Essentials

We support receiving syslog messages from any application, as long as they comply to either RFC-3164 or RFC-5424 (and RFC-5425 for TLS). The destination host is **logsene-receiver-syslog.sematext.com** and ports we use are:

- for Syslog over UDP: **514**
- for Syslog over TCP: **514**
- for Syslog over TLS: **10514** (get root certificate and intermediate certificate to get TLS working)
- for RELP: **20514**

Authorization

There are two ways to authorize when you send logs. Authorizing means telling Logsene which Logsene App to send logs to. We recommend you embed your Logsene App token in your syslog daemon's config in a CEE-formatted JSON message. Step-by-step instructions for rsyslog and syslog-ng, and a raw example are below.

Alternatively, authorize your public IPs and then send messages directly. Note that configuring your log shipper to send your Logsene App token is preferred to authorizing source IPs. You can see specific instructions for rsyslog, syslog-ng and syslogd for how to forward messages in this case.

Example

A quick way to ship messages via TCP syslog is with netcat:

```
echo 'my-host my-process:@cee: {"logsene-app-token": "LOGSENE_APP_TOKEN_GOES_HERE", "message"
```

Ways to Ship Logs

In production, you're probably going to use a syslog daemon. Details on configuring syslog daemons to send logs over TCP/UDP/RELP are below:

- rsyslog
- syslog-ng
- traditional syslogd

TLS Encryption

In addition to TCP, UDP and RELP, Logsene also supports RFC-5425 compliant Syslog over TLS. See instructions for rsyslog and syslog-ng on how to configure them.

HTTP or HTTPS

If you use a recent version of rsyslog (6.4.0 or later), you might want to send logs directly to Logsene's Elasticsearch API, through the omelasticsearch module. Details on how to do that are on the rsyslog howto page.