## **Nội dung usecase:**

Đầu vào monitor:

* Memory của VM (qua aodh hoặc zabix)
* Application(s) của VM (theo appname)

Kịch bản:

* Alarm 1: Memory của VM cao: Nếu % Mem của VM > 90% trong vòng 5 phút
* Alarm 2: Application(s) die:
  + Nếu app không chạy
  + Hoặc application ID bị đổi (do có 1 tiến trình trên VM monitor application, nếu application bị tắt thì tiến trình monitor sẽ bật lại => bị đổi PID)

Yêu cầu Graph:

* Gắn các App của VM vào VM

Yêu cầu tạo deduce:

* N/A

Yêu cầu tạo RCA:

* Tạo RCA Alarm 1 => Alarm 2

## **Thực hiện:**

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

Ở đây cần monitor instance memory và instance application.

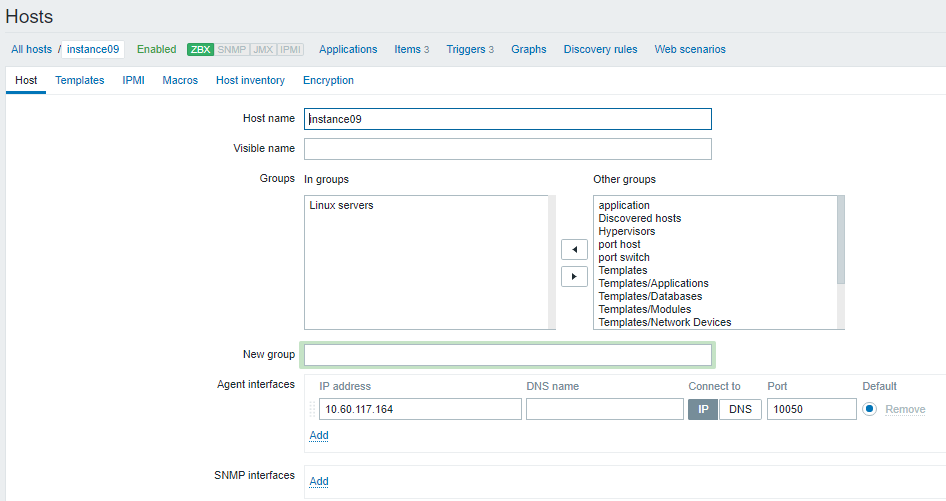
* Đầu tiên cần cài đặt zabbix agent cho mỗi instance. Các agent này cần gửi thông tin thu thập cho một zabbix server cài trên 1 host, mà host đó vitrage có thể kết nối được
* Trên UI của zabbix server, ta cấu hình để monitor memory và application như sau:

#### a) Monitor instance memory:

* Ứng với mỗi instance tạo 1 zabbix host tương ứng:

Hostname: tên instance.

Agent interface: interface của instance, kết nối zabbix server và agent.



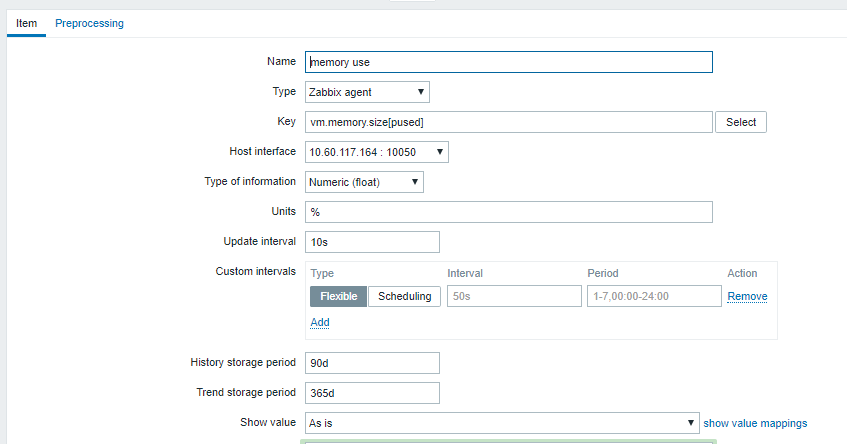
* Tạo item để thu thập thông tin memory của instance:

Vào tab Configuration > host > [instance09] > item > create item

Key: vm.memory.size(pushed)

