목차

VBMC를 통한 TripleO 및 IPMI 3
하이퍼 바이저에서 3

VBMC를 통한 TripleO 및 IPMI

instack을 사용하여 가상 환경에서 TripleO를 테스트 할 때 Ironic이 기본 pxe_ssh 드라이버 대신 IPMI를 사용하여 가상 시스템을 제어하기를 원했습니다.

내가 사용한 설정은 매우 간단합니다. Centos 7.2를 설치 한 Triplemetal 시스템 한 대는 TripleO documentation을 사용하여 환경 설정을 진행했습니다 . 글을 쓰는 시점에서 저는 Mitaka와 Newton에서 테스트를 해왔습니다.

이미 작동중인 언더 클라우드 VM 설정이 있고 그에 언더 클라우드 소프트웨어를 설치했다고 가정합니다 (openstack undercloud install). 또한 필요한 이미지 (overcloud-full, ironic-python-agent 등)를 이미 가져 왔습니다. 이제 " 베어 메탈 " 가상 머신을 정의하는 instackenv.json을 Ironic으로 가져올 준비가되었습니다.

호스트 (하이퍼 바이저-libvirt를 실행중인 베어 메탈 머신과 스택 된 " 베어 메탈 " 가상 머신)에서 VirtualBMC를 설치하고 구성합니다.

하이퍼 바이저에서

VirtualBMC는 pip install을 통해 가장 잘 설치할 수 있습니다. yum을 통해 pip를 설치할 수 있습니다. 그러나 그것을 복제하고 git 에서 수동으로 설치하도록 선택할 수 있습니다 .

```
$ sudo yum install python-pip -y
...
$ sudo pip install virtualbmc
```

주) 필요한 경우 추가로 'libvirt-devel' 패키지를 설치해야 할 수 도 있습니다.

```
$ sudo yum install -y gcc python-devel libvirt-devel
```

또한 방화벽을 사용하여 iptables를 제어하는 것이 좋습니다. 여전히 설치되어 있지 않은 경우 yum을 사용하여 설치하고 활성화 할 수 있습니다.

```
$ sudo yum install firewalld
$ sudo systemctl enable firewalld
$ sudo systemctl start firewalld
```

아이디어는 호스트에 하나의 IP (기본적으로 192.168.122.1)가 있으며 instack VM에서 액세스 할 수 있다는 것입니다. IPMI는 기본적으로 포트 623을 사용합니다. 예를 들어 ipmitool 명령을 실행하면 기본적으로이 포트가 사용됩니다. 그러나 -p 플래그를 제공하고 다른 포트를 지정할 수 있습니다. IP가 비슷하기 때문에 VM마다 다른 포트가 필요합니다 (호스트의 IP 192.168.122.1).

vbmc를 통해 제어하려는 가상 머신의 양에 따라 언더 클라우드 VM이 vbmc 데몬에 도달 할 수 있도록 필요한 포트에 대한 액세스를 허용하십시오. 제 경우에는 5 개의 오버 클라우드 가상 머신 (baremetalbrbm_0..4)이있었습니다. 이것은 포트를 할당하기로 선택한 방법입니다.

baremetalbrbm_0은 기본 IPMI 포트 623을 가져 옵니다. baremetalbrbm_1 포트 624 baremetalbrbm_2 포트 625 baremetalbrbm_3 포트 626 baremetalbrbm_4 포트 627

방화벽에서 다음 포트를 허용합니다.

```
$ sudo firewall-cmd --permanent --zone=public --add-port=623-627/udp
$ sudo firewall-cmd --reload
```

이제 가상 머신 (“ 도메인 ”) 을 vbmc에 추가 할 수 있습니다.

```
$ sudo vbmc list
+-------------+--------+---------+------+
| Domain name | Status | Address | Port |
+-------------+--------+---------+------+
+-------------+--------+---------+------+
$ sudo vbmc add baremetalbrbm_0 --username root --password calvin --port 623
$ sudo vbmc add baremetalbrbm_1 --username root --password calvin --port 624
$ sudo vbmc add baremetalbrbm_2 --username root --password calvin --port 625
$ sudo vbmc add baremetalbrbm_3 --username root --password calvin --port 626
$ sudo vbmc add baremetalbrbm_4 --username root --password calvin --port 627
$ sudo vbmc list
+-----------------+--------+---------+------+
|   Domain name   | Status | Address | Port |
+-----------------+--------+---------+------+
| baremetalbrbm_0 |  down  |    ::   | 623  |
| baremetalbrbm_1 |  down  |    ::   | 624  |
| baremetalbrbm_2 |  down  |    ::   | 625  |
| baremetalbrbm_3 |  down  |    ::   | 626  |
| baremetalbrbm_4 |  down  |    ::   | 627  |
+-----------------+--------+---------+------+
```

그래! 추가했습니다. 이제 시작하면됩니다.

```
$ for i in {0..4};do sudo vbmc start baremetalbrbm_${i};done
2016-12-06 13:36:40,806.806 685 INFO VirtualBMC [-] Virtual BMC for domain
baremetalbrbm_0 started
2016-12-06 13:36:41,034.034 694 INFO VirtualBMC [-] Virtual BMC for domain
baremetalbrbm_1 started
2016-12-06 13:36:41,251.251 703 INFO VirtualBMC [-] Virtual BMC for domain
baremetalbrbm_2 started
2016-12-06 13:36:41,467.467 712 INFO VirtualBMC [-] Virtual BMC for domain
baremetalbrbm_3 started
2016-12-06 13:36:41,684.684 721 INFO VirtualBMC [-] Virtual BMC for domain
baremetalbrbm_4 started
```

