

巡检工具数据统计

- K8S
- ES
- kafka
- codis
- Mysql
- host

K8S

资源类型	巡检内容	实现方式
k8s	获取指定时间间隔内Pod重启事件（个）	prometheus上获取， query = "max_over_time(sum by(container) ((kube_pod_container_status_last_terminated_reason{container=~"%s"} == 1 and on(pod) rate (kube_pod_container_status_restarts_total[1m]) * 60 >= 1) or on() vector(0))[%s])" % (sdu, interval)
k8s	获取POD在指定时间间隔内的最大出向网络流量（MB/s）	prometheus上获取， query = "max_over_time(max(sum by(pod_name) (pod:pod_network_transmit_bytes_total:sum_rate{pod_name=~"%s.*"}))[%s])" % (sdu,interval)
k8s	获取POD在指定时间间隔内的最大入向网络流量（MB/s）	prometheus上获取， query = "max_over_time(max(sum by(pod_name) (pod:pod_network_receive_bytes_total:sum_rate{pod_name=~"%s.*"}))[%s])" % (sdu,interval)
k8s	获取POD在指定时间间隔内的最大MEM使用率（%）	prometheus上获取， query = "max(max_over_time((sum(container_memory_rss{pod=~"%s.", container=""}) by (pod) / sum(kube_pod_container_resource_limits_memory_bytes{pod=~"%s."}) by (pod))[%s]))" % (sdu,sdu,interval)
k8s	获取POD在指定时间间隔内的平均MEM使用率（%）	prometheus上获取， query = "avg(avg_over_time((sum(container_memory_rss{pod=~"%s.", container=""}) by (pod) / sum(kube_pod_container_resource_limits_memory_bytes{pod=~"%s."}) by (pod))[%s]))" % (sdu,sdu,interval)
k8s	获取POD在指定时间间隔内的平均CPU使用率（%）	prometheus上获取， query = "avg(avg_over_time((sum(rate(container_cpu_usage_seconds_total{pod=~"%s.", container=""}){2m}) by (pod) / sum (kube_pod_container_resource_limits{pod=~"%s.", resource="cpu"} by (pod))[%s]))" % (sdu,sdu,interval)
k8s	获取正常POD数	prometheus上获取， query = "job:k8s_abnormal_pod{mon_env="live", mon_cluster="%s", mon_mod="%s", mon_cid="%s"} or on() vector(0)" % (cluster, project_module, cid)
k8s	获取POD在指定时间间隔内的最大CPU使用率（%）	prometheus上获取， query = "max(max_over_time((sum(rate(container_cpu_usage_seconds_total{pod=~"%s.", container=""}){2m}) by (pod) / sum (kube_pod_container_resource_limits{pod=~"%s.", resource="cpu"} by (pod))[%s]))" % (sdu,sdu,interval)

ES

资源类型	巡检内容	实现方式
ES	获取ES集群状态，“1”为正常，“0”为不正常	prometheus上获取， query = "elasticsearch_cluster_health_status{cluster="%s"} == 1"(cluster)
ES	获取ES集群在指定时间间隔内的平均查询请求速率（K/s）	prometheus上获取， query = "avg(avg_over_time(rate(elasticsearch_indices_search_query_total {cluster=~"%s"}[1m])[%s:1m]))" (cluster, interval)
ES	获取ES集群在指定时间间隔内的平均索引请求速率（K/s）	prometheus上获取， query = "avg(avg_over_time(rate(elasticsearch_indices_indexing_index_total {cluster=~"%s"}[1m])[%s:1m]))" (cluster, interval)

