Bifrost standalone Ironic を使った GPU基盤構築運用への適用について

GMO INTERNET

STOCK CODE : 9449

GMOインターネット株式会社
郷古直仁
@naoto_gohko

2019年07月22日 RoomF

Bifrost standalone Ironic を使ったGPU基盤構築運用への適用について

GMOインターネット株式会社
郷古直仁
@naoto_gohko

Agenda #1

- A) our environment
  - OpenStack Junoで運用しているIronic環境 (under cloud/Nova-Ironic)
  - GPUcloud環境
- B) 最近のbaremetal server deploy事情
  - Hardware側の状況
  - MAAS系
  - Ironic系 under cloud / nova compute Ironic
  - Ironic stand-alone (or + Bifrost)
  - OpenStack service stand-alone

Agenda #2

- C) 今回のNetwork/Server構成から適用を考える
  - 1) Redfish driverの充実度は?
  - 2) dhcpなしでできるのか、(Virtual media boot deploy)
  - 3) Linuxのsoftware RAID の適用は?
  - 4) どのBaremetal deploy系を使うのか?
- D) Bifrost Ironic stand alone deploy
  - Bifrost install
  - Config change for drivers
- summary

LT presenter(It's me)

- Naoto Gohko / 郷古　直仁
  (@naoto_gohko)

@MikumoConoHa

- Cloud Service development divistion, GMO Internet Inc.,
  - OpenStackでpublic cloudサービス
- 最近の活動主体
  - Japan OpenStack user会
- その他参加勉強会など
  - Rancher JP，PaaS，Serverless, SDNなど

A) our environment

これまでのIronic利用環境と、GPUcloud環境

History / Before ConoHa birth

なぜOpenStackを使うようになったのか

Our OpenStack env.

Public Clouds

GMO Internet, Inc: OpenStack infra.

Onamae.com VPS (2012/03) :
http://www.onamae-server.com/
Forcus: global IPs; provided by simple "nova-network"

お名前.com レンタルサーバー by GMO

OpenStack Diablo
on CentOS 6.x

Nova | Nova network | Keystone | Glance

tenten VPS (2012/12)
http://www.tenten.vn/
Share of OSS by Group companies in Vietnam

Shared codes → tenten.vn by GMO

ConoHa VPS (2013/07) :
http://www.conoha.jp/
Forcus: Quantam(Neutron) overlay tenant network

ConoHa by GMO

OpenStack Glizzly
on Ubuntu 12.04

Quantam | ovs + gre tunnel overlay | Nova | Keystone | Glance

GMO AppsCloud (2014/04) : http://cloud.gmo.jp/
OpenStack Havana based 1st region
Enterprise grade IaaS with block storage, object storage, LBaaS and baremetal compute was provided

GMO AppsCloud

OpenStack Havana
on CentOS 6.x

Keystone | Nova | Baremetal compute | Glance | Ceilometer | Cinder | Neutron | LBaaS

Shared codes → お名前.com クラウド by GMO

Onamae.com Cloud (2014/11)
http://www.onamae-cloud.com/
Forcus: Low price VM instances, baremetal compute and object storage

Glance | Keystone | Neutron | Ceilometer | Nova | Baremetal compute

ConoHa Cloud (2015/05/18) http://www.conoha.jp/
Forcus: ML2 vxlan overlay, LBaaS, block storage, DNSaaS(Designate) and original services by keystone auth

ConoHa by GMO | Z.com Cloud

OpenStack Juno
on CentOS 7.x

Designate | GSLB | Keystone | Nova | Glance | Ceilometer | Cinder | Neutron | LBaaS

GMO AppsCloud (2015/09/27) : http://cloud.gmo.jp/
2nd region by OpenStack Juno based
Enterprise grade IaaS with High IOPS Ironic Compute and Neutron LBaaS

