

6. Networking

宮崎 清人

6章の目次

本章では, OpenStackにおけるネットワークの概要と設定方法について説明する

6.1 ネットワークのオプション

6.2 Cloudpipe – ProjectごとのVPN

6.3 Compute Node上でのネットワーク設定

6.4 Projectからのネットワークの除去

6章の目次

6.1 ネットワークのオプション

6.2 Cloudpipe – ProjectごとのVPN

6.3 Compute Node上でのネットワーク設定

6.4 Projectからのネットワークの除去

6.1 ネットワークのオプション (1)

<Project>

ユーザの作成したInstanceをまとめたもの

各InstanceにはComputeがプライベートIPアドレスを割り当てる

<補足> 現在, NovaはLinuxのbridgeネットワークのみをサポートする

<Network Controller>

仮想ネットワークを提供し, computeのサーバがサーバ同士あるいは外部との通信を可能にする

6.1 ネットワークのオプション (2)

現在, Novaは3種類のNetwork Modeをサポートし,
それぞれNetwork Managerとして実装

<NovaにおけるNetwork Mode>

(1) Flat Network Mode

最もシンプルなモード

(2) Flat DHCP Network Mode

DHCPによりIPアドレスを割り当てるモード

(3) VLAN Network Mode

VLANを構築するモード

現在, 1つのComputeに対して複数のネットワークを設定することはできない

6.1. ネットワークのオプション – Flat Mode(1/2)

