## **Nội dung usecase:**

**Ceph unhealthy và IOPS của VM bị giảm**

Yêu cầu monitor:

* Trạng thái healthy của ceph cluster
* IOPS read/write của VM

Điều kiện alarm

* Alarm 1: ceph unhealthy (kết quả câu lệnh ceph -s) không trả về HEALTH OK
* Alarm 2: IOPS read/write của VM bị giảm:
  + Thông số IOPS read/write của VM bị giảm so với lần monitor gần nhất là 3 lần.
  + Hoặc thông số IOPS read/write của VM nhỏ hơn 1 ngưỡng cho trước

Yêu cầu Graph:

* 1 VM có thể nối với 1 or nhiều volume
* 1 Volume thuộc quản lý của 1 cinder-volume
* 1 cluster openstack có thể có nhiều cinder-volume
* 1 cinder-volume nối với 1 hoặc nhiều ceph cluster
* 1 ceph cluster có thể thuộc 1 hoặc nhiều cinder-volume

Yêu cầu deduce:

* Deduce state:
  + Nếu có alarm 1:Chuyển state của cinder-volume về WARNING, chuyển state của các VM đang dùng volume của cinder-volume đó về WARNING
* Deduce alarm:
  + Nếu có Alarm 1 => Tạo thêm alarm 3 cho instance về việc volume của VM bị ảnh hưởng

Yêu cầu RCA:

* Tạo RCA Alarm 1 => Alarm 2,|Alarm3

## **Thực hiện:**

### 1. Chuẩn bị monitor các thành phần:

Ở đây cần monitor resouce use của host và instance.

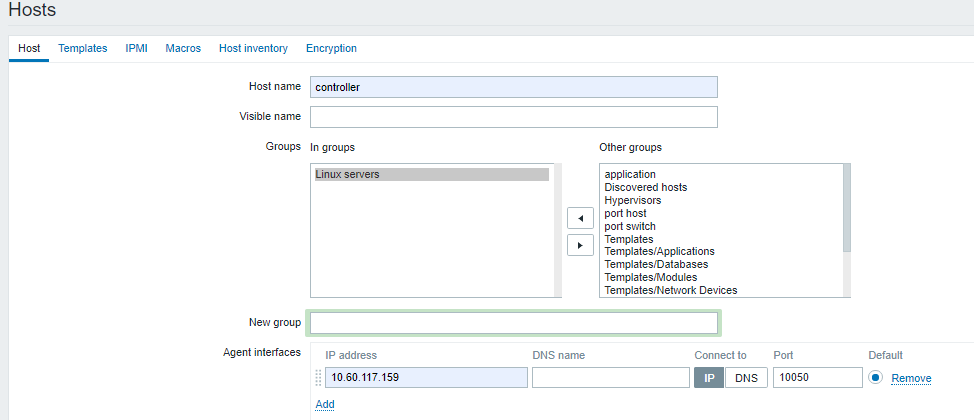
* Đầu tiên cần cài đặt zabbix agent cho mỗi instance, mỗi host. Các agent này cần gửi thông tin thu thập về cho một zabbix server cài trên 1 host (vm), mà host (vm) đó vitrage có thể kết nối được
* Trên UI của zabbix server, ta cấu hình để monitor host resource và instance như sau:

#### a) Monitor host: % use cpu và ram

* Ứng với mỗi host cài nova-compute tạo 1 zabbix host tương ứng:

Hostname: tên host

Agent interface: interface của host, mà host đó dùngkết nối zabbix server .



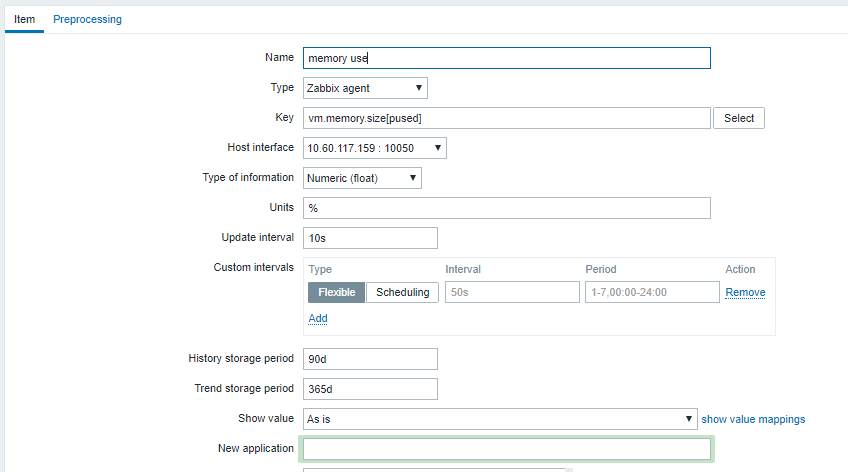
* Tạo item để thu thập thông tin của host:

