

# Ansibleで構築する Cloud-init対応の 最強VMテンプレート

JPmoxs勉強会 #5

2024.12.11

ネットワークコンテンツ研究会 Nekko Cloud Team

いるまる

# 菊池 兼矢 a.k.a いるまる

## 技術

- Kubernetes, Proxmox VE
- Docker, ArgoCD
- PHP, Go

## 所属

- ネットワークコンテンツ研究会  
Nekko Cloud Team創設者
- 千葉工業大学 情報ネットワーク学科 学部4年

## 趣味

- アニメ, 電子工作, 電車の電気設備

GitHub: <https://github.com/irumaru>



- Nekko Cloud Teamの紹介
- Nekko CloudでのProxmox VE利用事例
- 最強VM TemplateをAnsibleで構築する
  - VM作るのが大変
  - Ansible Playbookで構築
  - 工夫した話
  - これ公開しました！

# Nekko Cloudの紹介

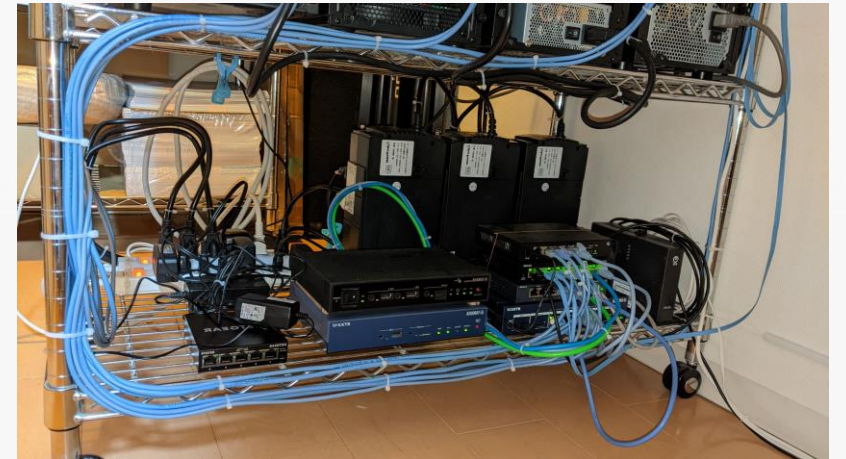
## Nekko Cloud Teamとは？

- 大学生が主要のサークル ネットワークコンテンツ研究会のクラウドチーム
- メンバーの自宅を使ってマルチリージョンプライベートクラウドを制作
- クラウドの設計・開発・運用を通してクラウド技術を学ぼう！