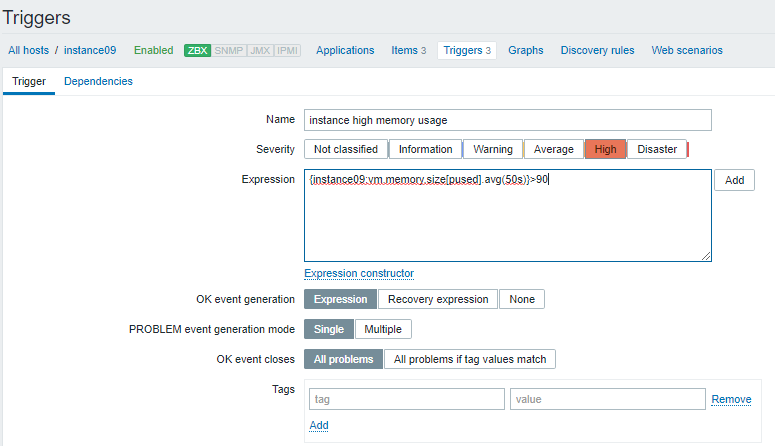
Key này mang ý nghĩa:  % (active + wired) / total memory

Tham khảo tại: https://www.zabbix.com/documentation/3.2/manual/appendix/items/vm.memory.size\_params



* Tạo trigger cảnh báo : khi mem dùng của instance vượt quá 90% thì bắn alarm

Configuration > host > [instance09] > Triggers > create trigger



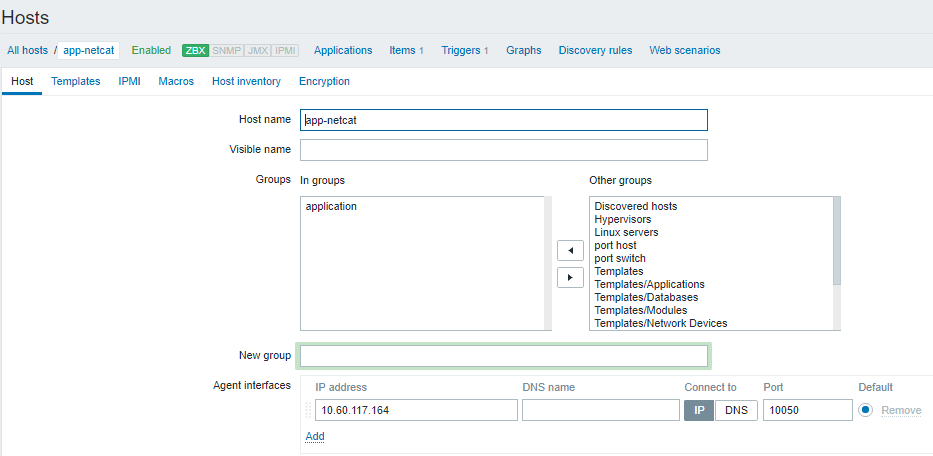
Ở đây avg(50s) tức nó sẽ xem xét giá trị trung bình trong khoảng 50s. Nếu cần trong 5 phút giá trị này cần đặt 1500

#### **b)Monitor instance application**

Ý tưởng: nhận biết application chết bằng cách kiểm tra có sự thay đổi pid không.

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

Giả sử app ta muốn monitor là netcat :



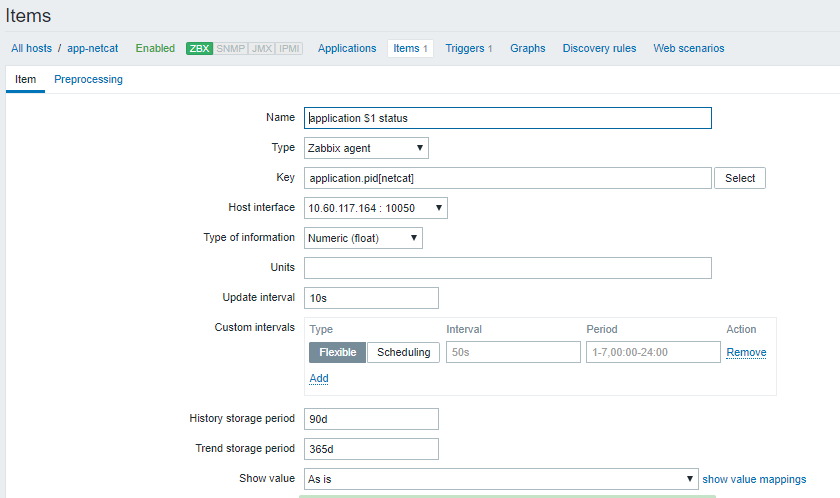
* Lấy thông tin pid: thêm user parameter:

Tạo file:

