Openstack

OpenStack est un outil permettant de mettre en œuvre des clouds Operating system

See : <http://www.openstack.org/software/>

Une fois installé il permet aux utilisateurs du cloud de créer des tenants, des virtuels machines, des réseaux et des infrastructures de stockages. Il permet aussi de créer tous les shared services pour la création d’un datacenter virtuel.

Les utilisateurs ont à leurs dispositions un portail WEB, un CLI et des API

WEB : <http://www.openstack.org/software/openstack-dashboard/>

CLI : <http://docs.openstack.org/cli-reference/content/>

API : <http://developer.openstack.org/api-ref.html>

**Sujet 1 : documentation and release management tools for Cloud infrastructure.**

Les différents objets virtuel qui peuvent être crées par les utilisateurs sont gérer par OpenStack. Malheureusement, il n’existe pas d’outil permettant d’historiser les configurations. Aucun outil de reporting n’est disponible.

Pour créer cet outil après avoir installé une VM chez cloudwatt avec les CLI, une base de données et un serveur WEB.

**Team 1 : Référenciel**

Appel des API / CLI pour rechercher les caractéristiques des tenant existant dans openstack et stockages dans une base de données. Les objects a documenté sont VM, Security group avec leurs rêgles , les reseaux, les FW avec leurs regles, les loadBalenceur et les routeurs.

Ce programme devra être appelé par cron toutes les jours.

exemple de la recherche d’information sur une VM :

user3@rebond:~$ nova list

+--------------------------------------+--------+--------+------------+-------------+----------------------------------+

| ID | Name | Status | Task State | Power State | Networks |

+--------------------------------------+--------+--------+------------+-------------+----------------------------------+

| 79a51cf6-5453-459f-80fe-3bbe636e6d3d | rebond | ACTIVE | - | Running | admin=10.100.100.2, 84.39.40.191 |

+--------------------------------------+--------+--------+------------+-------------+----------------------------------+

user3@rebond:~$ nova show 79a51cf6-5453-459f-80fe-3bbe636e6d3d

+--------------------------------------+--------------------------------------------------------------------+

| Property | Value |

+--------------------------------------+--------------------------------------------------------------------+

| OS-DCF:diskConfig | MANUAL |

| OS-EXT-AZ:availability\_zone | nova |

| OS-EXT-STS:power\_state | 1 |

| OS-EXT-STS:task\_state | - |

| OS-EXT-STS:vm\_state | active |

| OS-SRV-USG:launched\_at | 2015-12-08T06:01:51.000000 |

| OS-SRV-USG:terminated\_at | - |

| accessIPv4 | |

| accessIPv6 | |

| admin network | 10.100.100.2, 84.39.40.191 |

| config\_drive | |

| created | 2015-12-08T06:01:29Z |

| flavor | n1.cw.standard-1 (21) |

| hostId | f73ca67063247f1bc7f4ad7156c54c0cd9c0803d669e34e681b4e748 |

| id | 79a51cf6-5453-459f-80fe-3bbe636e6d3d |

| image | vdesktop-formation-20151017 (1e0db788-44e6-4daf-9776-84656736da00) |

| key\_name | gpocentek |

| metadata | {} |

| name | rebond |

| os-extended-volumes:volumes\_attached | [] |

| progress | 0 |

| security\_groups | admin, default |

| status | ACTIVE |

| tenant\_id | 586677920fc8482ca0d2d1159241b270 |

| updated | 2015-12-08T15:25:17Z |

| user\_id | c5cce8cc48444635b5a0f74a20d424c3 |

+--------------------------------------+--------------------------------------------------------------------+

user3@rebond:~$

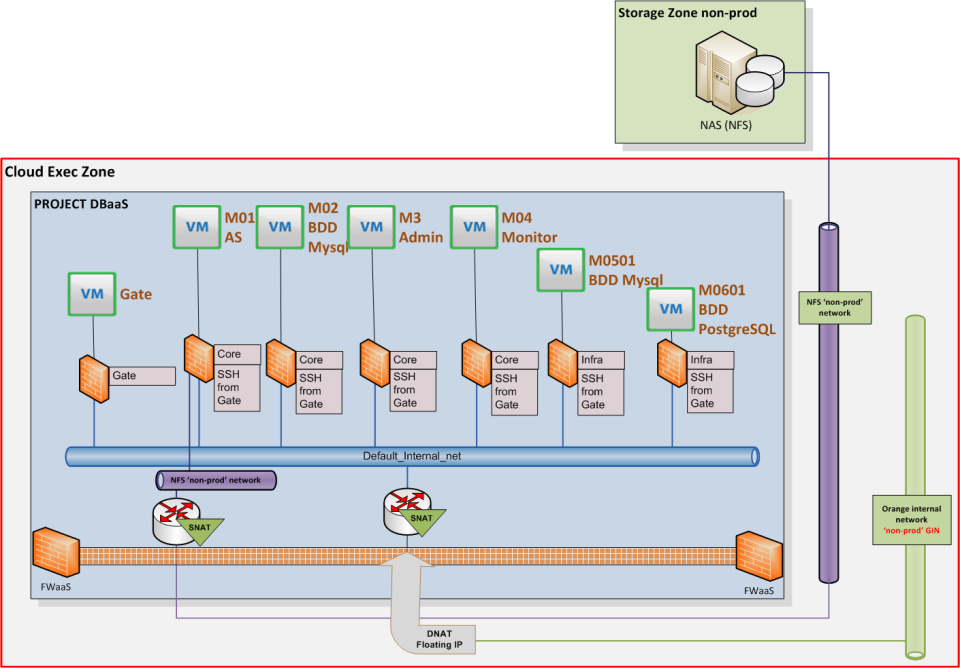
**Team 2.A : Doc**

Ce sujet n’est possible que si le sujet Team1 est pris. Le model de données devra être construit en concertation.