Vào zabbix UI. tab: Configuration > host > <tên zabbix host> > item > create item

**Memory use:**

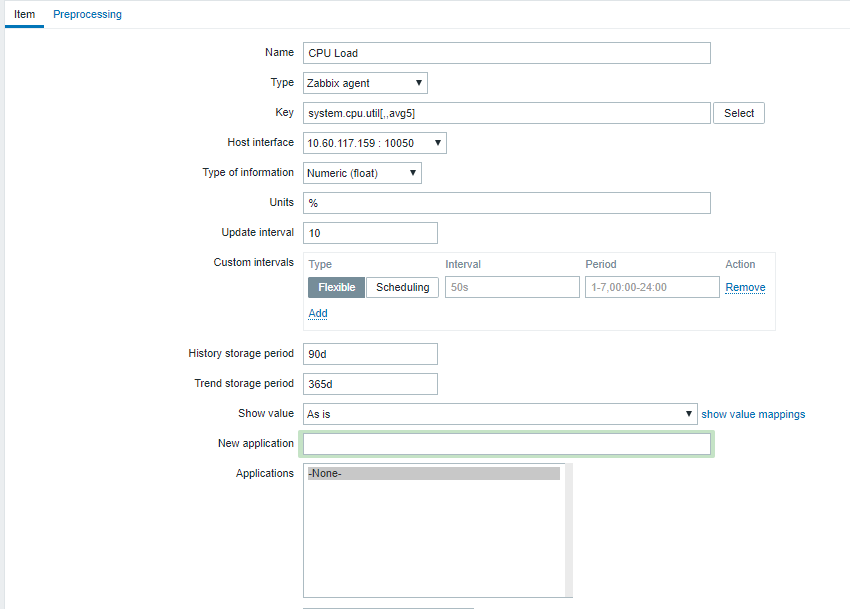
key: vm.memory.size[pused]

( tham khảo https://www.zabbix.com/documentation/3.0/manual/appendix/items/vm.memory.size\_params)



**Cpu use** :

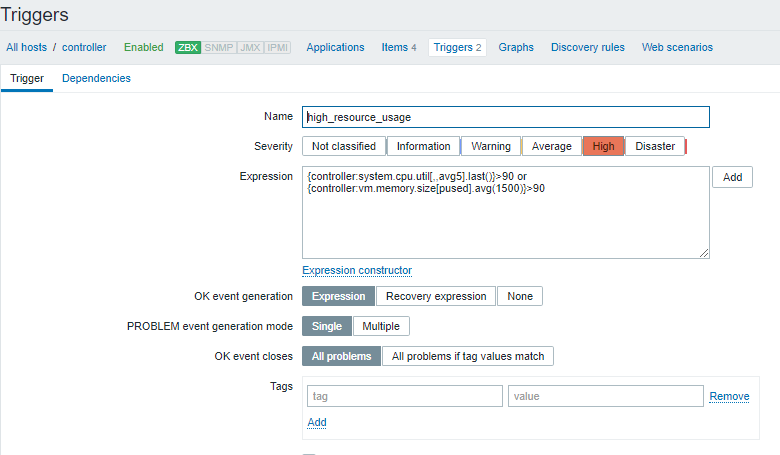
key: system.cpu.util[,,avg5]