GMO AppsCloud

Keystone | Ceilometer | Glance | Cinder | Nova | Ironic | Neutron | LBaaS

Shared cluster

Swift cluster

Swift

Swift

Upgrade Juno

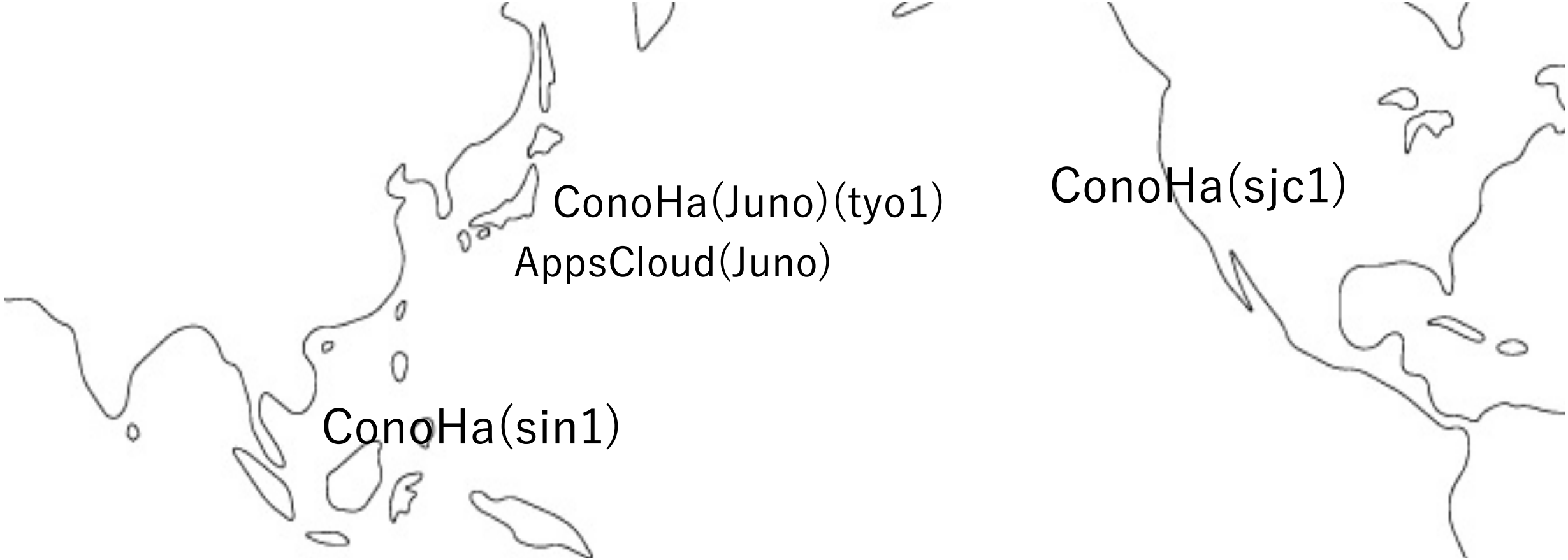
Swift

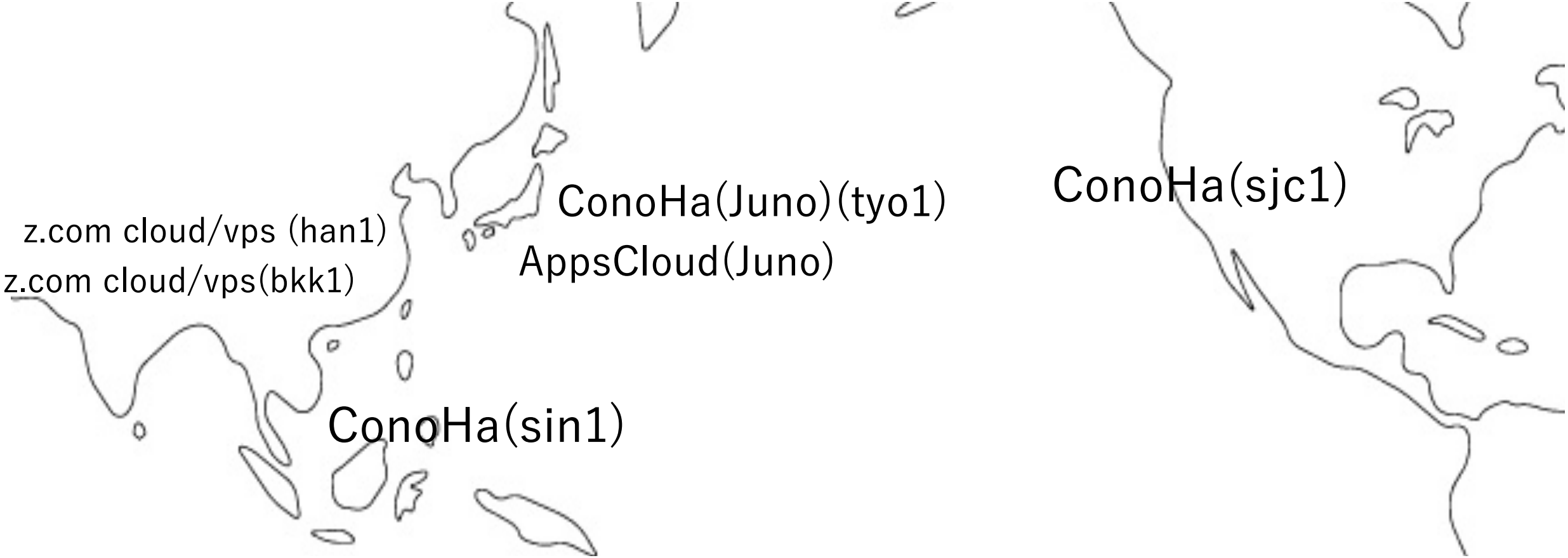
Swift

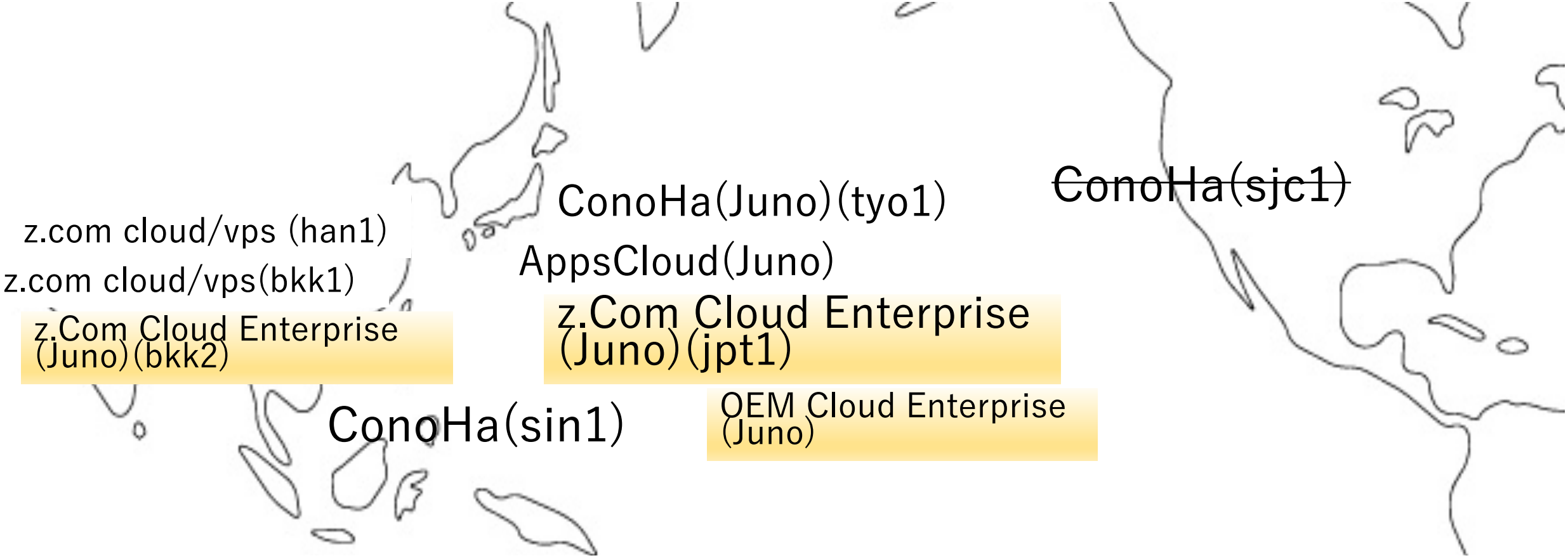
Swift

Increasing OpenStack Juno cluster (based on ConoHa cloud);

OpenStack Juno is EOL


ConoHa(Juno)(tyo1)
AppsCloud(Juno)
ConoHa(sin1)
