### sendsyslog-tcp

Converts incoming records to syslog messages and sends them to the destination syslog server via TCP communication.

#### Syntax

sendsyslog-tcp dst=IP\_ADDR [framing=lf|rfc6587] [format=json|txt] [port=INT] [pri=INT]

Required Parameter

**dst=IP\_ADDR**

IP address of the syslog server

Optional Parameters

**format=json|txt**

Format in which logs will be sent (Either json or txt). The default is txt.

* json: Converts all input data to JSON format before transmitting.
* txt: Transmits the raw string value of the **line** field as is.

**framing=lf|rfc6587**

How to determine where each Syslog message ends. This ensures that the receiving server correctly interprets the end of one message and the start of the next. The default is lf.

* lf: Uses a newline character (LF) to mark the end of each Syslog message.
* rfc6587: Prefixes each Syslog message with its length in bytes, according to [RFC 6587](https://tools.ietf.org/html/rfc6587).

**port=INT**

Listening port number of the syslog server (default: 514). Any integer between 1 and 65535.

**pri=INT**

PRI value defined in RFC 5424: <https://tools.ietf.org/html/rfc5424> (default: 134).

The PRI value is constructed by multiplying the Facility by 8 and then adding the Level. For example, **local0.info**, meaning facility=16 and level=6, becomes **134**." You can calculate the PRI value by using the appropriate values for Facility and Severity as specified in RFC 5424.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Facility(↓)<br />Severity(→) | 0/Emer | 1/Alert | 2/Crit | 3/Error | 4/Warn | 5/Notice | 6/Info | 7/Debug |
| 0 / kern | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 / user | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 2 / mail | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
| 3 / daemon | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| 4 / auth | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 |
| 5 / syslog | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 |
| 6 / lpr | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 |
| 7 / news | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 |
| 8 / uucp | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 |
| 9 / clock | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 |
| 10 / authpriv | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 |
| 11 / ftp | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 |
| 12 / ntp | 96 | 97 | 98 | 99 | 100 | 101 | 102 | 103 |
| 13 / audit | 104 | 105 | 106 | 107 | 108 | 109 | 110 | 111 |
| 14 / alert | 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 |
| 15 / solaris-cron | 120 | 121 | 122 | 123 | 124 | 125 | 126 | 127 |
| 16 / local0 | 128 | 129 | 130 | 131 | 132 | 133 | **134** (default) | 135 |
| 17 / local1 | 136 | 137 | 138 | 139 | 140 | 141 | 142 | 143 |
| 18 / local2 | 144 | 145 | 146 | 147 | 148 | 149 | 150 | 151 |
| 19 / local3 | 152 | 153 | 154 | 155 | 156 | 157 | 158 | 159 |
| 20 / local4 | 160 | 161 | 162 | 163 | 164 | 165 | 166 | 167 |
| 21 / local5 | 168 | 169 | 170 | 171 | 172 | 173 | 174 | 175 |
| 22 / local6 | 176 | 177 | 178 | 179 | 180 | 181 | 182 | 183 |
| 23 / local7 | 184 | 185 | 186 | 187 | 188 | 189 | 190 | 191 |

#### Description

The sendsyslog-tcp command transmits Syslog messages over TCP, providing reliable message delivery compared to UDP, which can suffer from IP packet fragmentation. However, using TCP also introduces the potential for delays or loss of connection to the Syslog server. If the connection is interrupted or delayed, the queue of unsent Syslog messages may become full. To prevent the buildup of unsent messages, the sendsyslog-tcp command discards any messages that remain unsent for more than 30 seconds.

To manage the Syslog message sending queue, Logpresso provides several switches within its shell:

|  |  |  |
| --- | --- | --- |
| Switch | Description | Value |
| logpresso.tcp\_sender.idle\_timeout | Maximum duration to wait for a server response before stopping the transmission. | 1 to 86,400 seconds (default: 300) |
| logpresso.tcp\_sender.max\_queue\_time | Maximum time messages can stay in the queue before being discarded. | 1 to 600 seconds (default: 30) |
| logpresso.tcp\_sender.max\_queue\_chars | Maximum number of characters allowed in the queue before stopping the transmission. | 1,000,000 to 1,000,000,000 characters (default: 100,000,000 (about 10MB)) |

To apply these switches, run the following commands in the Logpresso shell:

set logpresso.tcp\_sender.idle\_timeout=300set logpresso.tcp\_sender.max\_queue\_time=30set logpresso.tcp\_sender.max\_queue\_chars=100000000

Alternatively, you can add these switches through a configuration file such as config.sh or logpresso.sh:

JAVA\_OPTS="$JAVA\_OPTS -Dlogpresso.tcp\_sender.idle\_timeout=300"JAVA\_OPTS="$JAVA\_OPTS -Dlogpresso.tcp\_sender.max\_queue\_time=30"JAVA\_OPTS="$JAVA\_OPTS -Dlogpresso.tcp\_sender.max\_queue\_chars=100000000"