* Tạo trigger cảnh báo : khi cpu hoặc mem use trong vòng 5 phút > 90 % bắn alarm

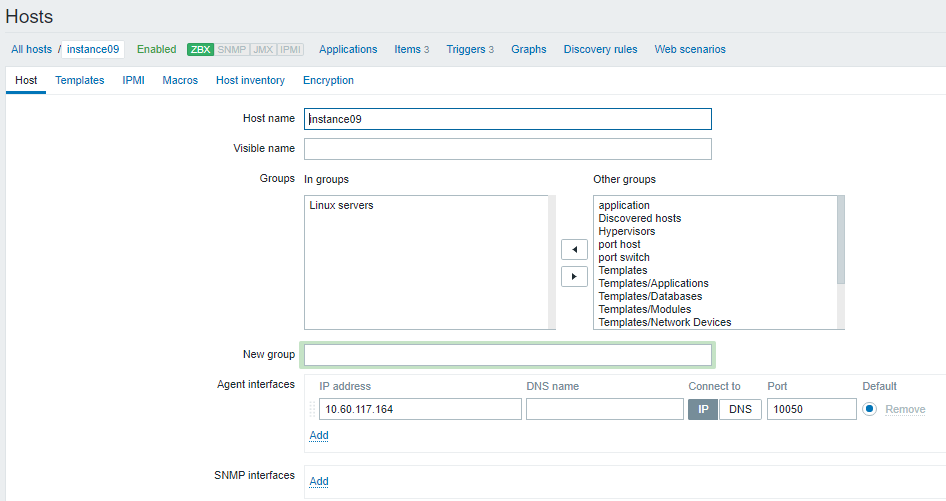
Configuration > host > [controller] > Triggers > create trigger . Tại mục *expression*

{controller:system.cpu.util[,,avg5].last()}>90 or {controller:vm.memory.size[pused].avg(1500)}>90



#### **b) Monitor instance** % use cpu và ram

* Tạo host: ứng với mỗi instance tạo 1 zabbix host:



Các bước sau làm hoàn toàn tương tự như của host

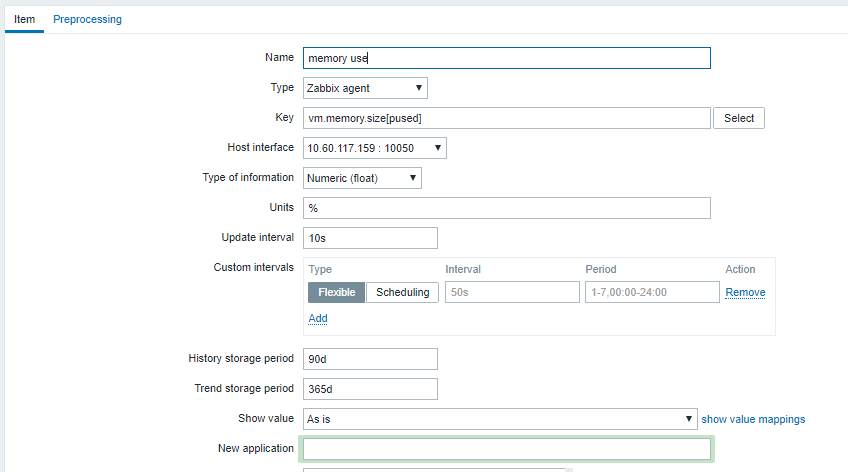
* Tạo item để thu thập thông tin của instance:

Vào zabbix UI. tab: Configuration > host > <tên zabbix host> > item > create item

**Memory use:**

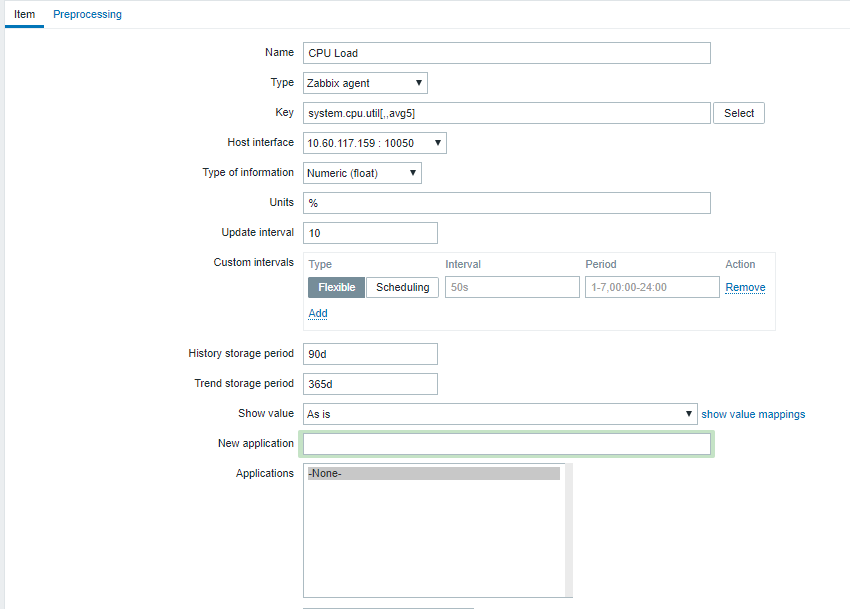
key: vm.memory.size[pused]