ConoHa(sjc1)


ConoHa(Juno)(tyo1)
AppsCloud(Juno)
ConoHa(sjc1)
z.com cloud/vps (han1)
z.com cloud/vps(bkk1)
ConoHa(sin1)



han: hanoi
bkk: bankoku

最新OpenStackが”Stein”とか
そのなかで、アジアの各所に
cluster増やしてきた

タイとはベトナムに
パートナーを増やしてきた

But!!!
OpenStack Juno is EOL

OpenStack Junoで運用しているIronic環境

既存のサービス

Ironic deployment for undercloud/nova

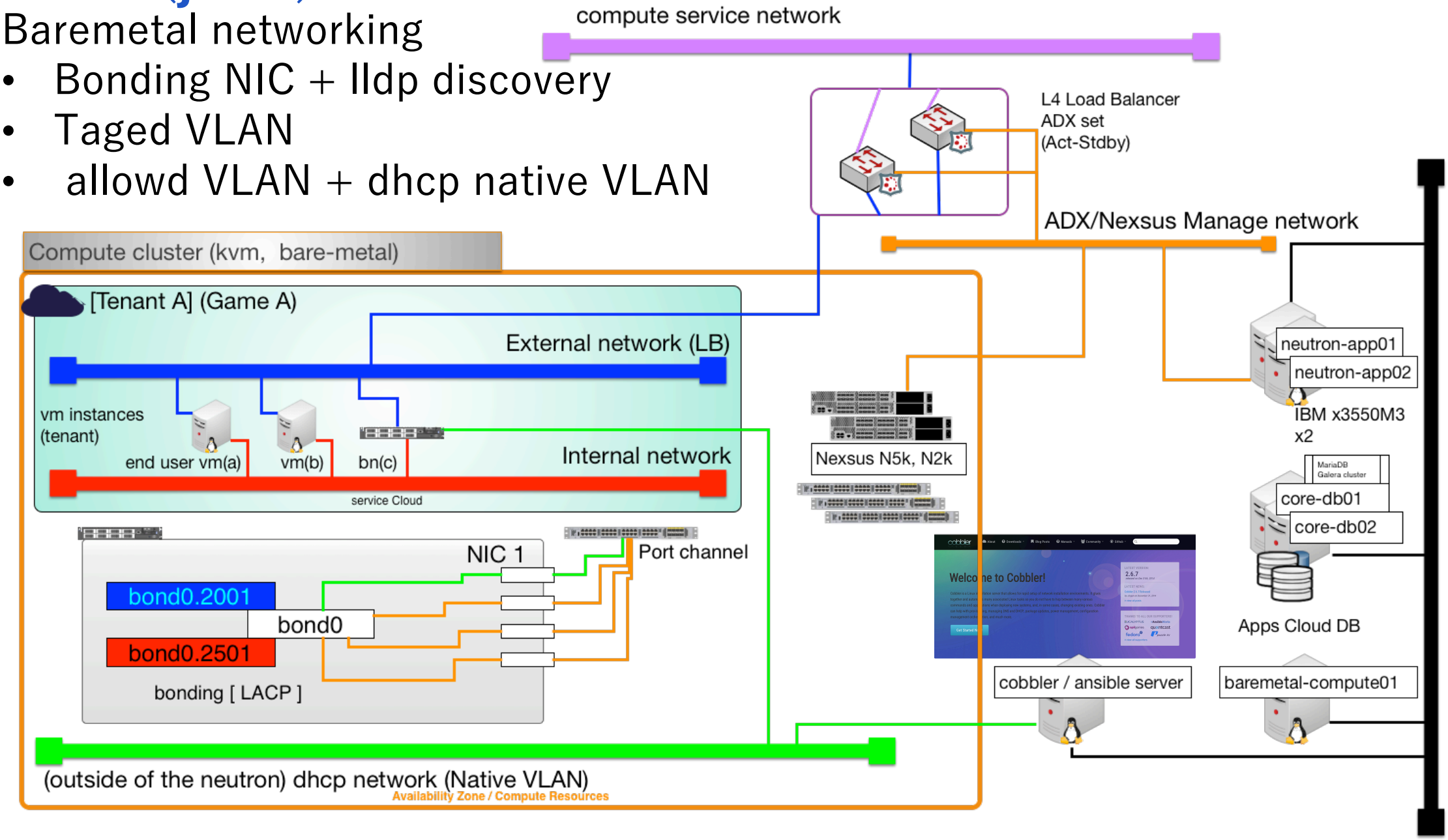
- Cloud
  - Z.com cloud public (Jpn, Thai)
  - GMO Apps Cloud