**NEKK**  **CLOUD**

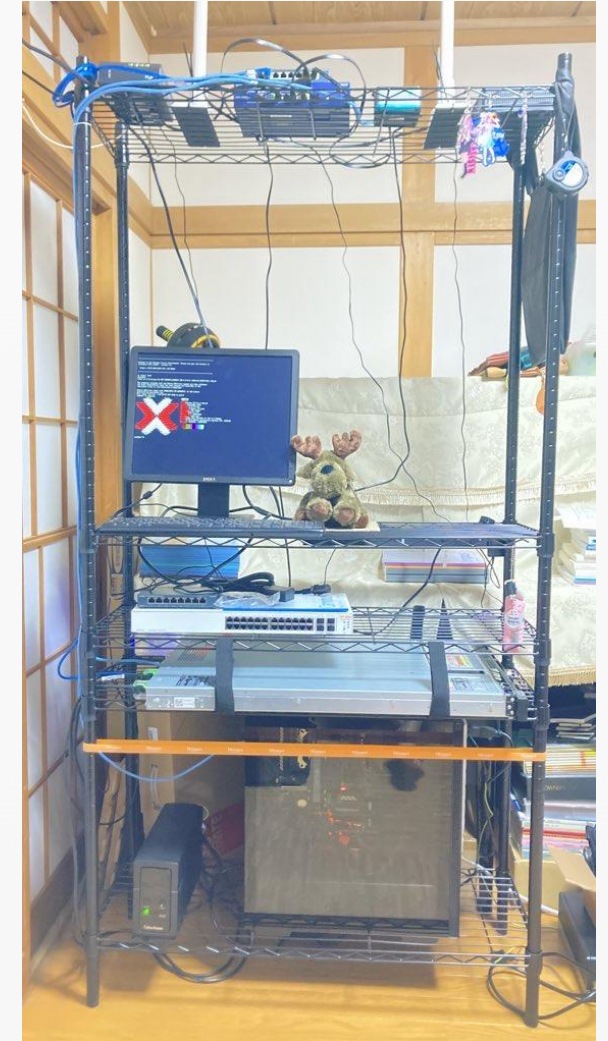
# Nekko Cloud幕張リージョン

いるまる宅



# Nekko Cloud浦和リージョン

ちょこざい宅



# Nekko Cloud津田沼リージョン

たけのこ宅

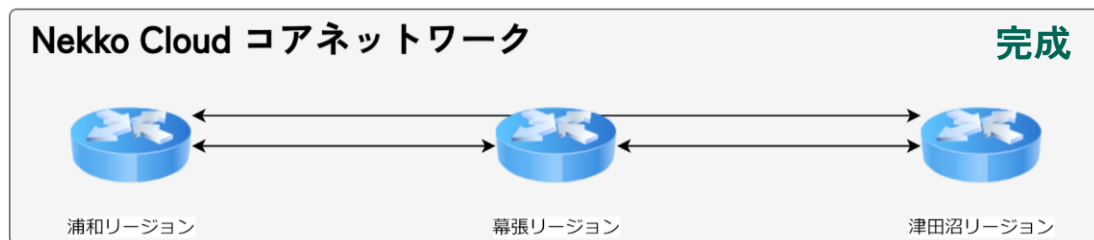
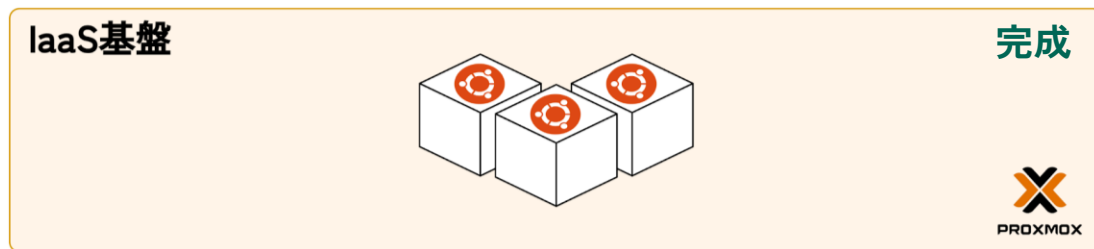
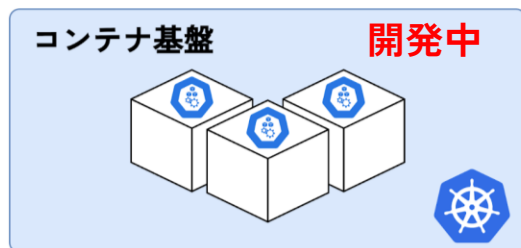




## 3つのリージョンを接続



## Nekko Cloudの基盤構成



NTT NGN網内折り返し

今回はProxmox VEで  
開発した  
IaaS基盤の話をしていきます！



# Nekko Cloudでの Proxmox VEの利用事例

## 要件(こんなシステムが欲しい！)

- サークルメンバーが自由に開発・検証に使える(**dev環境**)
  - Kubernetesの検証, VyOSの検証, 一般的な開発機(VSCoDe RemoteSSH)
    - VMが欲しい
    - **ダッシュボードから簡単にVMを、作って、使って、壊したい**
    - **Proxmox VEよさそう！**

## 要件(こんなシステムが欲しい！)

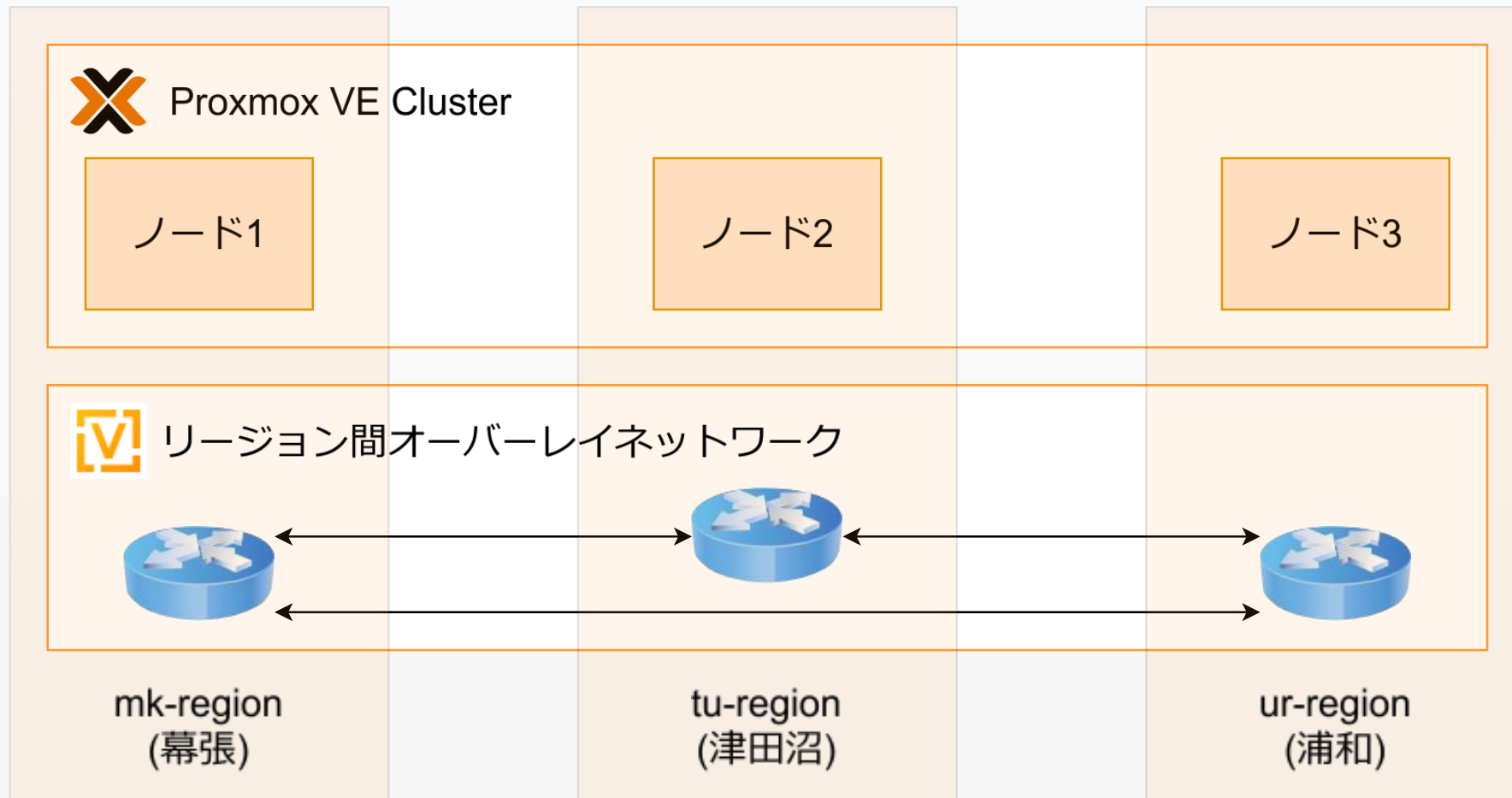
- サークル内で使用するサービスを動かせる(**prod環境**)
    - Growi, Nextcloud, Keycloak, ...
    - **外からのアクセス, 自動デプロイ, ノード障害時の冗長性**
    - **コンテナ化して、VM上のKubernetesで動かしたい**
- **Proxmox VEよさそう！！**