( tham khảo https://www.zabbix.com/documentation/3.0/manual/appendix/items/vm.memory.size\_params)



**Cpu use** :

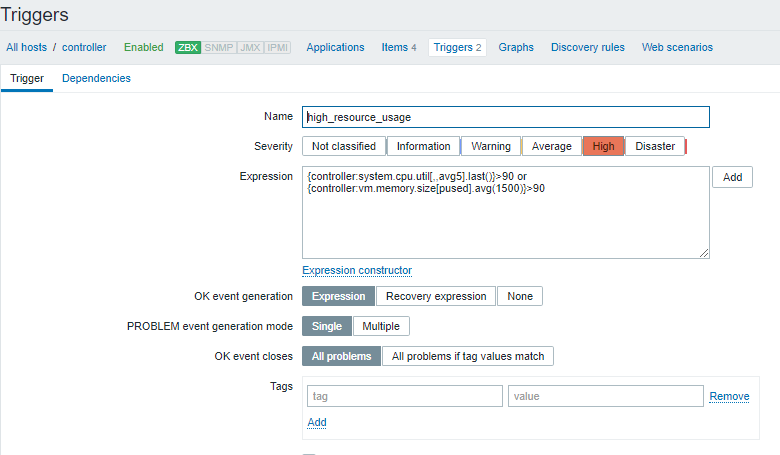
key: system.cpu.util[,,avg5]



* Tạo trigger cảnh báo : khi cpu hoặc mem use trong vòng 5 phút > 90 % bắn alarm

Configuration > host > [controller] > Triggers > create trigger . Tại mục *expression*

{controller:system.cpu.util[,,avg5].last()}>90 or {controller:vm.memory.size[pused].avg(1500)}>90



### 2. Chuẩn bị graph

#### a) add static datasoucre:

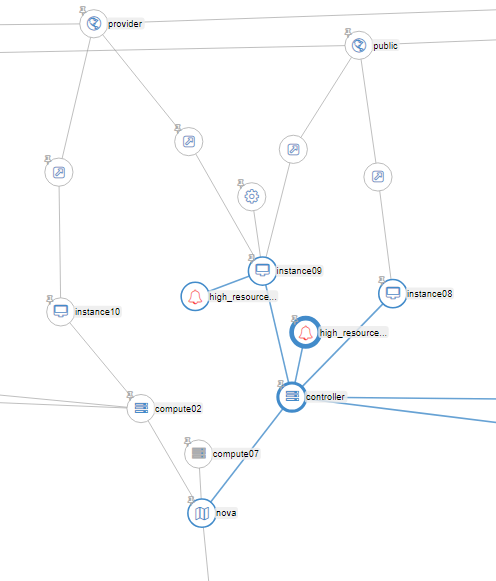
Không yêu cầu

#### b) mapping cho alarm của zabbix vào đồ thị:

Thêm vào file /etc/vitrage/zabbix\_conf.yaml

|  |
| --- |
| - zabbix\_host: instance09  type: nova.instance  name: eff1daed-0d97-4975-abbd-3d3e907aeedf  - zabbix\_host: controller  type: nova.host  name: controller |

Vậy ta đã chuẩn bị xong mô hình input, kết quả:



có zabbix đã bắn 2 alarm : cảnh báo high resouce use trên host và instance

Output: ta cần vitrage

- Deduce alarm-state:

* Deduce stace: Nếu có alarm host high resouce use thì chuyển state của Host + tất cả VM trên host về warning
* Deduce alarm 3: Tạo alarm trên tất cả VM trên host: nội dung thể hiện bị tài nguyên VM ảnh hưởng bởi host

- Thêm rca giữa alarm instance và host

### 3. Thêm template:

Tạo file template /etc/vitrage/templates/usecase-3\_deduce.yaml cho nhiệm vụ deduce

