$ tcpdump -i docker0 port 514
13:24:44.758302 IP deepthought.57574 > 172.17.0.16.syslog: SYSLOG user.notice, length: 123
```

Send a log entry

```
$ logger --server localhost --port 514 "this is a test"
```

Start and Stopping the RSyslogd Service

Get Status

```
$ systemctl status rsyslog

 rsyslog.service - System Logging Service
     Loaded: loaded (/lib/systemd/system/rsyslog.service; disabled; vendor preset: enabled)
     Active: active (running) since Thu 2021-07-22 13:34:41 EDT; 15s ago
       Docs: man:rsyslogd(8)
             https://www.rsyslog.com/doc/
   Main PID: 305322 (rsyslogd)
      Tasks: 10 (limit: 19045)
     Memory: 6.3M
     CGroup: /system.slice/rsyslog.service
             305322 /usr/sbin/rsyslogd -n -iNONE

Jul 22 13:34:41 deepthought systemd[1]: Starting System Logging Service...
Jul 22 13:34:41 deepthought systemd[1]: Started System Logging Service.
Jul 22 13:34:41 deepthought rsyslogd[305322]: rsyslogd's groupid changed to 110
Jul 22 13:34:41 deepthought rsyslogd[305322]: rsyslogd's userid changed to 104
Jul 22 13:34:41 deepthought rsyslogd[305322]: [origin software="rsyslogd" swVersion="8.2001.0" x-pid="305322" x-
info="https://www.rsyslog.c>
```

Restart

```
$ systemctl restart rsyslog
```

Forwarding to Elasticsearch

Install rsyslog-elasticsearch

```
$ apt-get install rsyslog-elasticsearch
```

Add config file for elasticsearch

```
$ vi /etc/rsyslog.d/10-rsyslog-elasticsearch.conf
```

```
module(load="omelasticsearch") # for outputting to Elasticsearch
# this is for index names to be like: logstash-YYYY.MM.wWW (where WW is the week number)
template(name="logstash-index"
  type="list") {
    constant(value="logstash-")
    property(name="timereported" dateFormat="rfc3339" position.from="1" position.to="4")
    constant(value=".")
    property(name="timereported" dateFormat="rfc3339" position.from="6" position.to="7")
    constant(value=".w")
    # here we use the week number to avoid creating lots of shards on Elasticsearch
    property(name="timereported" dateFormat="week")
}


# permits to have the part after `/` in programname
global(parser.permitSlashInProgramName="on")

# this is for formatting our syslog in JSON with @timestamp
template(name="plain-syslog"
  type="list") {
    constant(value="{")
      constant(value="\"@timestamp\":\"")     property(name="timereported" dateFormat="rfc3339")
      constant(value="\",\"host\":\"")        property(name="hostname")
      constant(value="\",\"severity\":\"")    property(name="syslogseverity-text")
      constant(value="\",\"facility\":\"")    property(name="syslogfacility-text")
      constant(value="\",\"programname\":\"")   property(name="programname" format="json")
      constant(value="\",\"procid\":\"")   property(name="procid" format="json")
      constant(value="\",\"message\":\"")    property(name="msg" format="json")
    constant(value="\"}")
}
# this is where we actually send the logs to Elasticsearch
action(server="127.0.0.1"
    serverport="9200"
    usehttps="off"
    type="omelasticsearch"
    template="plain-syslog"
    errorfile="/tmp/rsyslog-elasticsearch-error.log"
    searchIndex="logstash-index"
    dynSearchIndex="on")
```

Restart

```
$ systemctl restart rsyslog
```

Check Status

```
$ systemctl status rsyslog
```