## 要件(こんなシステムが欲しい！)

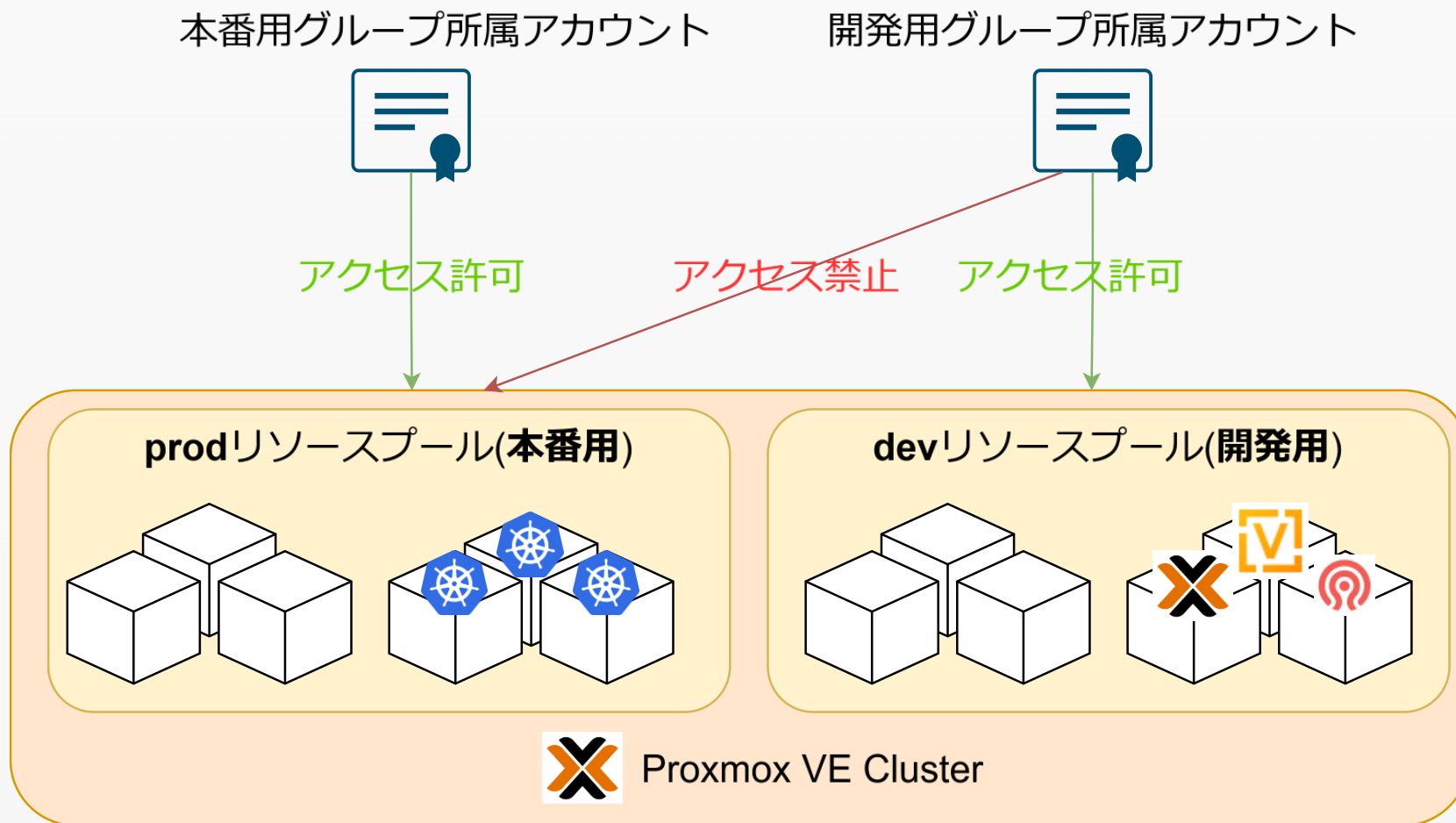
- お金がないので....
  - 複数拠点で、負荷(処理、お金、AZ)をサークルメンバーで分散
  - 各拠点の物理サーバー台数は減らしたい
  - devとprodで権限を分けて1クラスター内で運用 (誤削除事故防止のため)