- All in one OpenStack Juno with Ironic

Ironic(juno) nova-Ironic

Baremetal networking

- Bonding NIC + lldp discovery
- Taged VLAN
- allowd VLAN + dhcp native VLAN

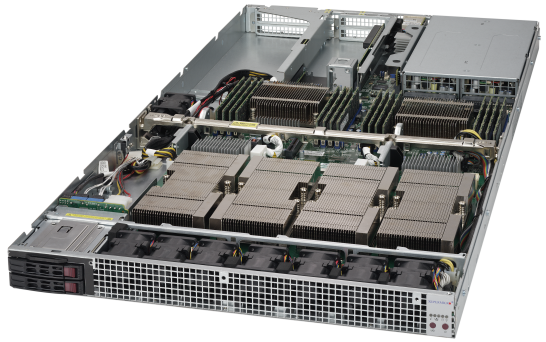


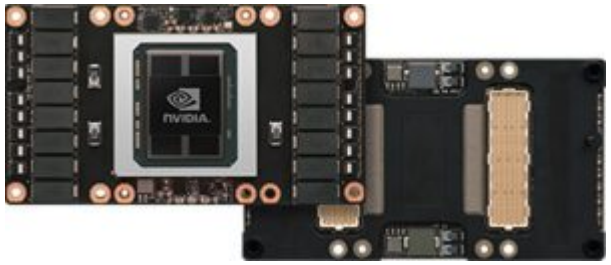
GPU cloud環境

新規サービス

https://gpu.cloud.gmo.jp/

GPU cloud SPEC





- GPU node
  - Supermicro
    - NVIDIA V100 NVLink™ model x1 socket, x2 socket, x4 socket
    - 2 Processors Intel
    - SSD local drive
    - IPMI2 ready
- Network
  - InfiniBand EDR 100G (Mellanox) IPoIB network
  - Vxlan based multi-tenant network
- Storage
  - Lustre (DDN) storage
- Computingでは
  - (1st) Singularity
  - (2nd) KVM passthrough

当初のbaremetal deploy計画

Supermicro X11 motherboard

- No DHCP network
  - InfiniBand boot deployはできない
- IPMI(BMC) network
  - Supermicro X11世代には、Redfish™ RESTful APIがあるらしい
  - ライセンス必要、HW監視ツールもRedfish APIが必要より必須
- BMC Virtaul Media
  - RedfishからもVirtual Media使えそう
  - Ironic redfish driver
- Software RAID mirror
  - Ironicでできないだろうか?

最終的なbaremetal deploy

Supermicro X11 motherboard

- BMC shared NIC with DHCP network (default vlan)
  - Dhcp + tftp boot
- IPMI(BMC) network
  - Supermicro X11世代には、Redfish™ RESTful APIがあるらしい
  - ライセンス必要、HW監視ツールもRedfish APIが必要より必須
- BMC Virtaul Media
  - RedfishからもVirtual Media使えそう
  - Ironic redfish driver ← virtual media ?
- Software RAID mirror
  - Ironicでできないだろうか? ← software RAID ?

B) 最近のbaremetal server deploy事情

Server Hardware and deployment systems

Hardware側の状況: Redfishの今?

Redfish™ API

https://www.dmtf.org/jp/standards/redfish

- DMTF (https://dmtf.org) によって開発され、Server BMCやエンクロージャ、シャーシなどのオブジェクトをRESTful (JSON, OData)にアクセス、管理できるツールAPI
- IPMIに変わるもの (IPMI + 拡張コマンド)
- データセンタ内のすべてのコンポーネントを対象とした一貫したAPIをゴールにしている

Redfish API

- Server vendorごとに実装進捗度/ライセンスが異なるので注意
  - Lenovo :
    - APIの殆どがGET methodしか使えない (read only)
    - Virtual Mediaも状態の確認 (GET method)
  - Supermicro :
    - X11世代
      🡐 IPMIは無料だが、Redfishは追加ライセンス購入が必要

MAAS系 (Ubuntu MaaS)

Baremetal deployment and manage systems

MAAS – Metal As A Service

- Ubuntu の Canonicalが提供しているツール (latest ver. : 2.6)
- Baremetal deploy
  - Custom image : squashfs
  - Cloud-init available : yes
  - Database : PostgreSQL
  - Supported driver type : Power Type
    ipmi(default), IntelAMT, libvirt(kvm, qemu), vmware,
    VirtualBox(user driver)
  - Virtual Media boot deploy : ?? Not found
  - Layout : RAID, bcache, LVM, zfs, etc.
    - Software RAID mirror support : yes ? (md0 is OK)
    - LVM layout : yes
  - Network bond and vlan : yes

MAAS - additional

- Rack, Regionの構成でdhcpセグメントごとscale out
- Discoverty : あり
- Hardware testingがある
- Ubuntu Jujuとの親和性
- Version upが早いので、安定版をおちついて選ぶ必要あり
- Python bind : python-libmaas
- Python3 available : yes
- CLI bind : maas
- webUI available: yes
- Redfishは汎用powerドライバではなく、Intel RSDなど決め打ち

Versionあがった時にコケることがある

Ironic系 (OpenStack Ironic under cloud (TripleO)/ nova compute Ironic)

deployment systems

Ironic (OpenStack Ironic)

- OpenStack の Baremetal hypervisorツール (latest ver. : Stein Series 12.1.x)
- Baremetal deploy
  - Custom image : qcow2, raw
  - Cloud-init available : yes
  - Database : MySQL

OpenStack Ironic: driver分化(Queens)

- Pikeより動作内容ごとに、細かく設定されるように(pike以降でconfig非互換、アラートが出る)

Exp) irmc driver

- Ocata:
  - pxe_irmc : PXE(iRMC) boot and deploy.
  - iscsi_irmc : Virtaul Media boot with iscsi volume boot deploy.
  - agent_irmc : Virtual Media boot with IPA deployment
- Pike:
  - boot : irmc-virtual-media, irmc-pxe
  - console : ipmitool-socat
  - inspect : irmc
  - management : irmc
  - power : irmc

OpenStack Ironic: 削除されたdriver

- Ocata以降削除されたドライバ: (staging driverへ)
  - Wake-on-lan
  - AMT
  - iBoot
  - Libvirt
    - Python の IPMI仮想化サーバである virtualbmc をつかって検証するようになったため
      https://github.com/openstack/virtualbmc
    - Libvirtで定義したvmをIPMIで操作できるようにするservice
      https://docs.openstack.org/virtualbmc/latest/

OpenStack Ironic: Redfish driver

- Sushy : https://github.com/openstack/sushy/

https://docs.openstack.org/ironic/queens/admin/drivers/redfish.html

/etc/ironic.conf
[DEFAULT]
...
enabled_hardware_types = ipmi,redfish
enabled_power_interfaces = ipmitool,redfish
enabled_management_interfaces = ipmitool,redfish

Ironic Nodeの”driver_info”の設定が必要
ipmiとほとんど変わらない

Ironic: Redfish virtual media driver

Ironicへの実装要求はかなり前からあった (2015)

https://blueprints.launchpad.net/ironic/+spec/redfish-virtual-media-boot

- Dell / Fujitsu / Supermicroなど、Virtual Media がサポートするストレージの種類が異なっていたりする。それにより、read(GET) only APIである場合が多かった
- Exp)
  - Dell : nfs://, http(s)://
  - Supermicro : smb://
- Dell ：PowerEdge 14G, iDRAC9よりPOST APIが使える

Ironic: Redfish virtual media driver

- では、Supermicro X11 系ではどうか

Virtual Media Storageが smb://

- うまくマウントできない

➔ 今回のGPU cloudへの適用では、Virtual Media boot driver作ることは、一旦断念

Ironic: Linux software RAID support

- 現在のIronicの実装はHardware RAIDのみ

- Image deploy，grub2処理，disk検知(md0 … mdX対応)，cleaning時のmd情報の削除など、
  - IPA (Ironic Python Agent)
  - ironic-conductor

に修正が必要

(specs) https://specs.openstack.org/openstack/ironic-specs/specs/not-implemented/software-raid.html

Ironic: Linux software RAID support #2

CERN がpatch作っていた

- Support for Software RAID

https://specs.openstack.org/openstack/ironic-specs/specs/not-implemented/software-raid.html

そろそろ、Git master branchにmergeされるらしい

Ironic: Linux software RAID support #3

CERN patch

Ironic patch

https://github.com/cernops/ironic/commit/581e65f1d8986ac3e859678cb9aadd5a5b06ba60

IPA patch

https://github.com/cernops/ironic-python-agent/commit/bddac76c4d100af0103a6bc08b81dd71681a9c02

https://www.slideshare.net/ArneWiebalck/integrating-baremetal-provisioning-into-cerns-private-cloud

Support for deploy-time s/w RAID

Clip slide

- Vast majority of the (compute) servers in the CERN DCs use software RAID
  - compute: RAID-0, e.g. batch farm, elastic search
  - services: RAID-1 & RAID-10, e.g. hypervisors

- Ironic server instantiation is a 2-step process for now
  - Via OpenStack (to get hold of a node and have it registered)
  - Via PXE to apply custom RAID config

→ Support for this be very welcome!