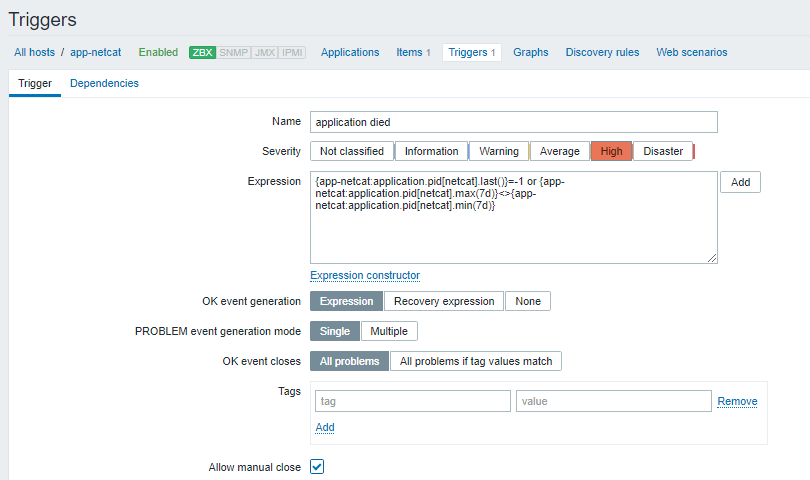
|  |
| --- |
| $ cat /etc/zabbix/zabbix\_agentd.d/userparameter\_application.conf  UserParameter=**application.pid**[\*],if [ "$(pidof $1)" = "" ]; then echo "-1"; else echo $(pidof $1); fi; |

Vào tab Configuration > host > [instance09] > item > create item

với key: appication.pid.[\*] thay \* bằng tên process muốn monitor



* Tạo trigger cảnh báo : khi app không chạy hoặc application pid bị đổi thì bắn alarm
* application.pid [\*] = -1
* OR application.pid [\*].max(7d) <> application.pid [\*].min(7d)



### 2. Chuẩn bị graph

#### a) add static datasoucre:

Thêm entity app vào đồ thị, vd ta muốn thêm monitor vào 1 app “netcat”

Tạo file /etc/vitrage/static\_datasources/app-netcat.yaml nội dung:

|  |
| --- |
| metadata:  name: list of application run on instance  description: list of application run on instance  definitions:  entities:  - static\_id: app-netcat  type: application  id: app-netcat  state: available  - static\_id: instance9  type: nova.instance  id: eff1daed-0d97-4975-abbd-3d3e907aeedf  relationships:  - source: instance9  target: app-netcat  relationship\_type: run |

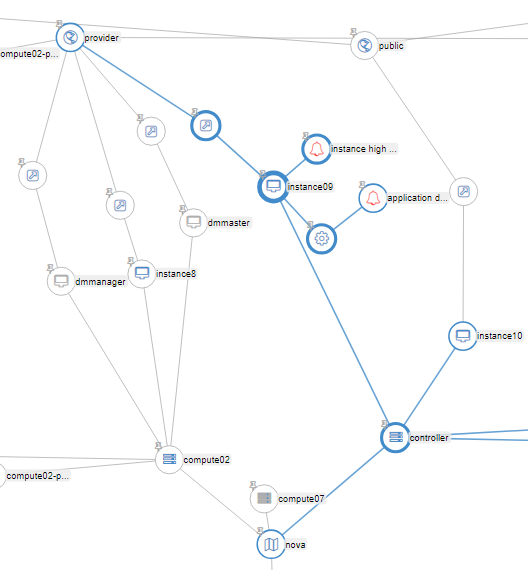
như vậy mối quan hệ: host “instance9” “run” “app-netcat”

#### 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: app-netcat  type: application  name: app-netcat |

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



output: ta cần vitrage thêm rca giữa 2 alarm instance và host:

### 3. Thêm template:

Tạo file template /etc/vitrage/templates/usecase-1.yaml

|  |
| --- |
| metadata:  name: rca application died caused by high mem on instance  description: rca application died caused by high mem on instance  definitions:  entities:  - entity:  template\_id: alarm\_high\_memory\_used  category: ALARM  name: instance high memory usage  - entity:  template\_id: alarm\_application\_died  category: ALARM  name: application died  - entity:  template\_id: instance  category: RESOURCE  type: nova.instance  - entity:  template\_id: application  category: RESOURCE  type: application  relationships:  - relationship:  template\_id : alarm\_high\_memory\_on\_instance  source: alarm\_high\_memory\_used  target: instance  relationship\_type: on  - relationship:  template\_id : alarm\_application\_died\_on\_applicaion  source: alarm\_application\_died  target: application  relationship\_type: on  - relationship:  template\_id : instance\_run\_application  source: instance  target: application  relationship\_type: run  scenarios:  - scenario:  condition: alarm\_high\_memory\_on\_instance and alarm\_application\_died\_on\_applicaion and instance\_run\_application  actions:  - action:  action\_type : add\_causal\_relationship  action\_target:  source: alarm\_high\_memory\_used  target: alarm\_application\_died |

Chạy lệnh:

|  |
| --- |
| $ vitrage template validate --type standard --path /etc/vitrage/templates/usecase1.yaml  $ vitrage template add --type standard --path /etc/vitrage/templates/usecase1.yaml |

Kết quả sau khi add template:

