Logstash to LA Connector

Last updated by | Naser Mohammed | Mar 19, 2023 at 7:18 AM CDT

Logstash basics

Introduction

- 1. Download log stash: https://www.elastic.co/downloads/logstash ☑ (Sentinel Supports only version 7.x.x)
- 2. Install Logstash:
- Linux: https://www.elastic.co/guide/en/logstash/current/installing-logstash.html#installing-logstash
- Windows: http://www.dissmeyer.com/2017/11/11/installing-logstash-on-windows/ Z
- 3. Install the log analytics plug in: https://github.com/yokawasa/logstash-output-azure_loganalytics https://github.com/yokawasa/logstash-output-azure_loganalytics
- 4. Create a conf file in bin (logstash folder location in linux is/usr/share/logstash), here is an example:

File Plugin: https://www.elastic.co/guide/en/logstash/current/plugins-inputs-file.html#plugins-inputs-file-start_position

□

- a. Add the workspace configuration (workspace Id and Key)
- b. Point the log file (json format)
- c. Add the correct columns
- d. Add the table name in LA
- e. Define TimeGenerated column

Here is an example for conf file:Logstash conf file example

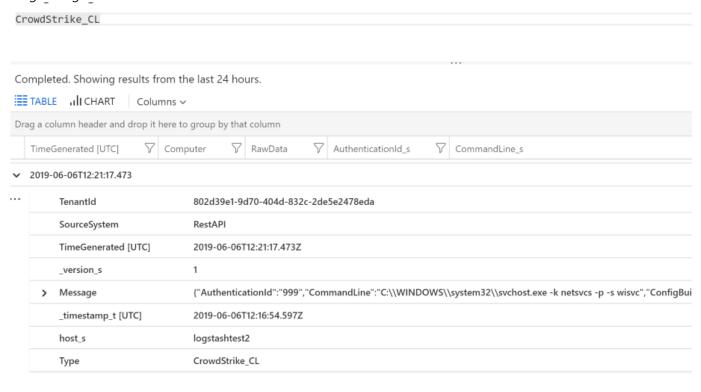
1. You will find the logs in the tale under CustomLogs

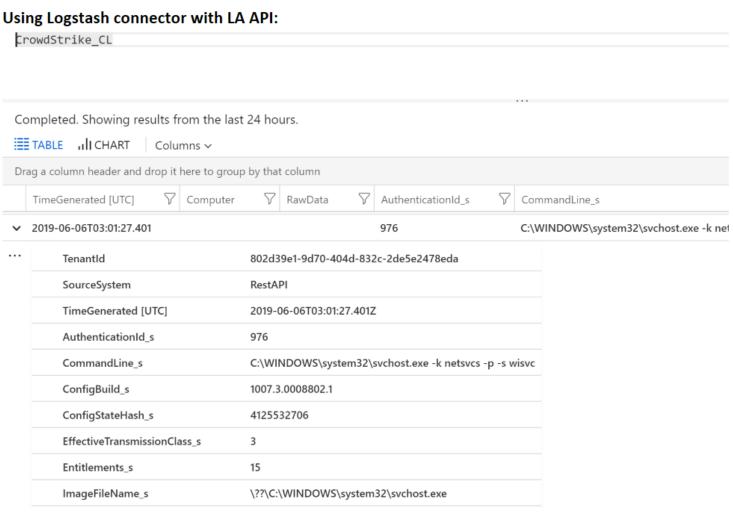
Some helpful links:

Date Plugin: https://www.elastic.co/quide/en/logstash/current/plugins-filters-date.html

Json Plugin: https://www.elastic.co/guide/en/logstash/current/plugins-filters-json.html ref

Using LA Agent:





Suggested commands:

```
java -version
sudo apt install default-jre
sudo apt install default-jdk
Download Logstash: <a href="https://www.elastic.co/downloads/logstash">https://www.elastic.co/downloads/logstash</a> □
Install:
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch ☑ | sudo apt-key add -
sudo apt-get install apt-transport-https
echo "deb https://artifacts.elastic.co/packages/7.x/apt \(\mathrix\) stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-
7.x.list
sudo apt-get update && sudo apt-get install logstash
Navigate /usr/share/logstash/bin -Run the following
  ./logstash-plugin install microsoft-logstash-output-azure-loganalytics
Create a conf file in bin --> /usr/share/logstash/bin/ (logstash folder location in linux is /usr/share/logstash)
  Cat >test.conf
  Vi test.conf
  ((Contents of conf file: https://docs.microsoft.com/en-us/azure/sentinel/connect-logstash#sample-configuratio
  input {
         tcp {
             port => "514"
             type => syslog #optional, will effect log type in table
```

workspace_key => "Z9DDT9p00QsrZjbr71FfoQqpVWdUquwRdmrviNKMj0ekpqjZt9z7Yyj+u5RLA9uuvqwiF+8CJY+hY0VH++lo

workspace_id => "7e31edf1-4929-449d-b5f4-274a447891d3" # <your workspace id>

Press escape and ":wq!"

filter {

output {

microsoft-logstash-output-azure-loganalytics {

custom log table name => "Testtable CL"

Run

Change the default TimeGenerated behaviour

In case we need to override the default TimeGenerated field in Log Analytics we can specify a new field for it.

In the following example we are:

- 1. Taking a test log in /var/log/test.log
- 2. Parsing the new timestamp value from the log string using a regex with grok and a Json string
- 3. Parsing the extracted timestamp string to be in the ISO 8601 format (YYYY-MM-DDThh:mm:ssZ)
- 4. Exporting this new field
- 5. Assigning the new field to the time_generated_field variable plus selecting what fields do we want to log

```
# take an input
input {
    file {
       path => "/var/log/test.log"
filter {
# extract the header timestamp and the Json section
    grok {
        match => {
            "message" => ["^(?<timestamp>.{24}):\s(?<json data>.*)$"]
# parse the extracted header as a timestamp
date {
    id => 'parse_metric_timestamp'
        match => [ 'timestamp', 'EEE MMM dd HH:mm:ss yyyy' ]
        timezone => 'Europe/Rome'
        target => 'custom_time_generated'
    }
json {
    source => "json_data"
# output to a file for debugging (optional)
output {
    file {
        path => "/tmp/test.txt"
        codec => Line { format => "custom format: %{message} %{custom_time_generated} %{json_data}"}
    }
# output to the console output for debugging (optional)
output {
    stdout { codec => rubydebug }
# log into Log Analytics
output {
    microsoft-logstash-output-azure-loganalytics {
        workspace_id => '[REDACTED]'
        workspace_key => '[REDACTED]
        custom_log_table_name => 'RSyslogMetrics'
        time_generated_field => 'custom_time_generated'
        key_names => ['custom_time_generated','name','origin','sender','messages']
    }
}
```

Example of /var/log/test.log

 $\textit{Mon Nov 07 20:45:08 2022: } \{ \textit{"name":"_custom_time_generated", "origin":"test_microsoft", \textit{"sender":"test@microsoft", "sender":"test@microsoft", "sender":"test_microsoft", "sender":"test_microott, "sender":"test_microsoft", "sender":"test_microsoft", "sende$

Edgie Enabled