→ Proxmox VEよさそう！！！！

## 構成/物理



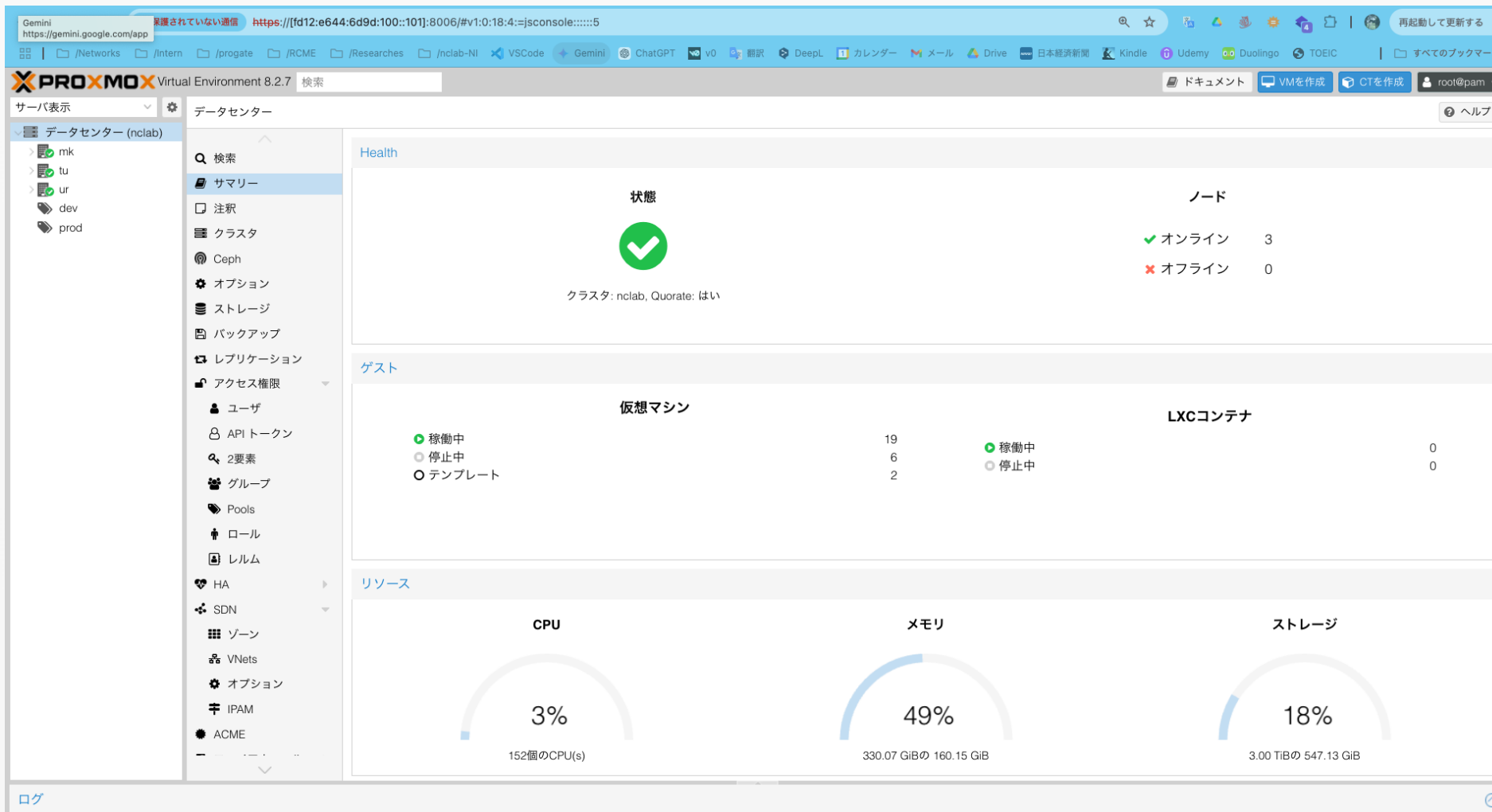
## 構成/論理



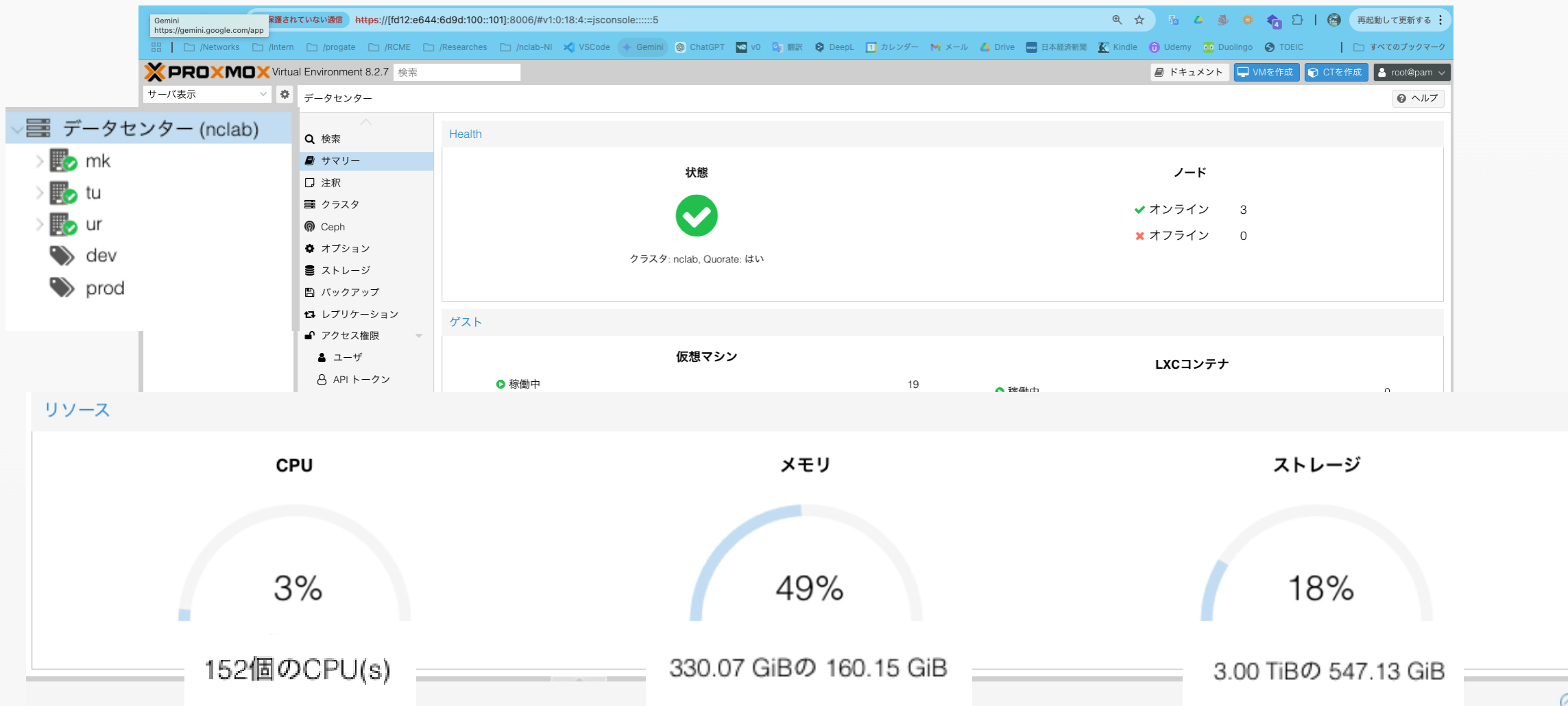
- 誤操作防止用  
リソースプール  
ごとの  
アカウント分離



## 処理能力



## 処理能力





# 最強VM Templateを Ansibleで構築する

## VM設定にかかる工程

工程	作業内容
VM作成	たくさん入力したり、選択したり...
OSインストール	IPアドレス・ユーザー名・SSH鍵・ qemu agentインストール

## VM設定の簡略化

工程	作業内容	解決策
VM作成	たくさん入力したり、選択したり... + Cloud-initが動くようにコマンド実行	<u>VM Templateの使用</u>
OSインストール	IPアドレス・ユーザー名・SSH鍵・ qemu agentインストール	<u>Cloud-Initで自動化</u>

Cloud-init:

Linuxシステムの初回起動時に設定を自動化するためのツール。

インスタンスのユーザーデータを元にネットワーク設定やパッケージインストールなどを行う。

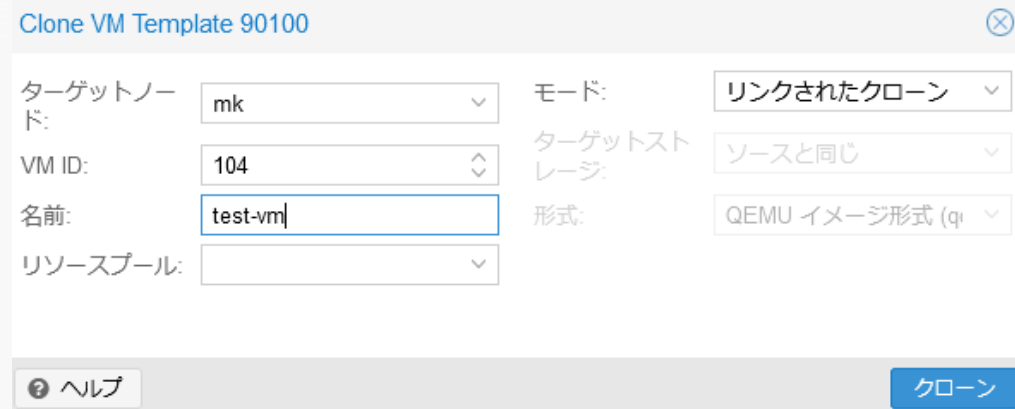
VM Template:

仮想マシンの構成を保存したテンプレート。新しいVM作成時にテンプレートを基にクローンを作成することで、効率的かつ一貫性のあるデプロイが可能。

## VM設定の簡略化

### VMの作り方

1. VM Templateをクローン
2. (オプション)ハードウェアを変更  
メモリ量, CPU数, ...
3. (オプション)Cloud-initを変更  
ssh鍵, ユーザー名, ...
4. VM起動



Clone VM Template 90100

ターゲットノード:	mk	モード:	リンクされたクローン
VM ID:	104	ターゲットストレージ:	ソースと同じ
名前:	test-vm	形式:	QEMU イメージ形式 (q)