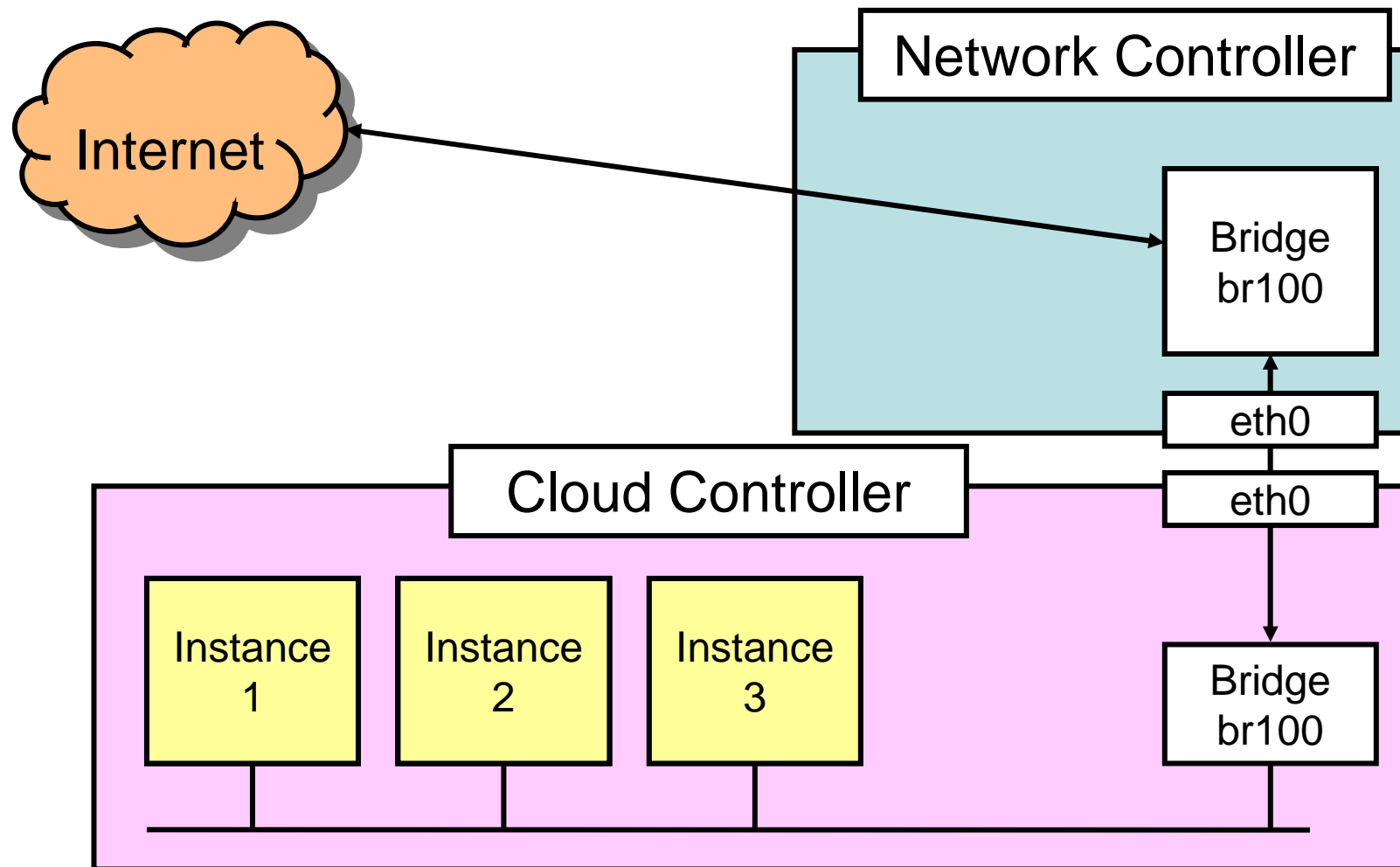
(1) Flat Mode

<特徴>

- (A) 管理者の設定したサブネットから, Instanceに固定IPが割り当てられる
- (B) novaはネットワーク管理を全く行わない
- (C) 仮想マシン Imageの起動時にファイルシステムを通じてIPを割り当てる
- (D) すべてのInstanceは1つのブリッジに接続される
- (E) network nodeはデフォルトゲートウェイの役割をせず, InstanceにはパブリックIPが割り当てられる

6.1. ネットワークのオプション – Flat Mode(2/2)