- WIP: `CERNHardwareManager`
  - Basic support for s/w RAID
  - Detect all devices and create one array of desired type
  - No partition support, no disk subsets, single level

→ Almost works ☺

Bifrost + Ironic stand-alone

deployment systems

https://docs.openstack.org/bifrost/stein/

Bifrost :https://github.com/openstack/bifrost

- Ansibleにてstand-alone Ironicをインストール、csvなどのinventoryからIronic nodeの追加とimage deployをするツール郡

Branch: master ▾ bifrost / playbooks / inventory / baremetal.csv.example Find file Copy path

dtantsur Remove support for classic drivers 212d25a on 26 Jun 2018

2 contributors

8 lines (7 sloc) | 944 Bytes Raw Blame History

```
00:1c:ab:8a:97:eb,root,undefined,192.168.122.1,1,8192,512,Control,VM,a8cb6624-0d9f-c882-affc-046ebb96ec01,hostname0,192.168.1.2,,,,,ipmi
00:2b:b7:65:83:19,root,undefined,192.168.122.1,1,8192,512,Control,VM,a8cb6624-0d9f-c882-affc-046ebb96ec02,hostname1,192.168.1.3,,,,,ipmi
00:3a:ca:56:7d:2e,root,undefined,192.168.122.1,1,8192,512,Control,VM,a8cb6624-0d9f-c882-affc-046ebb96ec03,hostname2,192.168.1.4,,,,
00:4e:d5:45:6f:31,root,undefined,192.168.122.1,1,8192,512,SwiftStorage,VM,a8cb6624-0d9f-c882-affc-046ebb96ec04,hostname3,192.168.1.5,,,,
00:5a:ed:39:57:31,root,undefined,192.168.122.1,1,8192,512,SwiftStorage,VM,a8cb6624-0d9f-c882-affc-046ebb96ec05,hostname4,192.168.1.6,,,,
00:6a:f8:2b:41:35,root,undefined,192.168.122.1,1,8192,512,Compute,VM,a8cb6624-0d9f-c882-affc-046ebb96ec06,hostname5,192.168.1.7,,,,
00:7f:9c:11:38:17,root,undefined,192.168.122.1,1,8192,512,Compute,VM,a8cb6624-0d9f-c882-affc-046ebb96ec07,hostname6,192.168.1.8,,,,
```

Bifrost :https://github.com/openstack/bifrost

- Bifrost Ironic standaloneをCIすることによって、Ironicの構築部分の検証までできるtest例もついている
  - CI用には、Vagrant_dev_env、Virsh_dev_env がついているので、これを利用して、Ironicのdriverなどの開発環境、deploy検証環境などをlocal PC/Serverに作ることもできる

Branch: master ▾ bifrost / tools / Create new file | Upload files | Find file | History

Jenkins and openstack-gerrit Merge "Centralize user documentation" Latest commit a8e2352 on 11 Aug 2017

..		
vagrant_dev_env	Fix tools/vagrant_dev_env/vagrant.yml	2 years ago
virsh_dev_env	Centralize user documentation	2 years ago
README.md	Add instructions to deploy bifrost on virsh	3 years ago

Bifrost : install and setup

- Kolla-ansible経由でのインストール
  - https://docs.openstack.org/kolla-ansible/latest/reference/deployment-and-bootstrapping/bifrost.html

  動作と設定が確定すればこちらでもok : dockerで起動

- Bifrostでのscript/bootstrapでansibleインストール
  - https://docs.openstack.org/bifrost/latest/install/index.html

  通常のサービスプロセスで起動

C) 今回のNetwork/Server構成から適用を考える

GPU cloudのbaremetal nodeへの適用

1) Redfish™の充実度は?

Which of the MAAS or Ironic is better with the Redfish API

1) Redfish API充実度

- MAAS
  - Intel RSD (Intel Rack Scale Design)など限定的

- Ironic
  - Redfish driverはdefault化(Pike)
  - だが、Redfish Virtual Media boot driverはまだ実装されてない
    - Redfish virtual media API自体がまだ一般化していない

➔ 現時点では、無理をしてRedfish APIを使う必要はない

2) Dhcp無しで固定IPのVirtual media deployができるか?

Can I deploy with fixed IP virtual media boot without using dhcp?

2) Dhcpを利用せずに固定IPのvirtual media boot deployは可能か?

- MAAS
  - Virtual Media bootができない

- Ironic
  - iLo, iRMCなどではできそうだ(要検証)
  - だが、Redfish Virtual Media boot driverはまだ実装されてない
    - Redfish virtual media API自体がまだ一般化していない

➔現時点では、無理をしてRedfish APIを使う必要はない

➔Dhcpサーバが必須となる

3) Support for Linux software RAID?

Support for Linux software RAID?

3) Linux software RAID support?

- MAAS
  - Storage Layoutでsupport

- Ironic
  - 開発中 (git master branch)
    - Merge from CERN patch
  - 利用する場合には、Ironicにpatchを当てる

➔CERN patchのupstreamへのmergeをIronicで待っても良い

➔台数が少ないうちは、script/ansibleでOS領域をs/w RAID化

4) どのBeremetal deploy系を使うのか(MAAS or Ironic)?

