Support for Diameter Application KPIs and Alerts

Feature Summary and Revision History

Summary Data

Table 1: Summary Data

Applicable Products or Functional Area	PCF
Applicable Platform(s)	SMI
Feature Default Setting	Enabled - Always-on
Related Documentation	Not Applicable

Revision History

Table 2: Revision History

Revision Details	Release
First introduced.	2023.03.0

Feature Description

PCF supports Diameter Application KPI's and Alerts support in parity with PCRF application.

How It Works

This section describes how this feature works.

Statistics

node[x].messages.e2e __[realm_] Gx_CCR-I_2001. qns_stat.success

Description: Success message Policy Director count for return code 2001

node[x].messages.e2e __[realm_] Gx_CCR-I_2001. qns_stat.total _time_in_ms

Description: Total milliseconds Policy Director of successful messages with return code matching 2001

node[x].messages.e2e __[realm_] Gx_CCR-I_3xxx. qns_stat.success

Description: Success count of Policy Director messages with return code matching 3XXX

node[x].messages. e2e__ [realm_] Gx_CCR-I_4xxx. qns_stat.success

Description: Success count of Policy Director messages with return code matching 4XXX

node[x].messages. e2e__ [realm_] Gx_CCR-I_5xxx. qns_stat.success

Description: Success count of Policy Director messages with return code matching 5XXX

node1.counters. [realm_] Gx_CCR-I.qns_count

Description: Count of messages Policy Server (qns) successful sent to the policy engine

node[x].messages. e2e__ [realm_] Gx_CCR-U_2001. qns_stat.success

Description: Success message count for return code 2001

node[x].messages. e2e__ [realm_] Gx_CCR-U_2001. qns_stat.total_ time_in_ms

Description: Total milliseconds of successful messages with return code matching 2001

node[x].messages. e2e__ [realm_] Gx_CCR-U_3xxx. qns_stat.success

Description: Success count of messages with return code matching 3XXX

node[x].messages. e2e__ [realm_] Gx_CCR-U_4xxx. qns_stat.success

Description: Success count of messages with return code matching 4XXX

node[x].messages. e2e__ [realm_] Gx_CCR-U_5xxx. qns_stat.success

Description: Success count of messages with return code matching 5XXX

node1.counters. [realm_] Gx_CCR-U. qns_count

Description: Count of messages Policy Server (qns) successful sent to the policy engine

node[x].messages. e2e__ [realm_] Gx_CCR-U_2001. qns_stat.success

Description: Success message count for return code 2001

node[x].messages. e2e__ [realm_] Gx_CCR-U_2001. qns_stat. total_time_in_ms

Description: Total milliseconds of successful messages with return code matching 2001

node[x].messages. e2e__ [realm_] Gx_CCR-U_3xxx. qns_stat.success

Description: Success count of messages with return code matching 3XXX

node[x].messages. e2e__ [realm_] Gx_CCR-U_4xxx. qns_stat.success

Description: Success count of messages with return code matching 4XXX

node[x].messages. e2e__ [realm_] Gx_CCR-U_5xxx. qns_stat.success

Description: Success count of messages with return code matching 5XXX

node1.counters. [realm_] Gx_CCR-U. qns_count

Description: Count of messages successful sent to the policy engine

node1.counters. [realm_] Gx_CCR-T. qns_count

Description: Success message count for return code 2001

node[x].messages. e2e__ [realm_] Gx_CCR-T_2001. qns_stat.success

Description: Total milliseconds of successful messages with return code matching 2001

node[x].messages.e2e_<domain>_[realm_]Gx_CCR-T_3xxx.qns_stat.success

Description: Success count of messages with return code matching 3XXX

node[x].messages. e2e__ [realm_] Gx_CCR-T_4xxx. qns_stat.success

Description: Success count of messages with return code matching 4XXX

node[x].messages. e2e__ [realm_] Gx_CCR-T_5xxx. qns_stat.success

Description: Success count of messages with return code matching 5XXX

node1.counters. [realm_] Gx_CCR-T.qns_count

Description: Count of messages successful sent to the policy engine

node1.counters. [realm_] Gx_RAR-T. qns_count

Description: Success message count for return code 2001

node[x].messages. e2e__ [realm_] Gx_RAR-T_2001. qns_stat.success

Description: Total milliseconds of successful messages with return code matching 2001

node[x].messages.e2e_<domain>_[realm_]Gx_RAR-T_3xxx.qns_stat.success

Description: Success count of messages with return code matching 3XXX

node[x].messages. e2e__ [realm_] Gx_RAR-T_4xxx. qns_stat.success

Description: Success count of messages with return code matching 4XXX

node[x].messages. e2e__ [realm_] Gx_RAR-T_5xxx. qns_stat.success

Description: Success count of messages with return code matching 5XXX

node[x].messages. e2e__ [realm_] Gx_RAR_timeout. qns_stat.success

Description: Success timeout Policy Director count for RAR message

node1.counters. [realm_] Gx_RAA.qns_count

Description: Count of all messages sent to the policy engine

node1.messages. in_q_Gx_RAA. qns_stat.error

Description: Count of messages failed to be sent to the policy engine

node1.messages. in_q_Gx_RAA. qns_stat.success

Description: Count of messages successful sent to the policy engine

node1.counters. [realm_] Gx_RAR.qns_count

Description: Count of messages successful sent to the Policy Director (LB)

node[x].messages. e2e__ [realm_] Rx_AAR_2001. qns_stat.success

Description: Success message count for return code 2001

node[x].messages. e2e__ [realm_] Rx_AAR_2001. qns_stat.total_ time_in_ms

Description: Total milliseconds of successful messages with return code matching 2001

node[x].messages. e2e__ [realm_] Rx_AAR_3xxx. qns_stat.success

Description: Success count of Policy Director messages with return code matching 3XXX

node[x].messages. e2e__ [realm_] Rx_AAR_4xxx. qns_stat.success

Description: Success count of Policy Director messages with return code matching 4XXX

node[x].messages. e2e__ [realm_] Rx_AAR_5xxx. qns_stat.success

Description: Success count of Policy Director messages with return code matching 5XXX

node1.counters. [realm_] Rx_RAA.qns_count

Description: Count of messages successful sent to the Policy Director (LB)

node1.counters. [realm_] Rx_AAR_drop. qns_count

Description: Count of messages dropped due to exceeding SLA

node1.counters. [realm_] Rx_AAA_2001. qns_count

Description: Count of AAA messages with result-code = 2001 sent successfully to the Policy Director (LB)

node[x].messages. e2e__ [realm_] Rx_ASR_2001. qns_stat.success

Description: Success message count for return code 2001

node[x].messages. e2e__ [realm_] Rx_ASR_2001. qns_stat.total_ time_in_ms

Description: Total milliseconds of successful messages with return code matching 2001

node[x].messages. e2e__ [realm_] Rx_ASR_3xxx. qns_stat.success

Description: Success count of Policy Director messages with return code matching 3XXX

node[x].messages. e2e__ [realm_] Rx_ASR_5xxx. qns_stat.success

Description: Success count of Policy Director messages with return code matching 5XXX

node1.counters. [realm_] Rx_ASA_bypass. qns_count

Description: Count of message that do not require processing by the policy engine

node1.counters. [realm_]Rx_ASA. qns_count

Description: Count of messages successful sent to the policy engine

node1.counters. [realm_] Rx_ASA_drop. qns_count

Description: Count of messages dropped due to exceeding SLA

node[x].messages. e2e__ [realm_] Rx_RAR_2001. qns_stat.success

Description: Success message count for return code 2001

node[x].messages. e2e__ [realm_] Rx_RAR_2001. qns_stat.total_ time_in_ms

Description: Total milliseconds of successful messages with return code matching 2001

node[x].messages. e2e_<domain>_[realm_] Gx_RAR-T_3xxx. qns_stat.success

Description: Success count of messages with return code matching 3XXX

node[x].messages. e2e__ [realm_] Gx_RAR-T_4xxx. qns_stat.success

Description: Success count of messages with return code matching 4XXX

node[x].messages. e2e__ [realm_] Gx_RAR-T_5xxx. qns_stat.success

Description: Success count of messages with return code matching 5XXX

node1.counters. [realm_] Rx_RAA_bypass. qns_count

Description: Count of messagethat do not require processing by the policy engine

node1.counters. [realm_] Rx_RAA.qns_count

Description: Count of message successful sent to the policy engine

node1.counters. [realm_] Rx_RAA_drop. qns_count

Description: Count of messages dropped due to exceeding SLA

node[x].messages. e2e__ [realm_] Rx_STR_2001. qns_stat.success

Description: Success message count for return code 2001

node[x].messages. e2e__ [realm_] Rx_STR_2001. qns_stat.total_time_in_m

Description: Total milliseconds of successful messages with return code matching 2001

node[x].messages. e2e__ [realm_] Rx_STR_3xxx. qns_stat.success

Description: Success count of messages with return code matching 3XXX

node[x].messages. e2e__ [realm_] Rx_STR_4xxx. qns_stat.success

Description: Success count of messages with return code matching 4XXX

node[x].messages. e2e__ [realm_] Rx_STR_5xxx. qns_stat.success

Description: Success count of messages with return code matching 5XXX

node1.counters. [realm_] Rx_STR.qns_count

Description: "Count of messages successful sent to the policy engine"

node1.counters. [realm_] Rx_STR_drop. qns_count

Description: Count of messages dropped due to exceeding SLA

node1.messages. in_q_Rx_STR. qns_stat.success

Description: "Count of messages successful sent to the policy engine"

node1.messages. in_q_Rx_STR. qns_stat. total_time_in_ms

Description: Total milliseconds of messages successfully sent to the policy engine

node1.messages. diameter_Rx_STR. qns_stat.success

Description: Success message count

node1.messages. diameter_Rx_STR. qns_stat. total_time_in_ms

Description: Total milliseconds of successful messages

node1.counters. [realm_] Rx_STA_2001. qns_count

Description: Count of STA messages with result-code = 2001 sent successfully to the Policy Director (LB)

Alarms

RxAAR

Description: "This alert is fired when the percentage of Success Rx AAR responses send is lesser threshold."

Formula:

sum(increase(diameter_responses_total{command_code=\”AAA\”,response_status=~\”2001\”}[5m])) / sum(diameter_responses_total(outgoing_request_total{command_code=\”AAA\”}[5m])) < 0.90”

RxSTA

Description: "This alert is fired when the percentage of Success Rx STA responses send is lesser threshold."

Formula:

sum(increase(diameter_responses_total{command_code=\”STA\”,response_status=~\”2001\”}[5m])) /sum(diameter_responses_total(outgoing_request_total{command_code=\”STA\”}[5m])) < 0.90

RxRAR

Description: "This alert is fired when the percentage of Success Rx RAR responses Received is lesser threshold."

Formula:

sum(increase(diameter_responses_total{command_code=\”RAA\”, response_status=~\”2001\”}[5m]))/sum(diameter_responses_total(outgoing_request_total{command_code=\”RAA\”}[5m])) < 0.90

RxASR

Description: "This alert is fired when the percentage of Success Rx ASR responses send is lesser threshold."

Formula:

sum(increase(diameter_responses_total{command_code=\”ASA\”,response_status=~\”2001\”}[5m])) /sum(diameter_responses_total(outgoing_request_total{command_code=\”ASA\”}[5m])) < 0.90

pod-down

Description: CDL EP Pod Down

Formula:

up{pod=~'cdl-ep.*'} == 0

pod-down

Description: CDL Pod Slot Change

Formula:

up{pod=\"cdl-slot-session-c1-m1-0\"} == 0

pod-down

Description: Diameter EP Change

Formula:

up{pod=~'diameter-ep.*'} == 0

pod-down

Description: EP Mapping Change

Formula:

up{pod=~'etcd-pcf.*'} == 0

pod-down

Description: Grafana Dashboard Change

Formula:

up{pod=~'grafana-dashboard.*'} == 0

pod-down

Description: Kafka Changed

Formula:

up{pod=~'kafka.*'} == 0

pod-down

Description: LDAP Pod Changed

Formula:

up{pod=~'ldap-pcf.*'} == 0

pod-down

Description: PCF Engine Changed

Formula:

up{pod=~'pcf-engine-pcf.*'} == 0

pod-down

Description: PCF Rest EP Change

Formula:

up{pod=~'pcf-rest-ep.*'} == 0

LDAP Query

Description: "This alert is fired when the success percentage of ldap query request is lesser threshold."

Formula:

sum(increase(message_total{type=~\".*_ldap_query\", status=\"success\"}[5m]))
/sum(increase(message_total{type=~\".*_ldap_query\"}[5m])) < 0.90

LDAP Modify

Description: "This alert is fired when the success percentage of ldap modify request is lesser threshold."

Formula:

sum(increase(message_total{component=\"ldap-ep\", type=~\".*_ldap_modify\",status=\"success\"}[5m]))
/ sum(increase(message_total{component=\"ldap-ep\",type=~\".*_ldap_modify\"}[5m])) < 0.90

PLF Request

Description: This alert is fired when the success percentage of PLF request is lesser threshold.

Formula:

sum(increase(message_total{type=~\"ldap_search-res_success\",status=\"success\"}[5m]))
/sum(increase(message_total{type=~\"ldap_search-res_.*\"}[5m])) <0.90

NAP Notification

Description: This alert is fired when the success percentage of NAP request is lesser threshold.

Formula:

sum(increase(message_total{type=~\"ldap_change-res_success\", status=\"success\"}[5m]))
/sum(increase(message_total{type=~\"ldap_change-res_.*\"}[5m])) <0.90

node-disk-running-full

Description: test

Formula:

node_filesystem_usage > 0.0001

vm-down

Description: VM Down

Formula:

up{pod=~\"node-expo.*\"} == 0

mem-util-high

Description: High Memory Usage

Formula:

avg(node_memory_MemAvailable_bytes /node_memory_MemTotal_bytes * 100) by (hostname) < 20

disk-util-high

Description: High Disk Usage

Formula:

avg (node_filesystem_avail_bytes{mountpoint =\"/\"} /node_filesystem_size_bytes{mountpoint =\"/\"} *100) by (hostname) <20

cpu-util-idle

Description: High CPU Usage

Formula:

avg(rate(node_cpu_seconds_total{mode='idle'}[1m])) by (hostname) *100 < 50