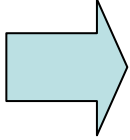
<概略図>



6.1 ネットワークのオプション – Flat DHCP Mode (1/2)

(2) Flat DHCP Mode

<特徴>

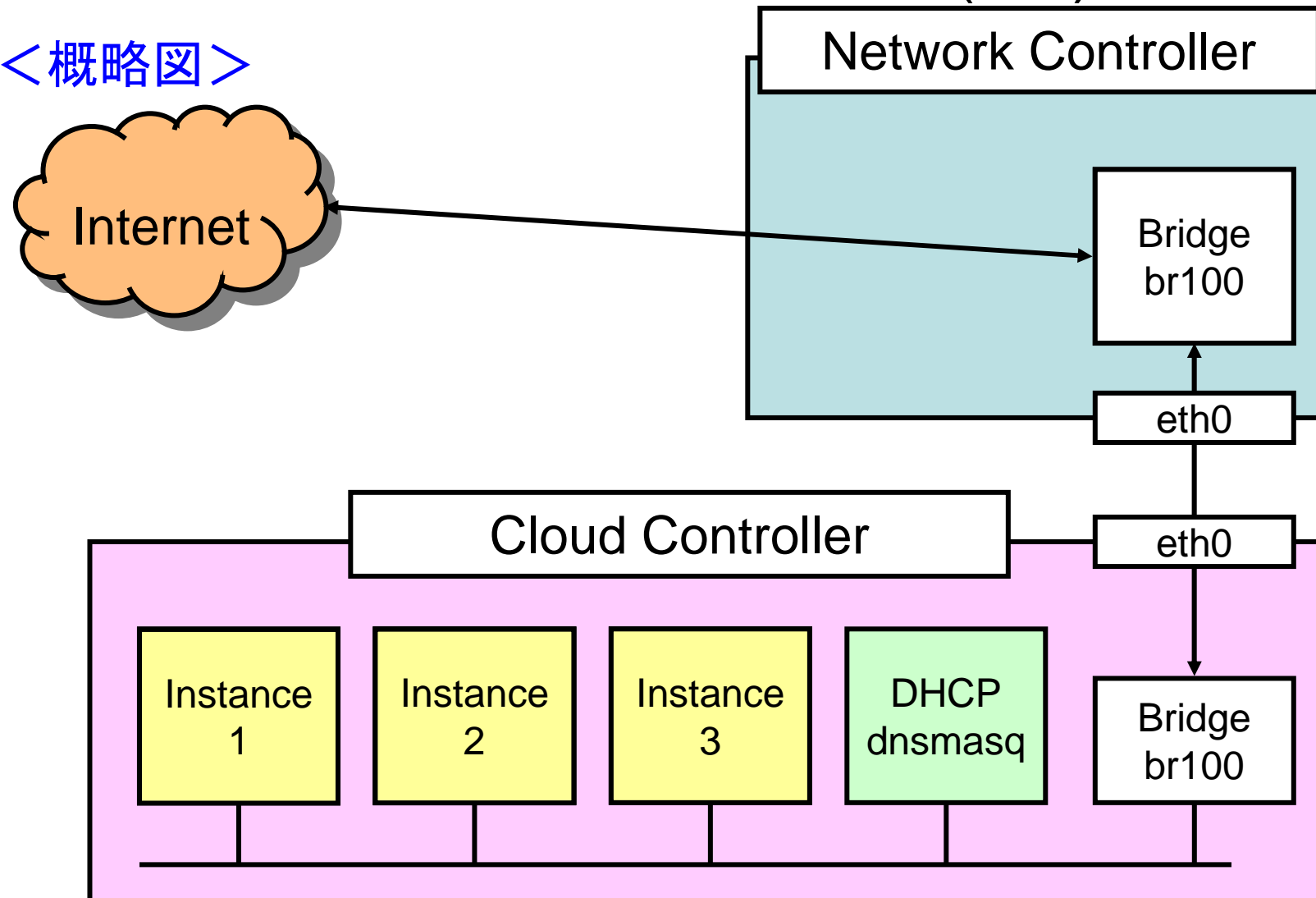
- (A) DHCPサーバによりInstanceにIPアドレスを割り当てる
- (B) ComputeはEthernetデバイスへとブリッジするためにより多くの設定を行う
- (C) DHCPサーバとしてdnsmasqを走行させ、ブリッジ上でlistenを行う
 -  Instanceはdhcpdiscoverをすることで固定IPを受け取る

