Expected Windows process not running

Many critical IT applications and services running on Windows operating systems run as a process. You want to detect when an expected process is not found in the process list on the host so you can proactively manage potential stability issues.

Data required

Technologies:

• Splunk Enterprise or Splunk Cloud Platform and Splunk Add-on for Microsoft Windows OR
• Splunk Infrastructure Monitoring and the Splunk OpenTelemetry Connector

Data:

• Windows event logs
• Windows update logs

Procedure

Option 1

1. In Splunk Enterprise or Splunk Cloud Platform, verify that you deployed the Splunk Add-on for Microsoft Windows add-on to your search heads, indexer, and Splunk Universal Forwarders on the monitored systems. For more information, see About installing Splunk add-ons.
2. Run the following search. You can optimize it by specifying an index and adjusting the time range.

```splunk
|mstats count WHERE index="<name of Windows metrics index>" AND metric_name=Process.* host="<name of host to check>" BY host, instance span=15m
| rename instance AS process
| eval expected_process_list=mvappend("<name of process to check>", "<name of process to check>")
| eval expected_process_count="<total number of processes expected per host>"
| eval expected_process_regex="(?i)".mvjoin(expected_process_list, "}")
| eval expected_process_found=if(match(process, expected_process_regex),1,0)
| stats values(expected_process_list) AS expected_processes values(expected_process_count) AS expected_process_count
| values(eval(if(expected_process_found>0,process,null()))) AS processes_found sum(expected_process_found) AS processes_found_count BY _time host
| eval count_of_missing_processes=expected_process_count - processes_found_count
| dedup host
| rename expected_processes AS "Expected Processes", expected_process_count AS "# of Expected Processes per Host", processes_found_count AS "# of Expected Processes"
```

The information provided in Splunk Lantern is intended for informational and educational purposes only. All information is provided in good faith, however, Splunk disclaims any and all representations and warranties, express and implied, regarding the information provided, including without limitation any warranties and representations regarding the completeness, adequacy or accuracy of the information. You agree to take full responsibility for the results arising from the use of the information provided.
**Search explanation**

The table provides an explanation of what each part of this search achieves. You can adjust this query based on the specifics of your environment.

<table>
<thead>
<tr>
<th>Splunk Search</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>`</td>
<td>mstats count WHERE index=”&lt;name of Windows metrics index&gt;” AND metric_name=Process.* host=”&lt;name of host to check&gt;” BY host, instance span=15m`</td>
</tr>
<tr>
<td>`</td>
<td>rename instance AS process`</td>
</tr>
<tr>
<td>`</td>
<td>eval expected_process_list=mvappend(&quot;&lt;name of process to check&gt;&quot;, &quot;&lt;name of process to check&gt;&quot;)`</td>
</tr>
<tr>
<td>`</td>
<td>eval expected_process_count=&quot;&lt;total number of processes expected per host&gt;&quot;`</td>
</tr>
<tr>
<td>`</td>
<td>eval expected_process_regex=&quot;(?i)&quot;.mvjoin(expected_processes_list, &quot;&quot;)`</td>
</tr>
<tr>
<td>`</td>
<td>eval expected_process_found=if(match(process, expected_process_regex),1,0)`</td>
</tr>
<tr>
<td>`</td>
<td>stats values(expected_process_list) AS expected_processes values(expected_process_count) AS expected_process_count values(eval(if(expected_process_found&gt;0,process,null( ))) AS processes_found_count sum(expected_process_found) AS processes_found_count BY _time host`</td>
</tr>
<tr>
<td>`</td>
<td>eval count_of_missing_processes=expected_process_count - processes_found_count`</td>
</tr>
<tr>
<td>`</td>
<td>dedup host`</td>
</tr>
<tr>
<td>`</td>
<td>rename expected_processes AS &quot;Expected Processes&quot;, expected_process_count AS &quot;# of Expected Processes per Host&quot;, processes_found_count AS &quot;# of Expected Processes&quot;`</td>
</tr>
</tbody>
</table>
### Splunk Search

<table>
<thead>
<tr>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>processes_found AS &quot;Expected Processes Found on Host&quot;, processes_found_count AS &quot;# of Expected Process Found on Host&quot;, count_of_missing_processes AS &quot;Expected Processes Missing&quot;</td>
</tr>
</tbody>
</table>

### Next steps

The Expected Processes Missing field indicates the total number of processes expected but missing from the most recent host process data. Any positive number indicates one or more expected processes missing. Zero indicates the number of running processes matches what is expected. A negative number indicates that a higher number of processes were found than expected.

To alert when an expected Windows process is not running, you can configure one of the following two recommendations:

- Use the SPL from this procedure to configure a Core Splunk alert.
- Build a new Vital Metric in IT Essentials Work for the desired entity type and configure vital metric alerting. After the vital metric has been created, configure it to alert when the number of expected processes not running is greater than zero.

Finally, you might be interested in other processes associated with the Maintaining Microsoft Windows systems use case.

### Option 2

1. Ensure that you have the Splunk OTEL Collector installed on the host you want to monitor.
2. Update the receivers section of the OTEL agent config file on the host to collect procstat metrics for each process.
   
   ```
   ... receivers: ...
   #The following config will collect process metrics for all processes. You can adjust the pattern parameter to filter down to a subset of processes
   smartagent/procstat:
   type: telegraf/procstat
   pattern: ".*"
   ```

3. Update the services.pipelines.metrics.receivers section of the OTEL agent config file to include the procstat receiver.
   
   ```
   ... service:
   extensions: ...
   pipelines:
   traces:
   ... 
   ```

The information provided in Splunk Lantern is intended for informational and educational purposes only. All information is provided in good faith, however, Splunk disclaims any and all representations and warranties, express and implied, regarding the information provided, including without limitation any warranties and representations regarding the completeness, adequacy or accuracy of the information. You agree to take full responsibility for the results arising from the use of the information provided.
metrics:
  receivers: [..., smartagent/procstat]
...

4. In Splunk Infrastructure Monitoring, use the following SignalFlow to search the procstat.cpu_usage streaming metric, filter down to the desired hosts and processes, and summarize results by counting the total number of processes found per host.

\[
A = \text{data('procstat.cpu_usage', filter=filter('host.name', '<name of host to check>') and filter('process_name', '<name of process to check>')).count(by=['host.name']).publish(label='A')}
\]

Next steps

To alert when no process data is flowing in for the selected host(s) and process(es), use the SignalFlow from this procedure to configure a detector with an alert condition of "Heartbeat" and alert settings of 15 minutes.

Finally, you might be interested in other processes associated with the Maintaining Microsoft Windows systems use case.