|  |
| --- |
| metadata:  name: deduce alarm when high resouce on host to alarm high resource on instance  description: deduce alarm high resouce on host to alarm high resource on instance  definitions:  entities:  - entity:  category: ALARM  name: high\_resource\_usage  template\_id: host\_alarm  - entity:  category: ALARM  name: high\_resource\_usage  template\_id: instance\_alarm  - entity:  category: RESOURCE  type: nova.host  template\_id: host  - entity:  category: RESOURCE  type: nova.instance  template\_id: instance  relationships:  - relationship:  source: host\_alarm  target: host  relationship\_type: on  template\_id : alarm\_on\_host  - relationship:  source: host  target: instance  relationship\_type: contains  template\_id : host\_contains\_instance  - relationship:  source: instance\_alarm  target: instance  relationship\_type: on  template\_id : alarm\_on\_instance  scenarios:  - scenario:  condition: alarm\_on\_host and host\_contains\_instance  actions:  - action:  action\_type: set\_state  properties:  state: suboptimal  action\_target:  target: host  - action:  action\_type: set\_state  properties:  state: suboptimal  action\_target:  target: instance  - action:  action\_type: raise\_alarm  properties:  alarm\_name: instance resource problem cause by high memmory-cpu host  severity: warning  action\_target:  target: instance |

Tạo file template /etc/vitrage/templates/usecase-2\_rca.yaml cho nhiệm vụ rca

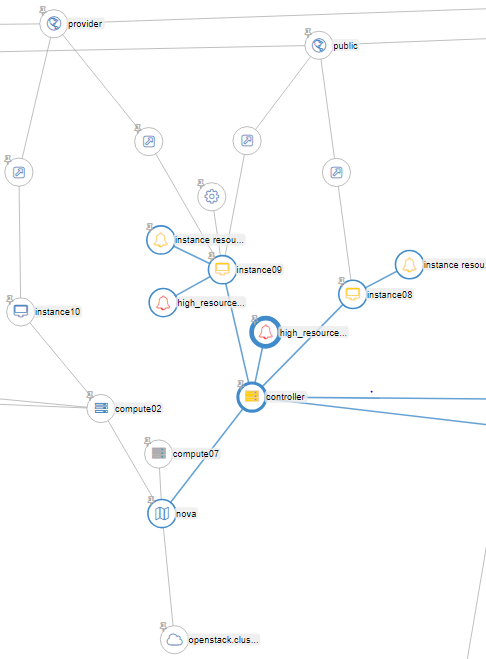
|  |
| --- |
| metadata:  name: rca alarm high resource on instance caused by alarm high resource on instance  description: rca alarm high resource on instance caused by alarm high resource on instance  definitions:  entities:  - entity:  category: ALARM  name: high\_resource\_usage  template\_id: host\_alarm  - entity:  category: ALARM  name: high\_resource\_usage  template\_id: instance\_alarm\_discovered  - entity:  category: ALARM  name: instance resource problem cause by high memmory-cpu host  type: vitrage  template\_id: instance\_alarm\_deduced  - entity:  category: RESOURCE  type: nova.host  template\_id: host  - entity:  category: RESOURCE  type: nova.instance  template\_id: instance  relationships:  - relationship:  source: host\_alarm  target: host  relationship\_type: on  template\_id : alarm\_on\_host  - relationship:  source: instance\_alarm\_discovered  target: instance  relationship\_type: on  template\_id : alarm\_discovered\_on\_instance  - relationship:  source: instance\_alarm\_deduced  target: instance  relationship\_type: on  template\_id : alarm\_deduced\_on\_instance  - relationship:  source: host  target: instance  relationship\_type: contains  template\_id : host\_contains\_instance  scenarios:  - scenario:  condition: alarm\_on\_host and host\_contains\_instance and alarm\_discovered\_on\_instance  actions:  - action:  action\_type: add\_causal\_relationship  action\_target:  source: host\_alarm  target: instance\_alarm\_discovered  - scenario:  condition: alarm\_on\_host and host\_contains\_instance and alarm\_deduced\_on\_instance  actions:  - action:  action\_type: add\_causal\_relationship  action\_target:  source: host\_alarm  target: instance\_alarm\_deduced |

Chạy lệnh:

|  |
| --- |
| $ vitrage template validate --type standard --path /etc/vitrage/templates/usecase3\_deduce.yaml  $ vitrage template add --type standard --path /etc/vitrage/templates/usecase-3\_deduce.yaml  $ vitrage template validate --type standard --path /etc/vitrage/templates/usecase-3\_rca.yaml  $ vitrage template add --type standard --path /etc/vitrage/templates/usecase-3\_rca.yaml |

Kết quả sau khi add template deduce:

có thêm các alarm type vitrage, nội dung: “instance resource problem cause by high memmory-cpu host”



Kết quả sau khi add template rca:

thêm mối quan hệ các alarm instace caused by alarm host