<Flat Network Modeとの共通点>

network nodeはデフォルトゲートウェイの役割をしない
InstanceにはパブリックIPが割り当てられる

6.1 ネットワークのオプション – Flat DHCP Mode (2/2)

<概略図>

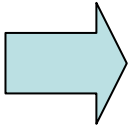


6.1 ネットワークのオプション

– VLAN Network Mode (1/2)

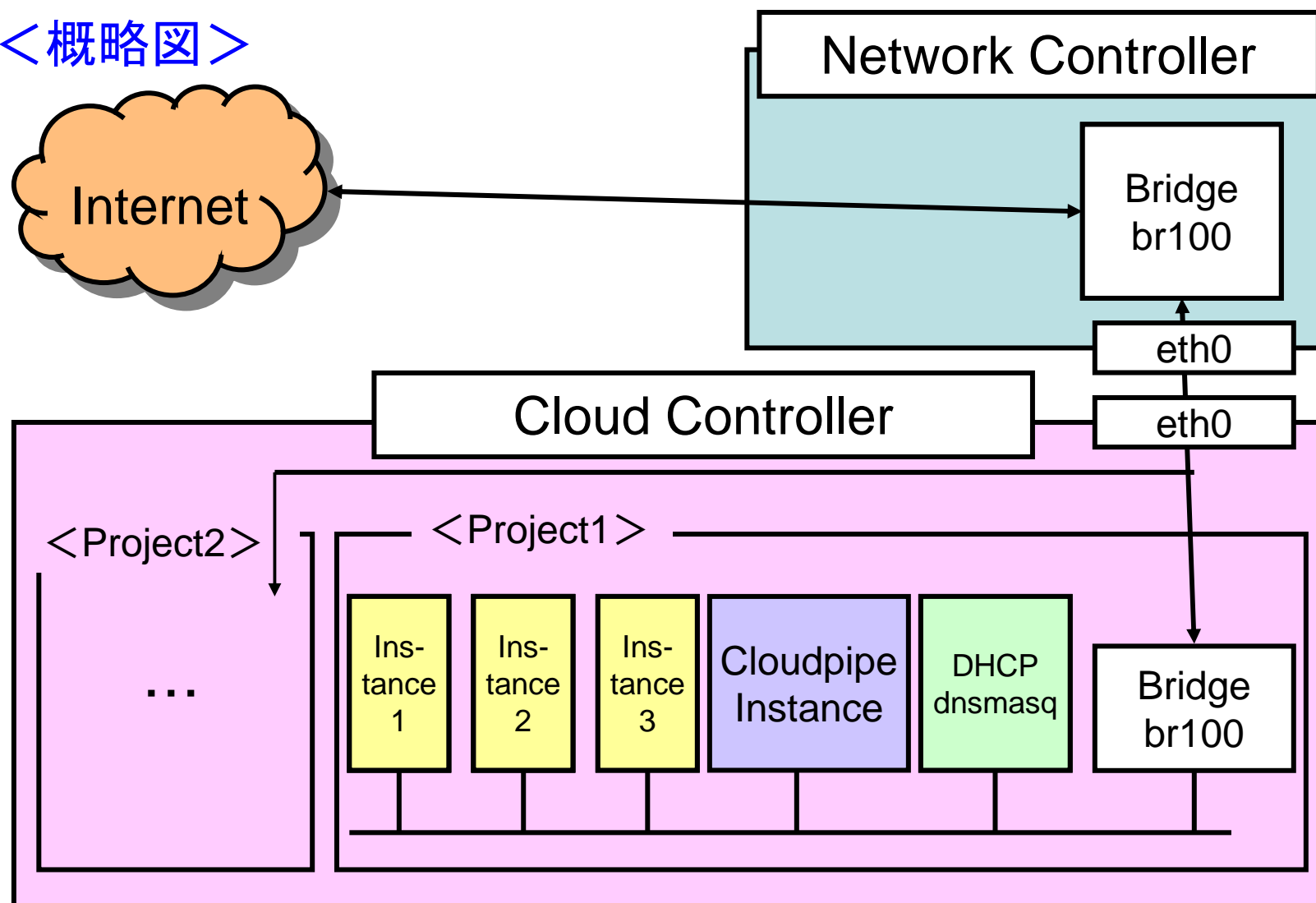
(3) VLAN Network Mode (デフォルト)

<特徴>

- (A) ProjectごとにVLAN, ブリッジ, およびDHCPサーバをもつ
- (B) Projectは, VLAN内部からのみアクセス可能なプライベートIPアドレスの範囲を得る
 -  VLAN外部からInstanceにアクセスするためには
Cloudpipe (後述) が必要
- (C) ComputeはVPNにアクセスするための証明書と鍵を生成し, VPNを自動的に開始する
- (D) スイッチはhost-managed VLAN taggingに対応している必要がある

6.1. ネットワークのオプション – VLAN Network Mode (2/2)

<概略図>



6章の目次

6.1 ネットワークのオプション

6.2 Cloudpipe – ProjectごとのVPN

6.3 Compute Node上でのネットワーク設定

6.4 Projectからのネットワークの除去

6.2. Cloudpipe – ProjectごとのVPN(1/3)

VLAN Network ModeにおいてInstanceに割り当てられるプライベートIPには, VLAN内部からのみアクセス可能

➡ VLANの外部からInstanceにアクセス可能にするため, **Cloudpipe**を用意

<Cloudpipe>

VLAN Network Modeにおいて, ユーザとInstanceを接続するための特殊なVPN Instance

VPNにより, Instanceをインターネット上にさらすことなく, ユーザはProject中のInstanceに自由にアクセスできる

6.2. Cloudpipe – ProjectごとのVPN(2/3)

Cloudpipe Imageの実体は, **openvpnがインストールされたLinux Instance**である

<Cloudpipe Imageに必要なもの>

以下のような処理を行うスクリプト

- (1) メタデータサーバからユーザデータを取得
- (2) 取得したユーザデータをBase64で復号してZIPファイルに変更
- (3) ZIPファイルの中にあるautorun.shを実行