Générateur de page HTML pour document et faire des liens sur les objets et leurs caractéristiques dans la base de données.

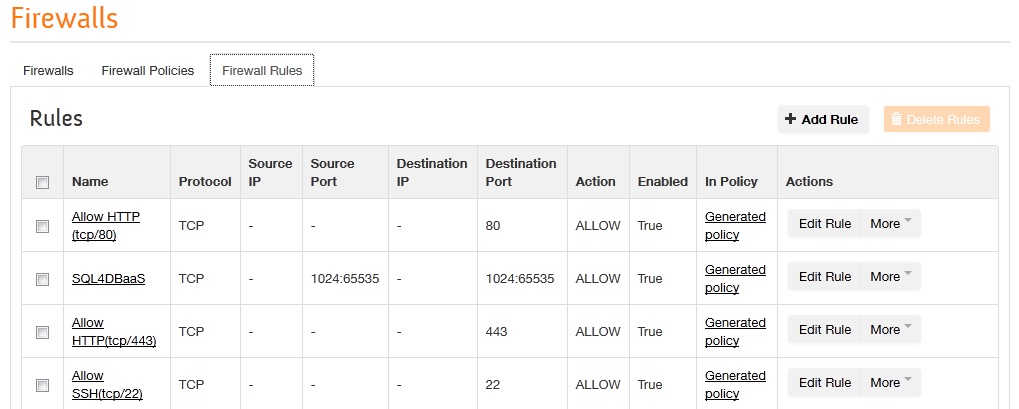
Il devra y avoir un graph de présentation du tenant et une suite de liste d’info.

exemple :

DBaaS Exemple  
  
  


The flow requiered by application

|  |  |  |
| --- | --- | --- |
| **​Source** | **destination ​** | **protocol​** |
| ​"non-prod" GIN (DBaaS User & Admin) | M01, M03, M4​ | HTTP/HTTPS​ |
| "non-prod" GIN (DBaaS Admin​) | Gate​ | SSH​ |
| ​Gate | M01,M02, M03, M04, M050x, M060x​ | SSH​ |
| ​M01, M03 | M050x, M060x​ | SSH​ |
| M01, M04, M04​ | M02​ | SQL​ |
| DBaaS User​ | M050x M060x​ | SQL 1024 : 65535​ |
| ​M01 | ​Filer | ​NFS |



$ nova list  
+--------------------------------------+-------------+--------+------------+-------------+  
| ID                                   | Name        | Status | Task State | Power State |  
+--------------------------------------+-------------+--------+------------+-------------+  
| 8a31f79e-3055-4e22-b4e1-675159e7db43 | dbaas-m01   | ACTIVE | -          | Running     |

| 0be5b29a-d31d-46ad-95fc-3caba63ba32c | dbaas-m02   | ACTIVE | -          | Running     |

| a7167423-7581-44c6-a9cf-c1af956533e1 | dbaas-m03   | ACTIVE | -          | Running     |

| a1202ff7-ac0a-4d43-bf7e-556d302ceb1a | dbaas-m04   | ACTIVE | -          | Running     |

| ac8b862b-bd80-41c1-aac4-d87f851e8a82 | dbaas-m0501 | ACTIVE | -          | Running     |

| 2bb2b190-5afc-4fde-a04c-d434383f1548 | dbaas-m0502 | ACTIVE | -          | Running     |

| 7a252e3f-2d9a-4206-bcc4-0b3fd447ed59 | dbaas\_gate  | ACTIVE | -          | Running     |

+--------------------------------------+-------------+--------+------------+-------------+

**Team 2.B Rules**

Ce sujet n’est possible que si le sujet Team1 est pris. Le model de données devra être construit en concertation.

Certaine règles doivent être respecté par les tenant admin. Quand une règle n’est pas respecter dans un tenant, un mail doit être envoyé au responsable sécurité et au tenant admin.

* tous les jours un check des règles doit être fait.
* Le non-respect d’une règle doit être historié.
* La notification du non-respect d’une règle peut être désactivée pour un tenant par l’administrateur.
* un portail avec des feux vert et rouge doit être à disposition.
* Les règles sont soit des requêtes SQL a exécuté dans la base de données soit des scripts shell qui renvois <int>:<texte> ou <int> est une gravité 0 vert, 1 orange, 2 rouge

**Sujet 2 : SLA for Cloud infrastructure**

Le Cloud OS ne dispose pas d’outil pour vérifier que l’infrastructure est opérationnelle. Pour calculer ce taux de disponibilité il faut toutes les x minutes regarder que le service est opérationnelle.

Pour ce faire faut lancer un ensemble de tache de contrôle et stoker les résultats dans une base de données. En cas de problème détecter (résultat plus long qu’un délai paramétrable par teste) il faut aussi envoyer un mail d’alerte a un administrateur paramétrable.

Chaque test peut se définir celons la structure suivante :

* ID
* Nom
* Type
* cmd
* Resultat
* responceTime (ms)
* intervale

le type peut être :

SSH : la machine ‘cmd’ doit être accessible en ssh

Ping : l’adresse IP ‘cmd’ doit répondre au ping

http : une requette http ‘cmd’ est envoyé, le ‘resultat’ doit être inclus dans la réponce

https : une requette https ‘cmd’ est envoyé, le ‘resultat’ doit être inclus dans la réponce

os : un script shell ‘cmd’ est executé le ‘resultat’ doit être inclus dans la sortie standard

Chaque test une fois exécuter est et noté OK ou KO avec l’horodatage et id. le résultat du test peut être stocké ou non en fonction d’un paramètre.

Un portail doit être mise à disposition pour afficher la météo du service.