ES	获取ES集群在指定时间间隔内的平均索引延迟 (ms)	prometheus上获取, query = "avg(avg_over_time(rate(elasticsearch_indices_indexing_index_time_seconds_total{cluster=~\"%s\"}[1m])/rate(elasticsearch_indices_indexing_index_total{cluster=~\"%s\"}[1m]))\"%(cluster, cluster, interval)"
ES	获取ES集群在指定时间间隔内的平均查询延迟 (ms)	prometheus上获取, query = "avg(avg_over_time(rate(elasticsearch_indices_search_query_time_seconds{cluster=~\"%s\"}[1m]) / rate(elasticsearch_indices_search_query_total{cluster=~\"%s\"}[1m]))\"%(cluster, cluster, interval)"
ES	获取ES集群在指定时间间隔内的最大写IO (MB/s)	prometheus上获取, query = "max(max_over_time(rate(elasticsearch_filesystem_io_stats_device_write_size_kilobytes_sum{cluster=~\"%s\"}[1m]))\"%(cluster, interval)"
ES	获取ES集群在指定时间间隔内的最大读IO (MB/s)	prometheus上获取, query = "max(max_over_time(rate(elasticsearch_filesystem_io_stats_device_read_size_kilobytes_sum{cluster=~\"%s\"}[1m]))\"%(cluster, interval)"
ES	获取ES集群在指定时间间隔内的最大磁盘使用率 (%)	prometheus上获取, query = "max(max_over_time((1 - elasticsearch_filesystem_data_available_bytes{cluster=~\"%s\"} / elasticsearch_filesystem_data_size_bytes{cluster=~\"%s\"})\"%(cluster, cluster, interval)"
ES	获取ES集群在指定时间间隔内的平均堆内存使用率 (%)	prometheus上获取, query = "avg(avg_over_time((elasticsearch_jvm_memory_used_bytes{cluster=~\"%s\"} / elasticsearch_jvm_memory_max_bytes{cluster=~\"%s\"})\"%(cluster, cluster, interval)"
ES	获取ES集群在指定时间间隔内的最大堆内存使用率 (%)	prometheus上获取, query = "max(max_over_time((elasticsearch_jvm_memory_used_bytes{cluster=~\"%s\"} / elasticsearch_jvm_memory_max_bytes{cluster=~\"%s\"})\"%(cluster, cluster, interval)"
ES	获取ES集群在指定时间间隔内的最大CPU使用率 (%)	prometheus上获取, query = "max(max_over_time(elasticsearch_os_cpu_percent{cluster=~\"%s\"})\"%(cluster, interval)"
ES	获取ES集群在指定时间间隔内的平均CPU使用率 (%)	prometheus上获取, query = "avg(avg_over_time(elasticsearch_os_cpu_percent{cluster=~\"%s\"})\"%(cluster, interval)"
ES	大促播报ES巡检: 过去一小时最大CPU (%) ; RPS/WPS (K/s) ; 水位值 (%) ;	prometheus上获取, query = "max(max_over_time(elasticsearch_os_cpu_percent{cluster=~\"%s\"})\"%(cluster, interval)"
ES	获取ES集群在指定时间间隔内的最大网络IO TX (MB/s)	prometheus上获取, query = "max(max_over_time(rate(elasticsearch_transport_tx_size_bytes_total{cluster=~\"%s\"}[1m]))\"%(cluster, interval)"
ES	获取ES集群在指定时间间隔内的最大网络IO RX (MB/s)	prometheus上获取, query = "max(max_over_time(rate(elasticsearch_transport_rx_size_bytes_total{cluster=~\"%s\"}[1m]))\"%(cluster, interval)"
ES	获取ES集群Data Node数量	prometheus上获取, query = "count(count(elasticsearch_nodes_roles{cluster=~\"%s\", role=\"data\"}) by (name))\"%(cluster)"