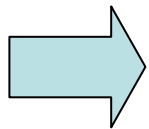
<autorun.sh>

openvpnを設定し, 実行するスクリプト

6.2. Cloudpipe – ProjectごとのVPN(3/3)

<補足>

定期的にメタデータのダウンロードと新しい証明書リストのダウンロードを行うcronスクリプトを用意すると役立つ



- ・無効になったユーザーの接続を禁止
- ・証明書が無効になったユーザーの切断

6.2.1. Cloudpipe Imageの作成(1/2)

以下のような手順でCloudpipe Imageを作成する

<手順>

- (1) ベースとするubuntuイメージにopenvpnをインストールする
- (2) /etc/openvpn/server.conf.templateを設定する
- (3) /etc/openvpn/up.shを設定する
- (4) /etc/openvpn/down.shを設定する
- (5) /etc/network/interfacesを設定する
- (6) フラグファイルに Imageを登録し, ImageIDを設定する

```
--vpn_image_id=ami-xxxxxxxxx
```

- (7) その他のフラグを設定し, VPNが正しく動作するようにする

```
--use_project_ca  
--cnt_vpn_clients=5
```


6.2.1. Cloudpipe Imageの作成

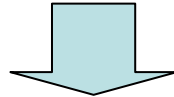
Nova-managerを使ってCloudpipeを起動する場合の手順を以下に示す

<手順>

- (1) <project_id>-vpnという名前のキーペアを作成し、キーのディレクトリに保存
- (2) <project_id>-vpnという名前のセキュリティグループを作成し、1194番ポートとicmpを開放
- (3) VPN Instanceに証明書と秘密鍵を作成し、CA/projects/<project_id>ディレクトリに保存
- (4) 証明書と秘密鍵をzip圧縮し、ユーザデータとしてb64エンコードする
- (5) フラグファイルで指定したとおりのImageにより、以上の設定でm1.tiny Instanceを起動する

6.2.2. VPN アクセス

VLAN network modeでは、各プライベートネットワークの2番目のIPは、Cloudpipe Instanceのために予約されている

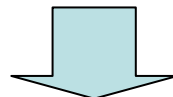


Cloudpipe Instanceに一貫したIPが与えられ、Novaネットワークが外部からのアクセスに対する転送ルールを作成可能

Projectごとのネットワークに、ネットワークホストのパブリックIPの特定のポート(高い番号)が与えられる



このポートは自動的にVPN Instanceの1194番ポートに転送される



ユーザが外部のネットワークから Instanceにアクセス可能

6.2.3. 認証と認証失効

`usr_project_ca`フラグをセットすると、各Projectが各々の認証局をもつように設定できる

➡ Couldpipeが安全に動作するために必要

Nova-manageを使っていて証明書が無効にされた場合、新しいCRLが作られている

➡ 無効されたユーザはVPN接続できなくなる

<注意>

現状では証明書が無効にされてもユーザデータは更新されない

➡ ユーザの証明書が無効になったとき
Cloudpipeインスタンスの再起動が必要

6.2.4. Cloudpipe VPNの再起動(1/2)

<Cloudpipeの再起動>

不具合が生じた場合、以下のようにCloudpipeの再起動が可能

```
euca-reboot-instances
```

ただし、新しいCRLを生成する場合は以下のような手順が必要

(1) Cloudpipeインスタンスを終了する

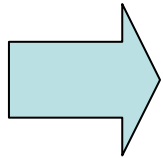
(2) Cloudpipeインスタンスを起動する

```
nova-manage vpn run <project_id>
```

6.2.4. Cloudpipe VPNの再起動(2/2)

<補足>

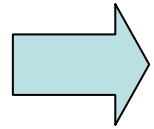
再起動後, CloudpipeインスタンスがIPアドレスを取得するため, 最大で10分程度の時間がかかる