リソースプール:			

ヘルプ クローン

## Cloud-init対応VMのVM Templateを作るのもなかなか大変

### 1. ファイルを準備する

(ubuntu-cloud-img.iso, ci-vendor.yaml)

### 2. コマンドを実行する

```
1  qm create {{vmid}} --name ubuntu22.04-template-v1.0.0 --pool {{resource_pool}} --cpu cputype=x86-64-v3 --core 4
   --memory 4096 --balloon 1024 --scsihw virtio-scsi-single --net0 virtio,bridge={{network_bridge}} --ipconfig0 ip=dhcp,
   ip6=dhcp --agent enabled=1,freeze-fs-on-backup=1,fstrim_cloned_disks=0,type=virtio
2
3  ● qm importdisk {{vmid}} /var/lib/vz/template/iso/ubuntu-22.04-server-cloudimg-amd64.img local-lvm
4
5  ● qm set {{vmid}} --scsi0 local-lvm:vm-{{vmid}}-disk-0,ssd=1,discard=on,iotread=1 --ide2 local-lvm:cloudinit --boot
   order=scsi0
6
7  ● qm resize {{vmid}} scsi0 +30516M
8
9  ● qm set {{vmid}} --ciuser ncadmin --cipassword masato1234 --cicustom "vendor=local:snippets/ubuntu22.04-template-v1.0.
   0.ci-vendor.yaml" --ciupgrade 0
10
11 qm template {{vmid}}
```