kafka

资源类型	巡检内容	实现方式
kafka	获取kafka集群的离线分区数量	prometheus上获取, query = "sum by(env) (kafka_controller_kafkacontroller_offlinepartitionscount{env=\"live\"})"
kafka	获取Kafka集群在指定时间间隔内某个消费组的平均消费延迟	prometheus上获取, query = "max(avg_over_time(kafka_consumergroup_lag{consumer_group=~\"%s.*\",cluster_name=\"foody_kafka_live\"})\"%(consumer_group,interval)"
kafka	获取Kafka集群目前brokers数量, 正常为3	prometheus上获取, query = "kafka_brokers{env=\"live\"}"
kafka	获取Kafka集群当前ActiveController数量, 正常为1	prometheus上获取, query = "sum by(env) (kafka_controller_kafkacontroller_activecontrollercount{env=\"live\"})"
kafka	获取Kafka集群指定时间间隔内的最大JVM mem使用率	prometheus上获取, query = "max(max_over_time(jvm_memory_bytes_used{area=\"heap\",env=\"live\"}) / jvm_memory_bytes_max{area=\"heap\",env=\"live\"} * 100)\"%(interval)"
kafka	获取Kafka集群在指定时间间隔内的最大CPU使用率 (%)	prometheus上获取, query = "max_over_time(max(1-min((avg without(cpu,mode)(irate(node_cpu_seconds_total{mon_proj=\"%s\",mode=\"idle\"}[2m]))>0))\"%(cluster, interval)"

kafka	获取Kafka集群在指定时间间隔内的单实例最大IO读写流量（MB/s）	prometheus上获取， query = ""max(max_over_time(sum by(instance)(irate (node_disk_written_bytes_total{mon_proj="%s"}[2m]))[%s:]))""%(cluster, interval)
kafka	获取Kafka集群在指定时间间隔内的最大堆内存使用率（%）	prometheus上获取， query = ""max(max_over_time((sum (java_lang_Memory_HeapMemoryUsage_used{mon_proj="%s"}) by (instance) /sum (java_lang_Memory_HeapMemoryUsage_max{mon_proj="%s"}) by (instance))[%s:]))""%(cluster, cluster, interval)
kafka	获取Kafka集群在指定时间间隔内的平均请求响应时延（ms）	prometheus上获取， query = ""avg(avg_over_time (kmf_services_produce_service_produce_delay_ms_99th{name="%s",mon_middleware_type="kafka-SLI"}[%s:]))""%(cluster, interval)
kafka	获取Kafka集群在指定时间间隔内的平均生产速率（K/s）	prometheus上获取， query = ""avg(avg_over_time(sum(rate(kafka_topic_partition_current_offset {cluster="%s", topic!~".*consumer_offsets"}[1m]))[%s:]))""%(cluster, interval)
kafka	获取Kafka集群在指定时间间隔内的平均消费速率（K/s）	prometheus上获取， query = ""avg(avg_over_time(sum(rate(kafka_consumergroup_current_offset {cluster="%s", topic!~".*consumer_offsets"}[1m]))[%s:]))""%(cluster, interval)
kafka	获取Kafka集群在指定时间间隔内的平均TPS_IN	prometheus上获取， query = ""avg(avg_over_time(sum (kafka_server_BrokerTopicMetrics_OneMinuteRate{mon_proj="%s",name="BytesInPerSec",topic=~".+"}[%s:]))""%(cluster, interval)
kafka	获取Kafka集群在指定时间间隔内的平均TPS_OUT	prometheus上获取， query = ""avg(avg_over_time(sum (kafka_server_BrokerTopicMetrics_OneMinuteRate{mon_proj="%s",name="BytesOutPerSec",topic=~".+"}[%s:]))""%(cluster, interval)
kafka	获取Kafka集群在指定时间内的down机事件，"1"为正常，"0"为不正常（个）	prometheus上获取， query = ""min_over_time(up{cluster=~"%s", instance=~".*:9100"}[%s:]))""%(cluster, interval)
kafka	获取Kafka集群在指定时间内的topic leader change事件（个）	prometheus上获取， query = ""max_over_time(sum(changes(kafka_topic_partition_leader{cluster="%s"}[2m]))[%s:]))""%(cluster, interval)
kafka	获取Kafka集群在指定时间间隔内的平均实例最大IO读写流量（MB/s）	prometheus上获取， query = ""avg(max_over_time(sum by(instance)(irate (node_disk_read_bytes_total{mon_proj="%s"}[2m]))[%s:]))""%(cluster, interval)
kafka	获取Kafka集群在指定时间间隔内的平均实例TPS（MB/s）	prometheus上获取， query = ""avg(avg_over_time(sum by(instance) (kafka_server_BrokerTopicMetrics_OneMinuteRate{mon_proj="%s",name=~"BytesInPerSec BytesInPerSec",topic=~".+"}[%s:]))""%(cluster, interval)
kafka	大促播报Kafka巡检：过去一小时最大CPU（%）；每秒写入消息数（K/s）；每秒写入吞吐量（MB/s）	prometheus上获取， query = ""max_over_time(max(1-min((avg without(cpu,mode)(irate (node_cpu_seconds_total{mon_proj="%s",mode='idle'}[2m]))>0))[%s:]))""%(cluster, interval)