再起動時間を許容できない場合は, 手動でIPを更新することも可能である

6.2.5. Cloudpipe VPNへのログイン

Cloudpipe Instanceを起動するために使用したキーペアを,
keys/<project_id>に置く



このキーを使って、デバッグのために
Cloudpipe Instanceにログイン可能である

6章の目次

6.1 ネットワークのオプション

6.2 Cloudpipe – ProjectごとのVPN

6.3 Compute Node上でのネットワーク設定

6.4 Projectからのネットワークの除去

6.3 Compute Node上でのネットワーク設定(1/2)

Compute Nodeのネットワーク設定の手順の概要を以下に示す

<手順>

(1) nova.confの-network-managerフラグをセットする

```
network_manager=nova.network.manager.FlatManager
```

(2) VMが使用するサブネットを作成する

```
nova-manage network create CIDR n n
```

(3) ブリッジとネットワークを統合

6.3 Compute Node上でのネットワーク設定(2/2)

<補足>

ComputeのNetwork Modeは, nova.confにおいて
以下のようなフラグで指定する

(1) Flat Mode

```
network_manager=nova.network.manager.FlatManager
```

(2) Flat DHCP Mode

```
network_manager=nova.network.manager.FlatDHCPManager
```

(3) VLAN Mode(デフォルト)

```
network_manager=nova.network.manager.VlanManager
```

6.3.1. Flat Networkの設定(1/3)

Flat Network Modeでのネットワーク設定の手順を以下に示す

<手順>

(1) Network Modeの確認

nova.confに次の行が入っていることを確認

```
--network_manager=nova.network.manager.FlatManager
```

(2) ブリッジデバイスの名前の設定

ブリッジデバイスの名前はデフォルトでbr100となっている
これを変更したい場合はNova databaseを編集する

6.3.1. Flat Networkの設定(2/3)

(3) /etc/network/interfacesの編集

Compute Nodeをブリッジに接続するため, 以下の例を参考に/etc/network/interfacesを編集

```
# The loopback network interface
Auto lo
Iface lo inet loopback

#Networking for OpenStack Compute
Auto br100
Iface br100 inet dhcp
        bridge_ports      eth0
        bridge_step off
        bridge_maxwait    0
        bridge_fd
```

6.3.1. Flat Networkの設定(3/3)

(4) ネットワークの再起動

```
$ sudo /etc/init.d/networking restart
```

6.3.2 Flat DHCP Networkの設定(1/2)

Flat DHCP Network Modeでのネットワーク設定の手順を以下に示す

<注意>

以下の手順は, nova-networkが走行しているホストに, IPの割り当てられていないインターフェースがある場合の手順である

すでにIPを割り当てられたインターフェースは指定しないこと！
(SSH接続が不能になる)

6.3.2 Flat DHCP Networkの設定(2/2)

<手順>

- (1) nova-network が走行しているホストの指定
nova-computeが走行しているホスト上のnova.confの以下のフラグを編集

```
--network_host
```

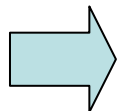
- (2) Flat interfaceの設定
ブリッジするデバイスを設定する

```
--flat_interface=<interface>
```

6.3.3. VLAN Networkの設定(1/8)

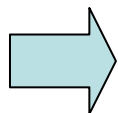
<VLANの特徴>

(1) 大きなIPアドレス空間を, より小さな**サブネットに分割**



ブロードキャスト範囲を制御できる

(2) サブネット同士は**スイッチレベルで結合**



このIPアドレス空間内の全てのマシンが通信可

VLAN環境では, 各VLANを論理的に分割する方法として
Projectを使うことで, OpenStackを構築できる

<注意>

VLAN ModeではIPフォワーディングを有効にする必要がある

6.3.3. VLAN Networkの設定(2/8)

<必要なもの>

VLAN Networkを使用するには、以下のものが必要である

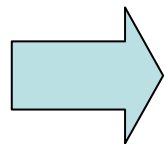
(1) 各ネットワークのパラメータ

ネットマスク, ブロードキャスト, ゲートウェイ,
Ethernetデバイス, VLAN ID

(2) VLAN Taggingに対応したネットワークハードウェア

<注意>

デフォルトフラグのvlan_interfaceはeth0でハードコードされている



ブリッジにeth0以外のデバイスを接続する場合,
以下のようにnova.confを編集する

```
--vlan_interface=eth1
```


6.3.3. VLAN Networkの設定(3/8)

VLANネットワークを設定するための前準備について、
以下で説明する

<前準備>

(0) ComputeのネットワークモードがVLANになっていることを
確認する

```
</etc/nova/nova.conf>  
--network_manager=nova.network.manager.VlanManager
```

