

syslog-ng

an alternative syslog replacement

David Morgan

© David Morgan 2006

reasons

- syslog accepts from network from all-or-nothing
- multi-hop forwards sourced to most recent hop
- messages are in cleartext
- configuration is inflexible
- uses connectionless UDP

© David Morgan 2006

syslog-ng vs syslog

- syslog-ng runs instead of syslog
 - # service syslog stop
 - # service syslog-ng start
- apps are unaffected
 - they follow the same API to emit messages
 - syslog-ng picks them up if so configured
 - and disposes of them the same way if configured

© David Morgan 2006

Config elements to define

- source <name> { <some available sources> }
- destination <name> { <some available destinations> }
- log { source(<name>); destination(<name>); }

“log” marries named sources and destinations, causes messages that emanate from the former to go to the latter

a “message route”

© David Morgan 2006

Example sources

- file – from a file (or pseudofile, eg kernel messages)
- udp or tcp – from a specified port
- pipe – from a fifo/named pipe
- unix-stream – messages from a process or device
- internal – messages from syslog-ng itself

© David Morgan 2006

Example destinations

- file – messages deposited in a specified file
- usertty – to a specified user's terminal
- udp or tcp – to a specified network socket (host/port)
- pipe – to a fifo/named pipe
- program – to standard input of a process it launches
- unix-stream – to a process or device

© David Morgan 2006

One more element: filter

- added to a message route's sources & destinations
- selects only certain of the sources' messages for actual dispatch to the destinations

© David Morgan 2006

Example filter functions

- facility – select by syslog-type message facility
- priority – by syslog message priority
- program – by message's program name field
- host – by message's hostname field
- match – by message itself
- netmask – by whether sender IP's in specified subnet

© David Morgan 2006

Logging to a network loghost

- Make the client emit to the network

- include udp or tcp in a destination
- log something to it

```
destination loghost {udp("192.168.3.12" port(514));};  
log {source(s_sys); destination(loghost);};
```

- Make the loghost/server pick up from the network

- include udp or tcp in a source
- put messages from that source where you want

```
source <name> {  
    ...  
    udp(ip(0.0.0.0) port(514));  
};  
log {source(<name>); destination(<desired>);};
```

© David Morgan 2006

Extension: multi-hop loghosting

- Make the client define udp and/or tcp in *both*

- a source definition, for incoming
- a destination definition, for outgoing
- a log path to marry and activate them

- Host field preserves id of host-of-origin

- syslog reflects only last, not original, host in the chain
- syslog-ng on a centralized loghost records true origins

© David Morgan 2006

Multi-hop loghosting

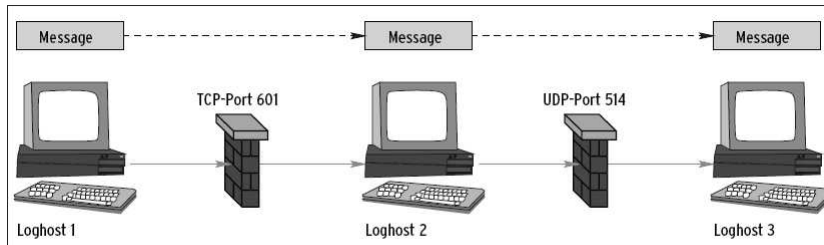


Figure 2: Depending on the configuration, Syslog-NG uses either TCP or UDP to forward messages across the network. This allows you to customize Syslog-NG for use in segmented environments with multiple firewalls

Linux Magazine, "Flight Recording Box," Christian Schmitz, December 2003

© David Morgan 2006

info

- http://www.balabit.com/products/syslog_ng/
- http://www.balabit.com/products/syslog_ng/reference-1.6/syslog-ng.html/index.html
- <http://www.linux-magazine.com/issue/37/Syslog-NG.pdf>
- [/usr/share/doc/syslog-ng-xxx/syslog-ng-conf.doc](#)

© David Morgan 2006