codis

资源类型	巡检内容	实现方式
codis	获取codies的一段时间的最大cpu	prometheus上获取， query = ""max_over_time(max(sum(irate(redis_used_cpu_sys{alias=~'.%s'}[30s])) by (addr) + sum(irate(redis_used_cpu_user{alias=~'.%s'}[30s])) by (addr))by(addr)[%s:]))"" % (cluster_name,cluster_name,interval)
codis	获取codies的一段时间的最大mem	prometheus上获取， query = ""max_over_time(max(redis_memory_used_bytes{job="redis_exporter",alias=~'.%s'}/redis_memory_max_bytes{job="redis_exporter",alias=~'.%s'})by (addr)[%s:]))"" % (cluster_name,cluster_name,interval)
codis	获取redisCluster的一段时间的最大连接数实例的最大连接数	prometheus上获取， query = ""max_over_time(max(redis_connected_clients{alias=~'.%s'})by (addr)[%s:]))"" % (cluster_name,interval)
codis	获取codies最近MASTER实例的qps总和	prometheus上获取， query=""max_over_time(max(sum(irate (redis_command_call_duration_seconds_count{alias=~'.%s',cmd!~"ping info config",addr=~"%s"}[30s])) [%s:]))"" % (cluster_name, addrs, interval)
codis	获取codies查询网络流量IN	prometheus上获取， query = ""max(max_over_time(sum(rate(redis_net_input_bytes_total {alias=~'.%s'}[1m])) by (addr) [%s:]))"" % (cluster_name, interval)

codis	获取codies查询网络流量Out	prometheus上获取, query = "max(max_over_time(sum(rate(redis_net_output_bytes_total{alias=~'.*%s'}[1m])) by (addr)[%s:]))" % (cluster_name, interval)
codis	获取codies ProxyUp	prometheus上获取, query = "codisproxy_up{alias=~'.*%s'}[%s:]" % (cluster_name, interval)
codis	获取codies RedisUp	prometheus上获取, query = "redis_up{alias=~'.*%s'}[%s:]" % (cluster_name, interval)
codis	获取Codis Request Latency	prometheus上获取, query = "max_over_time(max(sum (irate (redis_command_call_duration_seconds_count{alias=~'.*%s',cmd='latency',addr=~'%s'}[30s]))[%s:]))" % (cluster_name, addrs, interval)
codis	获取codies的一段时间的最大连接数实例的最大连接数	prometheus上获取, query = "max_over_time(max(redis_connected_clients{job='redis_exporter', alias=~'.*%s'})by(addr)[%s:]))" % (cluster_name,interval)
codis	获取codies最近单实例最大的qps	prometheus上获取, query="max_over_time(max(sum(irate (redis_command_call_duration_seconds_count{alias=~'.*%s',cmd!~'ping info config'} [1m])) by (addr)[%s:]))" % (cluster_name, interval)
codis	DP大促播报Codis巡检: 过去一小时最大CPU (%) ; 最大QPS (K/s) ;可承载QPS (K/s) ; 水位值 (%) ;	prometheus上获取, query = "max_over_time(max(sum(irate(redis_used_cpu_sys{alias=~'.%s'}[30s])) by (addr) + sum(irate(redis_used_cpu_user{alias=~'.%s'}[30s])) by (addr))by(addr)[%s:]))" % (cluster_name, cluster_name, interval)