Cloud-init対応VMのVM Templateを作るのもなかなか大変

手作業でqmコマンドを打つのは

# 面倒くさい

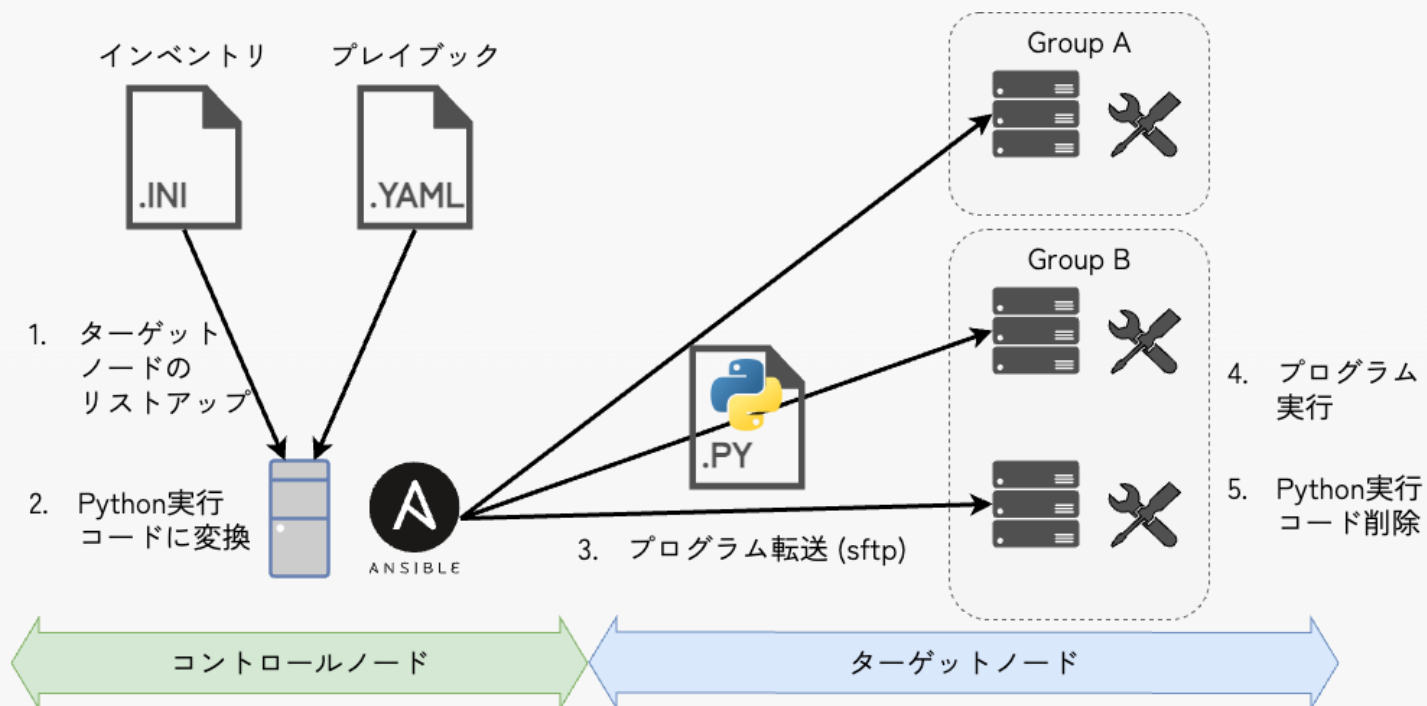
サークル内では同じReadmeを見て作っているはずなのに、  
なぜか毎回違うVM Templateができる...

# Ansible Playbookで解決



## Ansible Playbookって何？

- システムの設定やアプリケーションのデプロイを自動化するための設定ファイル
- YAML形式で記述され、各タスクを順に実行



## Ansible Playbookを使うと何がうれしいか？

- 設定ファイルを元に、サーバーに対して操作を自動で行う
  - → サーバーに対して、何台でも、1かカ月後でも、同じ操作を実行
  - → git管理できる

## Ansible Playbookで複数台のクラスターへVM Templateを1コマンド展開

### 1. 環境構築

```
sudo apt install sshpass ansible -y
```

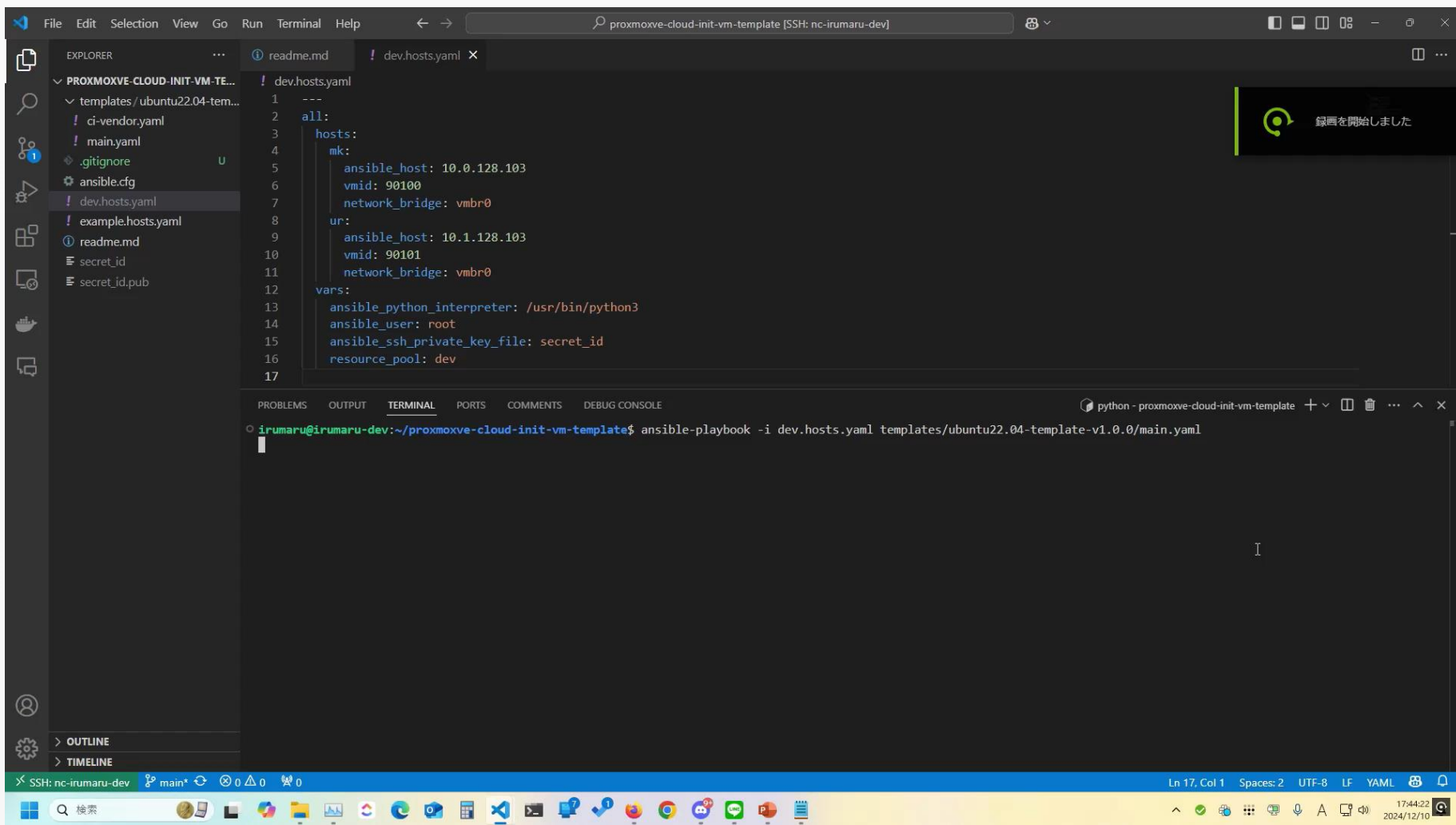
### 2. 複数のサーバーへデプロイ

```
ansible-playbook -i example.hosts.yaml templates/ubuntu22.04-template-v1.0.0/main.yaml
```