VLAN Network ModeはComputeのネットワークモード
のデフォルトであるため、この行はなくてもよい

6.3.3. VLAN Networkの設定(4/8)

VLAN Networkを構築する手順について、以下で説明する

<手順>

(1) ネットワークを作成する. 以下はコマンドの例である

```
nova-manage --flagfile=/etc/nova/nova.conf  
network create 10.1.171.0/24 1 256  
nova-manage --flagfile=/etc/nova/nova.conf  
network create 10.1.172.0/24 1 256  
nova-manage --flagfile=/etc/nova/nova.conf  
network create 10.1.173.0/24 1 256  
nova-manage --flagfile=/etc/nova/nova.conf  
network create 10.1.174.0/24 1 256
```

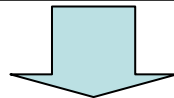
6.3.3. VLAN Networkの設定(5/8)

- (2) novaデータベースにログインし、各VLANに割り当てられたIDを決定する

```
select id,cidr from networks;
```

- (3) ネットワーク設定と合致するようにデータベースを更新する

OPENSTACK COMPUTE MANUALの6.3.3.の例を参照し、スクリプトを作成



スクリプトの動作確認後、スクリプトをnovaデータベースに対して各VLAN環境ごとに走行させる

6.3.3. VLAN Networkの設定(6/8)

(4) ComputeProjectのProjectマネージャを作成する

```
nova-manage --flagfile=/etc/nova/nova.conf  
user admin $username
```

(5) Projectを作成し, ユーザをadminユーザとして指定する

```
nova-manage --flagfile=/etc/nova/nova.conf  
project create $projectname $username
```

(6) 作成したユーザに権限を取得する

```
nova-manage --flagfile=/etc/nova/nova.conf  
project zipfile $projectname $username
```

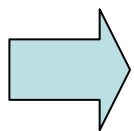
6.3.3. VLAN Networkの設定(7/8)

<補足1>

新しくVMを作成するには、そのInstanceがどのVLANに属するかを決定し、**対応するProject内でInstanceを開始**する必要がある

<補足2>

場合によっては、ネットワークマネージャが停止したとき、ブリッジデバイスとVLANタグが適切に取り外されないことがある



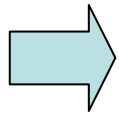
これによってネットワークマネージャの再起動に失敗する場合は、ブリッジとVLANを手動で取り外す

```
vconfig rem vlanNNN  
ifconfig br_NNN down  
brctr delbr br_NNN
```

6.3.3. VLAN Networkの設定(8/8)

< Instanceへのアクセス >

ユーザがProject中の InstanceにVLANを通じてアクセスする必要がある場合, Cloudpipeの作成が必要



6.2節を参照

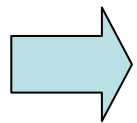
6.3.4. VM上でのPingとSSHの有効化

VM上でPingとSSHを有効化するには、以下のコマンドを入力

```
euca-authorize -P icmp -t -1:-1 default  
euca-authorize tcp -p 22 default
```

<注意>

以上のコマンドを実行してもPing(またはSSH)が有効にならない場合



走行中のdnsmasqプロセスの数を確認

プロセスの数が2でなければ次のコマンドを実行

```
killall dnsmasq  
service nova-network restart
```

6.3.5 IPアドレスと Instanceの関連付け (1/3)

Flat DHCP ModeまたはVLAN Modeで使われる固定IPの管理が必要  **Euca2ools**を使用

<IPアドレスの関連付け>

利用可能な浮動IPアドレスをアドレスプールから取得して取り除き, Instanceに割り当てる手順を以下に示す

- (1) 浮動IPアドレスをアドレスプールから取得し, Projectに割り当てる

```
euca-allocate-address
```

- (2) 浮動IPアドレスを Instanceに割り当てる

```
euca-associate-address -i [instance_id] [floating_ip]
```


