Following best practices for metrics ingestion

Your environment has hundreds of independently developed services, immutable infrastructure, and frequent code pushes. You want to get the most valuable and efficient information out of your metrics monitoring, so you need information on best practices for data submission and problem detection.

For infrastructure metrics, use the SignalFx Smart Agent where possible

The Smart Agent is the recommended mechanism for submitting metrics for common infrastructure and services, in part due to support of the Smart Agent, and in part due to the pairing of those submitted metrics with built-in dashboards and other visualizations. If you use other agents or integrations, you will need to create your own dashboards from scratch.

Limit how sparse your metrics are

The Splunk Infrastructure Monitoring metric system as a whole - from ingestion and processing to analysis and problem detection - is optimized to work with time series data. This is data about services or resources that exhibit some trend over time, and ideally, report on a regular and frequent basis. There are a number of mechanisms that make the Splunk Infrastructure Monitoring system more resilient than other metrics-based monitoring and alerting systems to issues caused by sparseness or aperiodicity, but require that users proactively make use of them.

Do not use reserved terms in metric names, dimensions, properties, or tags

Splunk Infrastructure Monitoring reserves the prefix of sf_, sf.num and aws_. If you use these in custom metric names, dimension names or property names, the associated metric datapoints will not be ingested and not be available for use in any way.

Use consistent signal types

For a detector to work properly, the signal that is evaluated should represent a consistent type of measurement. This is the normal state of affairs when you choose a metric; for example, `cpu.utilization` as reported by the collectd agent is a value between 0 and 100 and represents the average utilization across all CPU cores for a single Linux instance or host.

Use caution with wildcards in naming

If you use wildcards in your metric name, you should make sure that the wildcards do not mistakenly include metrics of different types. For example, if you enter `jvm.*` as the metric name, this could cause your detector to evaluate `jvm.heap`, `jvm.uptime`, or `jvm.cpu.load` (assuming those are all metric names in use in your organization).
against the same threshold, which may lead to unexpected results.

**Next steps**

These additional Splunk resources might help you understand and implement these recommendations:

- Splunk Docs: [Migrate from SignalFx Smart Agent to the Splunk Distribution of OpenTelemetry Collector](#)
- Splunk Docs: [Metric name standards](#)
- Splunk Docs: [Metrics, dimensions, and metric time series](#)