環境ごとに異なる値を記述  
例: VM ID, ノードIP, ...

OSごとに異なる値を記述  
Ubuntu22.04, Debian12.8, ...

構築の様子 <https://www.youtube.com/watch?v=w3doTLcGAc0>



The screenshot shows a VS Code editor window with a file explorer on the left and a terminal at the bottom. The file explorer shows a project named 'PROXMOXVE-CLOUD-INIT-VM-TE...' with a subdirectory 'templates/ubunt22.04-tem...'. The terminal window shows the command 'ansible-playbook -i dev.hosts.yaml templates/ubuntu22.04-template-v1.0.0/main.yaml' being executed. The terminal output is currently blank.

```
dev.hosts.yaml
1 ---
2 all:
3   hosts:
4     mk:
5       ansible_host: 10.0.128.103
6       vmid: 90100
7       network_bridge: vmbro
8     ur:
9       ansible_host: 10.1.128.103
10      vmid: 90101
11      network_bridge: vmbro
12 vars:
13   ansible_python_interpreter: /usr/bin/python3
14   ansible_user: root
15   ansible_ssh_private_key_file: secret_id
16   resource_pool: dev
17
```

```
python - proxmoxve-cloud-init-vm-template + v ... ^ x
irumaru@irumaru-dev:~/proxmoxve-cloud-init-vm-template$ ansible-playbook -i dev.hosts.yaml templates/ubuntu22.04-template-v1.0.0/main.yaml
```

※時間の都合上、5分ほどかかる構築を30秒にしています

## Ansible Playbookの構成

### 自動デプロイの内容

1. ci-vendor.yamlをダウンロード
2. ubuntu-cloud-imgをダウンロード
3. VMを作成
4. VMにubuntu-cloud-imgを起動ディスクとしてインポート
5. 起動ディスクに`discard=on`オプションを追加
6. 起動ディスクサイズを拡張 2GiB → 32GiB
7. Cloud-initをセットアップ(ci-vendor.yamlの設定等)
8. VMをテンプレートへ変換

#### 初回のみ実行

- ファイアーウォールの有効化
  - 22/tcpのみ開放
- `qemu-guest-agent`のインストール
- `qemu-guest-agent`の有効化
- `apt upgrade`の実行

# VM Templateの構成/工夫した部分その1

システムディスクのオプション、discard=onについて

```
qm set {{vmid}} --scsi0 local-lvm:vm-{{vmid}}-disk-0,ssd=1,discard=on,iothread=1  
--ide2 local-lvm:cloudinit --boot order=scsi0
```

Ansible Playbookの構成/discard=onオプション ※lvmthin限定の話です

- ゲストOSがTrimコマンドを発行できるようにするオプション
  - ゲストOSでファイルを削除して容量が空いた際に  
ホスト側でも容量を空ける

## Ansible Playbookの構成/discard=onオプション

計測前の処理内容	ディスク使用量の計測	
	オプションなし	discard=onの場合
VMを作成	VM: 5.3%, Node: 6.16GB	VM: 1.7GB, Node: 14.04GB
2.4GBのファイルをダウンロード	VM: 14%, <b>Node: 8.72GB</b>	VM: 4.3GB, <b>Node: 16.8GB</b>
2.4GBのファイルを削除	VM: 6%, <b>Node: 8.72GB</b>	VM: 1.7GB, <b>Node: 14.04GB</b>

変化なし (blue arrow) | 減る (orange arrow)



## Ansible Playbookの構成/discard=onオプション

### lvmthinボリュームを作成する際の挙動

- LVMは通常、ボリュームの作成時にブロックを割り当てる
- LVMのシンプールは、代わりに**書き込み時にブロックを割り当てる**

[https://pve.proxmox.com/wiki/Storage:\\_LVM\\_Thin](https://pve.proxmox.com/wiki/Storage:_LVM_Thin)