Which of the MAAS or Ironic baremetal deploy to use?

4) どちらのBaremetal deployを使うの?

- MAAS
  - Redfish support弱い
  - Virtual Media boot deploy supportがない

- Ironic
  - Junoでも利用していて、Ironic知見がある
  - 利用する場合には、Standalone利用で、サーバ構築に特化させる
  - s/w RAIDの対応は台数が増えたときにpatchあてる

➔s/w RAID化、Redfish導入、など将来性からIronic に

➔Standaloneで利用するので、Bifrostで導入し、Ansible flow化させる

D) Bifrost Ironic standalone deploy

GPU cloudのbaremetal nodeへの適用

Bifrost install and setup

deployment systems

https://docs.openstack.org/bifrost/stein/

https://docs.openstack.org/bifrost/stein/

OpenStack Docs: | bifrost Ironic - Go | OpenStack Docs: | Ironic Software R/ | Automate bare m | Merge "Software | Google 翻訳

https://docs.openstack.org/bifrost/stein/install/index.html

This is the current supported release. Back to Top

OpenStack Documentation

bifrost

Page Contents

Bifrost Installation

THIS PAGE LAST UPDATED: 2019-04-21 12:28:12

Introduction

Installation and use of bifrost is split into roughly three steps:

- **install**: prepare the local environment by downloading and/or building machine images, and installing and configuring the necessary services.
- **enroll-dynamic**: take as input a customizable hardware inventory file and enroll the listed hardware with ironic, configuring each appropriately for deployment with the previously-downloaded images.
- **deploy-dynamic**: instruct ironic to deploy the operating system onto each machine.

Supported operating systems:

- Ubuntu 14.04, 14.10, 15.04, 16.04
- Red Hat Enterprise Linux (RHEL) 7
- CentOS 7
- Fedora 22
- openSUSE Leap 42.1, 42.2

Installation

Pre-install steps

Installing bifrost on RHEL or CentOS requires a few extra pre-install steps, in order to have access to the additional packages contained in the EPEL repository. Some of the software bifrost leverages, can only be obtained from EPEL on RHEL and CentOS systems.

> **Note**
>
> Use of EPEL repositories may result in incompatible packages being installed by the package manager. Care should be taken when using a system with EPEL enabled.

Bifrost install

(1) my環境
CentOS 7.5.1804

(2) install
# su – baremetal　# setup user : exp) baremetal
$ sudo yum install epel-release
$ git clone https://opendev.org/openstack/bifrost　# clone / checkout taged
$ cd bifrost
$ scripts/env-setup.sh　# ansible install to $HOME/.local dir
$ echo ‘export PATH=$PATH:$HOME/.local/bin:$HOME/bin’ ¥
　>> ~/.bash_profile
$ source ~/.bash_profile

Bifrost install #2

(3) pre-installation settings : inventory/group_vars/baremetal

```
[baremetal@seed-server11001 ~]$ cat bifrost/playbooks/inventory/group_vars/baremetal | grep -v '^$' | grep -v '^#'
---
network_interface: "eth1"
ssh_public_key_path: "{{ lookup('env', 'HOME') }}/.ssh/id_rsa.pub"
ssh_public_key_2nd_path: "{{ lookup('env', 'HOME') }}/.ssh/authorized_keys"
testing_user: root
ipa_kernel: "{{http_boot_folder}}/ipa.image-custom-f29mini-net.vmlinuz"
ipa_ramdisk: "{{http_boot_folder}}/ipa.image-custom-f29mini-net.initramfs"
ipa_kernel_url: "http://172.29.129.217:8080/ipa.image-custom-f29mini-net.vmlinuz"
ipa_ramdisk_url: "http://172.29.129.217:8080/ipa.image-custom-f29mini-net.initramfs"
ironic_node_base_url: "http://172.29.129.217:8080"
http_boot_folder: /httpboot
deploy_image_filename: "xenial-cimg-amd64-ttyS1-raid1-debug.img"
deploy_image: "{{http_boot_folder}}/{{deploy_image_filename}}"
deploy_image_md5sum: "271fc360394667532833e5f68c0d220e"
wait_for_node_deploy: false
create_image_via_dib: true
transform_boot_image: false
node_default_network_interface: eth0
ipv4_subnet_mask: 255.255.254.0
ipv4_gateway: 172.29.128.1
ipv4_nameserver: 8.8.8.8
[baremetal@seed-server11001 ~]$
```

Bifrost install #3

(3) pre-installation settings : inventory/group_vars/target

```
[baremetal@seed-server11001 ~]$ cat bifrost/playbooks/inventory/group_vars/target | grep -v '^$' | grep -v '^#'
---
ironic_db_password: [redacted]3z
mysql_username: root
mysql_password:
ssh_public_key_path: "{{ lookup('env', 'HOME') }}/.ssh/id_rsa.pub"
testing: true
create_image_via_dib: true
dib_image_type: vm
transform_boot_image: false
create_ipa_image: true
```