그들이 잘 듣고 있는지 확인합시다 :

```
$ sudo vbmc list
+-----------------+---------+---------+------+
|   Domain name   |  Status | Address | Port |
+-----------------+---------+---------+------+
| baremetalbrbm_0 | running |    ::   | 623  |
| baremetalbrbm_1 | running |    ::   | 624  |
| baremetalbrbm_2 | running |    ::   | 625  |
| baremetalbrbm_3 | running |    ::   | 626  |
| baremetalbrbm_4 | running |    ::   | 627  |
+-----------------+---------+---------+------+

$ sudo lsof -Pni:623-627
COMMAND PID USER   FD   TYPE  DEVICE SIZE/OFF NODE NAME
vbmc    685 root    3u  IPv6 4622045      0t0  UDP *:623
vbmc    694 root    3u  IPv6 4622046      0t0  UDP *:624
vbmc    703 root    3u  IPv6 4622047      0t0  UDP *:625
vbmc    712 root    3u  IPv6 4625461      0t0  UDP *:626
vbmc    721 root    3u  IPv6 4616064      0t0  UDP *:627
```

Instack VM에서

이제 instack vm에 로그인하고 ipmitool을 사용하여 연결을 확인할 수 있습니다.

```
[stack@instack ~]$ ipmitool -I lanplus -U root -P calvin -H 192.168.122.1
power status     # defaults to 623
Chassis Power is off
[stack@instack ~]$ ipmitool -I lanplus -U root -P calvin -H 192.168.122.1
power status -p 627
Chassis Power is off
```

instack.json은 다음과 같아야합니다 (“name”: “bm [0-4]“속성을 추가하여 나중에 전체 UUID를 복사 / 붙여 넣기하지 않고도 명령을 더 쉽게 실행할 수 있기 때문에 추가했습니다)

```
{
  "nodes": [
    {
      "mac": [
        "00:3f:f7:9f:2e:eb"
      ],
      "name": "bm0",
      "cpu": "2",
      "memory": "8192",
      "disk": "42",
      "arch": "x86_64",
      "pm_user": "root",
      "pm_addr": "192.168.122.1",
      "pm_password": "calvin",
```

```
      "pm_type": "pxe_ipmitool"
    },
    {
      "mac": [
        "00:1e:c1:2c:e0:35"
      ],
      "name": "bm1",
      "cpu": "2",
      "memory": "8192",
      "disk": "42",
      "arch": "x86_64",
      "pm_user": "root",
      "pm_addr": "192.168.122.1",
      "pm_password": "calvin",
      "pm_type": "pxe_ipmitool"
    },
    {
      "mac": [
        "00:56:34:66:7a:fc"
      ],
      "name": "bm2",
      "cpu": "2",
      "memory": "8192",
      "disk": "42",
      "arch": "x86_64",
      "pm_user": "root",
      "pm_addr": "192.168.122.1",
      "pm_password": "calvin",
      "pm_type": "pxe_ipmitool"
    },
    {
      "mac": [
        "00:8a:c5:c7:22:46"
      ],
      "name": "bm3",
      "cpu": "2",
      "memory": "8192",
      "disk": "42",
      "arch": "x86_64",
      "pm_user": "root",
      "pm_addr": "192.168.122.1",
      "pm_password": "calvin",
      "pm_type": "pxe_ipmitool"
    },
    {
      "mac": [
        "00:a3:bd:79:54:b6"
      ],
      "name": "bm4",
      "cpu": "2",
```

```
      "memory": "8192",
      "disk": "42",
      "arch": "x86_64",
      "pm_user": "root",
      "pm_addr": "192.168.122.1",
      "pm_password": "calvin",
      "pm_type": "pxe_ipmitool"
    }
  ]
}
```

이제 instackenv.json (openstack baremetal import instackenv.json)을 가져올 수 있습니다 .

그리고 네, 당신은 올바르게 알고 있습니다 – 우리는 여전히 노드 당 포트를 정의하지 않았습니다. 내가 직면 한 문제는 instackenv.json에 “pm_port”를 추가 할 수 없다는 것입니다. 노드를 Ironic으로 가져온 후에 이것을 구성해야했습니다.

```
$ ironic node-update bm1 add driver_info/ipmi_port=624
$ ironic node-update bm2 add driver_info/ipmi_port=625
$ ironic node-update bm3 add driver_info/ipmi_port=626
$ ironic node-update bm4 add driver_info/ipmi_port=627
```

이것은 노드 bm0을 제외한 모든 노드에 대해 수행되어야합니다 (기본 IPMI 포트 623에서 청취 중이므로 필요하지 않습니다). 그게 다야! 이제 내부 검사를 실행하고 배포를 진행할 수 있습니다.

Ironic이 동일한 IP 주소 (192.168.122.1)로 여러 노드를 부팅해야한다는 사실이 마음에 들지 않았기 때문에 벌크 내부 검사를 실행할 수 없다는 점에주의해야합니다. 하이퍼 바이저의 VirtualBMC에서 우리가 “ 해킹 ” 하고 있는지는 알 수 없습니다. 물론 실제 베어 메탈 환경에서는 동일한 네트워크에서 동일한 IP를 갖는 것이 매우 나쁩니다.

편집 : 이미 pxe_ssh에서 멀어지고 기본적으로 virtualbmc와 함께 pxe_ipmi를 사용하는 작업이 있습니다 :

https://blueprints.launchpad.net/tripleo/+spec/switch-to-virtualbmc

From:
https://atl.kr/dokuwiki/ - **AllThatLinux!**

Permanent link:
https://atl.kr/dokuwiki/doku.php/vbmc%EB%A5%BC_%ED%86%B5%ED%95%9C_tripleo_%EB%B0%8F_ipmi

Last update: **2021/02/05 03:04**