### ファイル作成時の挙動

1. ゲストのファイルシステムがファイルを作成する際、ブロックデバイスのブロックを使用する
2. ホストのブロックデバイスのブロックが使用される

## Ansible Playbookの構成/discard=onオプション

### discard=onオプションなしの場合ファイル削除時の挙動

1. ゲストのファイルシステムがファイルを削除して、ブロックを未使用としてマークする
2. ホスト側はなにもしない → **ゲストでファイルを削除しても、ホストの空き容量は増えない**

### discard=onの場合のファイル削除時の挙動

1. ゲストのファイルシステムがファイルを削除して、ブロックを未使用としてマークする
2. **コントローラーがこの情報をストレージに伝える**
3. ストレージはそれに応じてディスクイメージを縮小する  
→ ゲストでファイルを削除した際、ホストの空き容量を増やせる

[https://pve.proxmox.com/pve-docs/pve-admin-guide.html#qm\\_hard\\_disk\\_discard](https://pve.proxmox.com/pve-docs/pve-admin-guide.html#qm_hard_disk_discard)

<https://dustymabe.com/2013/06/11/recover-space-from-vm-disk-images-by-using-discardfstrim/><sup>34</sup>

Ansible Playbookの構成/discard=onオプション

Nekko Cloudでは、discard=onオプションを有効にしています

理由

- SSDを効率的に使いたい
- SSDの寿命を延ばしたい
- テストの結果、データが消える、パフォーマンスが落ちるといった問題が生じない

## Ansible Playbookの構成/discard=onオプション

### 有効にする条件

- discard=onオプションをつける
- ストレージがシンプロビジョニングをサポートしている
- Linux5.0以上のバージョン

(一部のOSではSSDエミュレーションを有効化する必要がある)

[https://pve.proxmox.com/pve-docs/pve-admin-guide.html#qm\\_hard\\_disk\\_discard](https://pve.proxmox.com/pve-docs/pve-admin-guide.html#qm_hard_disk_discard)

# VM Templateの構成/工夫した部分その2

qemu-guest-agentについて

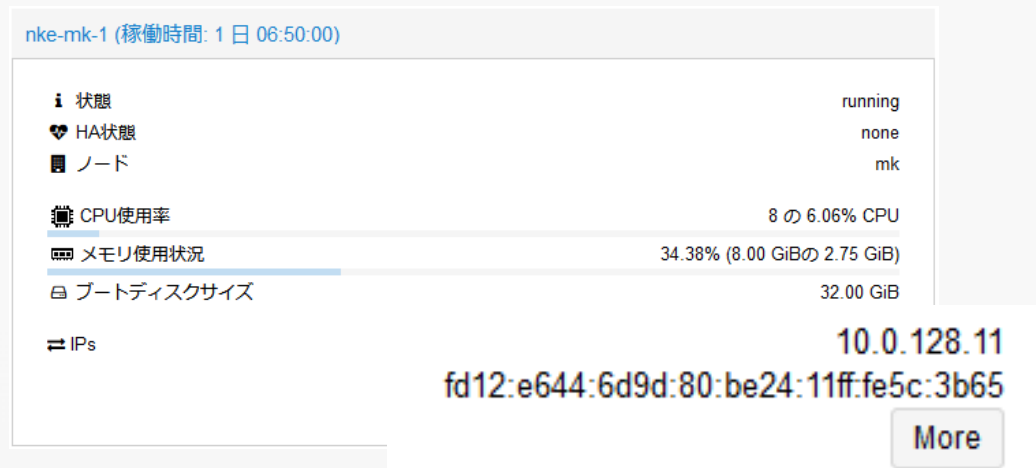
ci-vendorにて実行

```
apt install -y qemu-guest-agent  
systemctl enable qemu-guest-agent
```

## qemu-guest-agentのインストールと有効化

qemu-guest-agentを使用する理由

- VMのIPアドレスをProxmox VEのダッシュボードから確認したい



nke-mk-1 (稼働時間: 1日 06:50:00)

i 状態	running
♥ HA状態	none
🏠 ノード	mk
📊 CPU使用率	8の6.06% CPU
📄 メモリ使用状況	34.38% (8.00 GiBの2.75 GiB)
🗄️ ブートディスクサイズ	32.00 GiB
📄 IPs	10.0.128.11 fd12:e644:6d9d:80:be24:11ff:fe5c:3b65

More

- 今のところ、有効にしたことによる問題は起きていない

<https://pve.proxmox.com/wiki/Qemu-guest-agent>

# VM Templateの構成/工夫した部分その3

apt upgradeについて(ci-vendorにて実行)

## apt upgradeの実行

Q. ci-configでもapt upgradeできるのに  
なぜci-vendorのshellでapt upgradeするのか？

A. 初回起動時のみ自動でapt upgradeしたい

- ci-configは起動時に毎回実行される
- apt upgradeをセットアップ後のサーバーに対して自動で行うと  
予期せぬことが起こりそう...という恐怖 🤖



VM Template構築用  
Ansible Playbookを  
オープンソースで公開しました

VM Template構築用Ansible Playbookをオープンソースで公開しました



公開する目的

- より良いVM Template構築ツールを作りたい！
- コミュニティへの貢献

懇親会での質問, Issue, Pull request

お待ちしております！！！！🙏

(主要メンバーの多数が卒業研究が修羅場のため、返答が遅くなったらごめんなさい🙏)

<https://github.com/nekko-lab/proxmoxve-cloud-init-vm-template>



The background is a solid teal color with two diagonal stripes. One stripe is a lighter shade of teal and runs from the top-left towards the center. The other stripe is a darker shade of teal and runs from the bottom-right towards the center.

**Thank you for watching!**