Bifrost install #4

(3) pre-installation settings : inventory/group_vars/localhost

```
[baremetal@seed-server11001 ~]$ cat bifrost/playbooks/inventory/group_vars/localhost | grep -v '^$' | grep -v '^#'
---
ironic_db_password: a[redacted]z
mysql_username: root [redacted]
mysql_password:
create_image_via_dib: true
dib_image_type: vm
transform_boot_image: false
create_ipa_image: false
```

Bifrost install #5

(4) installing

```
$ pwd
/home/baremetal/bifrost
$bash ./scripts/env-setup.sh
$ export PATH=$HOME/.local/bin:$PATH
$ cd playbooks/

$ ansible-playbook -vvvv –I inventory/target install.yaml ¥
  -e staging_drivers_include=true
$ source ~/openrc bifrost && openstack baremetal driver list
```

Bifrost Ironic standalone config

```
[DEFAULT]
debug = True
enabled_network_interfaces = noop
default_deploy_interface = direct
enabled_inspect_interfaces = no-inspect,inspector
default_inspect_interface = inspector
enabled_boot_interfaces = pxe
enabled_management_interfaces = ipmitool
enabled_power_interfaces = ipmitool
enabled_deploy_interfaces = direct,ansible
enabled_console_interfaces = ipmitool-socat,no-console
default_console_interface = ipmitool-socat
enabled_rescue_interfaces = no-rescue,agent
default_rescue_interface = agent
enabled_raid_interfaces = agent,no-raid
enabled_hardware_types = ipmi
transport_url = rabbit://ironic:[redacted]3z@127.0.0.1:5672/
auth_strategy = noauth
```

Bifrost Ironic standalone config #2

```
[pxe]
pxe_append_params = nofb nomodeset vga=normal ipa-debug=1 console=tty1 console=/dev/ttyS1,115200n8 rootpwd="\$1\$1C8RdhnX
\$eE[redacted]vEm." biosdevname=0 net.ifnames=0 ETHERNET=eth0 ip=dhcp
pxe_config_template = $pybasedir/drivers/modules/ipxe_config.template
tftp_server = 172.29.132.217
tftp_root = /tftpboot
pxe_bootfile_name = undionly.kpxe
ipxe_enabled = true
ipxe_boot_script = /etc/ironic/boot.ipxe
tftp_master_path = /var/lib/ironic/master_images
uefi_pxe_bootfile_name = ipxe.efi
uefi_pxe_config_template = $pybasedir/drivers/modules/ipxe_config.template
```

Bifrost Ironic standalone config #3

```
pxe_config_template = $pybasedir/drivers/modules/ipxe_config.template
tftp_server = 172.29.132.217
tftp_root = /tftpboot
pxe_bootfile_name = undionly.kpxe
ipxe_enabled = true
ipxe_boot_script = /etc/ironic/boot.ipxe
tftp_master_path = /var/lib/ironic/master_images
uefi_pxe_bootfile_name = ipxe.efi
uefi_pxe_config_template = $pybasedir/drivers/modules/ipxe_config.template
[deploy]
http_url = http://172.29.132.217:8080/
http_root = /httpboot
default_boot_option = local
```

Bifrost Ironic standalone config #4

```
[deploy]
http_url = http://172.29.132.217:8080/
http_root = /httpboot
default_boot_option = local
[ansible]
verbosity = 3
ansible_playbook_script = ansible-playbook
playbooks_path = $pybasedir/drivers/modules/ansible/playbooks
config_file_path = $pybasedir/drivers/modules/ansible/playbooks/ansible.cfg
post_deploy_get_power_state_retries = 6
post_deploy_get_power_state_retry_interval = 5
force_power_state_during_sync = false
default_username = ansible
default_deploy_playbook = deploy.yaml
default_shutdown_playbook = shutdown.yaml
default_clean_playbook = clean.yaml
[conductor]
```

Bifrost Ironic standalone config #5

```
[conductor]
clean_nodes = false
automated_clean = false
[database]
connection = mysql+pymysql://ironic:[redacted]3z@localhost/ironic?charset=utf8
[dhcp]
dhcp_provider = none
[ilo]
use_web_server_for_images = true
[inspector]
enabled = true
auth_type=none
endpoint_override=http://127.0.0.1:5050
[service_catalog]
auth_type = none
endpoint_override = http://172.29.132.217:6385
```

最終的なbaremetal deploy (fin)

Supermicro X11 motherboard

- BMC shared NIC with DHCP network (default vlan)
  - Dhcp + tftp boot
- IPMI(BMC) network ← 今回はIronicでは”IPMI”ドライバ利用
  - Supermicro X11世代には、Redfish™ RESTful APIがあるらしい
  - ライセンス必要、HW監視ツールもRedfish APIが必要より必須
- BMC Virtaul Media
  - RedfishからもVirtual Media使えそう
  - Ironic redfish driver ← virtual media ? ← 今回はdhcp boot deploy
- Software RAID mirror
  - Ironicでできないだろうか? ← software RAID ?← Ironic upgrade待ち

summary

- Case by caseだが、今回の我々の始まったばかりのGPUcloud環境のように、ansible workflowが主体な環境において、Bifrost + Ironic Standaloneは運用適合性がよかった
- IronicへのSoftwareRAID適用は、Ironic側のドライバのupstream対応待ちであり、今後のupgrade待ちということになった