

CheatSheet: Prometheus

CLOUD

- PDF Link: [cheatsheet-prometheus-A4.pdf](#), Category: Cloud
- Blog URL: <https://cheatsheet.dennyzhang.com/cheatsheet-prometheus-A4>
- Related posts: Nagios CheatSheet, Kubectrl CheatSheet, #denny-cheatsheets

File me Issues or star this repo.

1.1 Prometheus Commands

| Name | Command |
|---|---|
| Run prometheus server with docker | <code>docker run -p 9090:9090 prom/prometheus, http://localhost:9090/graph,</code> |
| Run cadvisor to get local containers' metrics | <code>docker run -v /var/run:/var/run -v /sys:/sys -p 8080:8080 google/cadvisor,</code> |
| Query metrics by api, instead of web console | <code>curl http://localhost:9090/api/v1/query?query=container_memory_usage{instance="localhost:9090"}</code> |
| List all alerts of alertmanager | <code>curl http://localhost:9093/api/v1/alerts</code> |
| Prometheus tech stack footprint | prometheus(350MB RAM), node-exporter(10MB), kube-state-metrics(20MB), alertmanager(10MB) |
| Example of client libraries | Link: prometheus-python-example.py |
| Prometheus Online Demo | Live demo from CloudAlchemy |
| Prometheus Config file | <code>/etc/prometheus/prometheus.yml</code> Sections in conf: <code>global</code> , <code>rule_files</code> , <code>scrape_configs</code> |

1.2 Prometheus Components

| Name | Command |
|---------------------------|---|
| Prometheus server | Scrapes and store time series data. It uses mainly pull model, instead of push. |
| Special-purpose exporters | Get metrics for all kinds of services. e.g, Node Exporter, Blackbox Exporter, SNMP Exporter, JMX Exporter |
| Client libraries | Instrument application code. |
| Alertmanager | Handle alerts. |
| Push gateway | Support short-lived jobs. Persist the most recent push of metrics from batch jobs. |
| Reference | Link: Exporters And Integrations , Link: Default port allocations |

<https://raw.githubusercontent.com/dennyzhang/cheatsheet-prometheus-A4/master/prometheus-overview.png>

1.3 Prometheus Metric Types

| Name | Command |
|-----------|---|
| Counter | It only goes up (and resets), counts something. e.g, the number of requests served, tasks completed, or errors. |
| Gauge | It goes up and down, snapshot of state. e.g, temperatures or current memory usage, etc |
| Summary | It samples observations, especially over a sliding time window. e.g, <code>rate(http_request_duration_seconds_sum[5m])</code> |
| Histogram | It samples observations and counts them in configurable buckets. |

1.4 Prometheus Concepts

| Name | Summary |
|-----------|--|
| Target | A target is the definition of an object to scrape. |
| Job | A collection of targets with the same purpose. |
| Instance | A label that uniquely identifies a target in a job. |
| Exporter | Expose metrics from a non-Prometheus format into a format Prometheus supports. |
| Collector | A part of an exporter that represents a set of metrics. |
| Handler | |
| Rule | |

1.5 Kubernetes Metrics Targets & Samples

| Name | Command | Sample Metrics |
|--------------------------------|---|-----------------------------------|
| cadvisor | http://\protect\T1\textdollarip:10255/metrics/cadvisor | Link: cadvisor-sa |
| node-exporter | http://\protect\T1\textdollarip:9100/metrics | Link: node-expor |
| kubelet | http://\protect\T1\textdollarip:10255/metrics | Link: kubelet-sa |
| kube-dns | http://\protect\T1\textdollarip:10054/metrics | Link: kube-dns-sa |
| kube-state-metrics http-metric | http://\protect\T1\textdollarip:8080/metrics | Link: kube-state- |
| kube-state-metrics telemetry | http://\protect\T1\textdollarip:8081/metrics | Link: kube-state- |
| apiserver | https://\protect\T1\textdollarapi_server:443/metrics | |

<https://raw.githubusercontent.com/dennyzhang/cheatsheet-prometheus-A4/master/prometheus-deployment.png>

1.6 Prometheus PromQL Query

| Name | Command |
|--|---|
| Reference | Link: query |
| Find metric by name+job+group | <code>somemetric{job="prometheus",group="canary"}</code> |
| The avg network traffic received per second, over the last min | <code>rate(apiserver_request_count{verb="GET",code="200"}[1m])</code> |
| topk query | <code>rate(node_network_receive_bytes_total[1m])</code> |
| join | Link: query-topk.txt |
| cut | |
| slice | |
| count | |
| predict | |
| sum | |
| min | |
| max | |
| avg | |

1.7 Prometheus Alerts

| Name | Command |
|--|---------|
| How full will the disks be in 4 hours? | |
| Which services are the top 5 users of CPU? | |
| What's the 95th percentile latency in EU datacenter? | |

1.8 More Resources

License: Code is licensed under MIT License.

<https://prometheus.io/>

<https://povilasv.me/prometheus-tracking-request-duration/>