Mysql

资源类型	巡检内容	实现方式
Mys ql	获取mysql集群的单实例最大连接数 (个)	prometheus上获取, query = "max(max by(target_instance) (mysql_global_status_threads_connected) * on (target_instance) group_right database_target_relation{db_name='%s'})" % (database)
Mys ql	获取mysql集群最大同步延迟 (s)	prometheus上获取, query = "max(mysql_slave_status_seconds_behind_master * on (target_instance) group_right database_target_relation{db_name='%s'})" % (database)
Mys ql	获取mysql集群的错误连接数 (个)	prometheus上获取, query = "sum(sum by(target_instance)(irate(mysql_global_status_aborted_connects[2m])) * on (target_instance) group_right database_target_relation{db_name='%s'})" % (database)
Mys ql	获取mysql集群在指定时间间隔内的最大链接数	prometheus上获取, query = "max_over_time(sum(mysql_global_status_threads_connected{name=~'.%s.'})[%s:]))" % (database,interval)
Mys ql	获取mysql集群在指定时间间隔内的最大慢查询数	prometheus上获取, query = "max_over_time(sum(rate(mysql_global_status_slow_queries{name=~'%s.'}[5m]))[%s:15s]) or max_over_time(sum(irate(mysql_global_status_slow_queries{name=~'%s.'}[5m]))[%s:15s]))" % (database, interval,database,interval)
Mys ql	获取mysql集群在指定时间间隔内的最大运行线程数	prometheus上获取, query = "max_over_time(sum(mysql_global_status_threads_running{name=~'%s.*'})[%s:15s]))" % (database,interval)

host

资源类型	巡检内容	实现方式
host	获取HOST在指定时间间隔内的最大CPU使用率	prometheus上获取, query = "max_over_time(max(100-(avg(irate(node_cpu_seconds_total{mon_ip=~'%s', mode='idle'}[30s])) by (instance))*100)[%s:]))" % (host_ip,interval)
host	获取HOST在指定时间间隔内的最大内存使用率	prometheus上获取, query = "max_over_time(max(avg((1 - (node_memory_MemAvailable_bytes{mon_ip=~'%s'} / (node_memory_MemTotal_bytes{mon_ip=~'%s'})))* 100)[%s:0m]))" % (host_ip,host_ip,interval)
host	获取HOST的负载 (5 m)	prometheus上获取, query = "avg(node_load5{mon_ip='%s'}) / count(count(node_cpu_seconds_total{mon_ip='%s'}) by (cpu)) * 100" % (host_ip, host_ip)

host	获取HOST在指定时间间隔内的最大timewait数	prometheus上获取， query = ""max_over_time(max(node_sockstat_TCP_tw{mon_ip="%s"})[%s])"" % (host_ip, interval)
host	获取HOST在指定时间间隔内的最大Establish数	prometheus上获取， query = ""max_over_time(max(node_netstat_Tcp_CurrEstab{mon_ip="%s"})[%s])"" % (host_ip, interval)
host	获取过去一段时间内Host的状态，"1"为正常，"0"为不正常	prometheus上获取， query = ""up{mon_ip="%s"}[%s]"" % (host_ip, interval)
host	获取HOST在指定时间间隔内的最大TCP重传数	prometheus上获取， query = ""max_over_time(irate(node_netstat_Tcp_RetransSegs{env="live",instance=~"%s.*"}[1m])[%s])""%(host_ip,interval)
host	获取HOST挂载点当前的最大磁盘使用率	prometheus上获取， query = ""max((100 - ((sum by(instance, mountpoint) (node_filesystem_avail_bytes{instance=~"%s.*"} * 100) / sum by(instance, mountpoint) (node_filesystem_size_bytes{instance=~"%s.*"}))))""%(host_ip,host_ip)
host	获取HOST在指定时间间隔内的最大磁盘IO使用率	prometheus上获取， query = ""max(max_over_time(rate(node_disk_io_time_seconds_total{instance=~"%s.*"}[1m])[%s:]))""%(host_ip,interval)