6.3.5 IPアドレスと Instanceの関連付け (2/3)

<IPアドレスの関連付けの解除>

IPアドレスと Instanceとの関連付けを解除し, IPアドレスをアドレスプールに返す手順を以下に示す

(1) IPアドレスと Instanceの関連付けを解除する

```
euca-disassociate-address [floating_ip]
```

(2) IPアドレスをプールに戻す

```
euca-deallocate-address [floating_ip]
```

6.3.5 IPアドレスと Instanceの関連付け (3/3)

<補足>

Euca2oolsと同様に, nova-manageコマンドにも
浮動IPアドレスを管理する機能がある

(1) 浮動IPアドレスプールにあるIPアドレスを表示する

```
nova-manage floating list
```

(2) 特定のネットワークホストと, アドレスまたはサブネットに対して浮動IPアドレスを作成する

```
nova-manage floating create [hostname] [cidr]
```

(3) 浮動IPアドレスを取り除く(パラメータはcreateと同様)

```
nova-manage floating destroy [hostname] [cidr]
```

6.3.6. パブリックIPアドレスの割り当て(1/5)

NATによってVM InstanceにパブリックIPを利用するには、
以下の手順で設定を行う

<前準備>

(1) nova.confの編集

nova.confの—public_interfaceの行を次のように書き換える

```
--public_interface=vlan100
```

(2) nova-networkを再起動する

(nova-networkの実行中に(1)の手順を行った場合)

(3) 22番ポートが開放されていることを確認する

6.3.6. パブリックIPアドレスの割り当て(2/5)

パブリックIPアドレスを Instanceに割り当て, 利用可能にする手順を以下に示す

<手順>

- (1)パブリックIPアドレス(またはそのブロック)を
浮動IPアドレスリストに追加する

```
nova-manage floating create my-hostname  
68.99.26.170/31
```

※ 68.99.26.170/31は追加するパブリックIPアドレスのブロック

- (2) 追加したパブリックIPアドレスを Instanceに割り当てる

```
euca-allocate-address 68.99.26.170  
euca-associate-address -i i-1 68.99.26.170
```

6.3.6. パブリックIPアドレスの割り当て(3/5)

(3) セキュリティグループが解放されていることを確認する

```
root@my-hostname:~# euca-describe-groups  
GROUP admin-project default default  
PERMISSION admin-project default ALLOWS icmp -1 -1  
FROM CIDR 0.0.0.0/0  
PERMISSION admin-project default ALLOWS tcp 22 22  
FROM CIDR 0.0.0.0/0
```

6.3.6. パブリックIPアドレスの割り当て(4/5)

(4) 以下のようなNATのルールがiptablesに追加されていることを確認する

```
-A nova-network-OUTPUT -d 68.99.26.170/32 -j DNAT --to-destination 10.0.0.3  
-A nova-network-PREROUTING -d 68.99.26.170/32 -j DNAT --to-destination 10.0.0.3  
-A nova-network-floating-snat -s 10.0.0.3/32 -j SNAT --to-source 68.99.26.170
```

6.3.6. パブリックIPアドレスの割り当て(5/5)

(5) パブリックインタフェースにパブリックIPアドレスが追加されていることを確認する

```
$ ip addr
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu
1500 qdisc mq state UP qlen 1000
link/ether xx:xx:xx:17:4b:c2 brd ff:ff:ff:ff:ff:ff
inet 13.22.194.80/24 brd 13.22.194.255 scope global eth0
inet 68.99.26.170/32 scope global eth0
inet6 fe80::82b:2bf:fe1:4b2/64 scope link
valid_lft forever preferred_lft forever
```

注意: 同じサーバ内で, パブリックIPにより InstanceにSSH
接続することはできない

6章の目次

6.1 ネットワークのオプション

6.2 Cloudpipe – ProjectごとのVPN

6.3 Compute Node上でのネットワーク設定

6.4 Projectからのネットワークの除去

6.4 Projectからのネットワークの削除

Projectと関連付けられたネットワークを単純に削除することはできない

➡ scrubコマンドによりProjectをネットワークから切り離す

```
nova-manage project scrub projectname
```