

NetBoxを利用した IPアドレスの自動払い出し

株式会社エーピーコミュニケーションズ
宮下 悠生

自己紹介

所属

株式会社エーピーコミュニケーションズ

名前

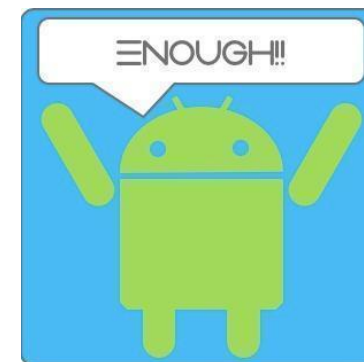
宮下 悠生(みやした ゆうき)

略歴

Slerとしてインフラ全般の設計・構築を数年
2021/11にエーピーコミュニケーションズに入社
ネットワーク・サーバの自動化案件に参画

趣味

ドライブ、釣り、サウナ、自宅自動化



本日の発表について

- NetBoxのIPAM機能を利用してVM構築時のIP払い出しを自動化
- Software Design2023年1月号の「現場を支えるPlaybook」に寄稿させていただいた内容を簡易説明
- Software Designのバックナンバーにあるので興味あれば是非！
<https://gihyo.jp/magazine/SD/archive/2023/202301>

目次

- 業務内容と問題点
- 実装
- 導入効果と新たな課題

業務内容と問題点

前提となる業務

- vSphere仮想基盤環境のLinux VM構築をAnsibleを使って自動化
- 申請を別途サービスポータルで受け持って一部内容を設計
- VM構築をAAPのワークフローで実行
- 月に20～30件程度の申請



VM構築ワークフロー

↓今日話すところはここ

VM作成依頼

VM設計

インスタンス
作成

OS
インストール

VM設計内容

- 利用OS
- ホスト名
- 利用サブネット(IP自動採番)
- 利用OS、MW
- etc...

OS基本設定

MW
インストール

VM引き渡し

サブネットからIPアドレスを割り当てする方法

動的割り当て

例) DHCP

DHCPのみでの実装は簡単だが
別VMにIPアドレスが流用される可能性がある

静的割り当て

例) IPアドレス管理簿、IPAM

IPアドレスを別途管理する必要があるが
IPアドレスでVMを特定可能

⇒ACL設定などでIPアドレスとVMを紐づける必要があるため
静的割り当てが必須

IPアドレス管理簿を使う上での問題点

- 自動化できないため構築対応者による手作業
⇒VM構築のリードタイムに影響
⇒オペレーションミスによる作業品質低下
- Excel自体の管理が必要
⇒ファイル変更の衝突が起きる可能性
⇒VMとの紐づけ・棚卸が困難



IPAM製品の選定

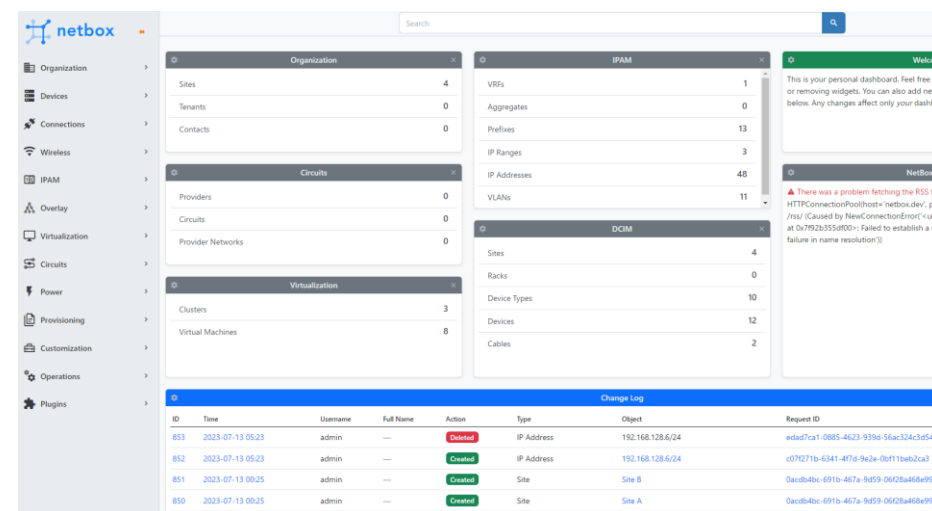
- IPAM(IP Address Management)とは
以下のようなNW要素を管理できるソフトウェア製品のこと
 - IPアドレス
 - プレフィックス(サブネット)
 - VLAN
 - VRF
 - etc...
- Ansibleと連携しやすく検証段階から使いやすい製品



<https://docs.ansible.com/ansible/latest/collections/netbox/netbox/index.html>

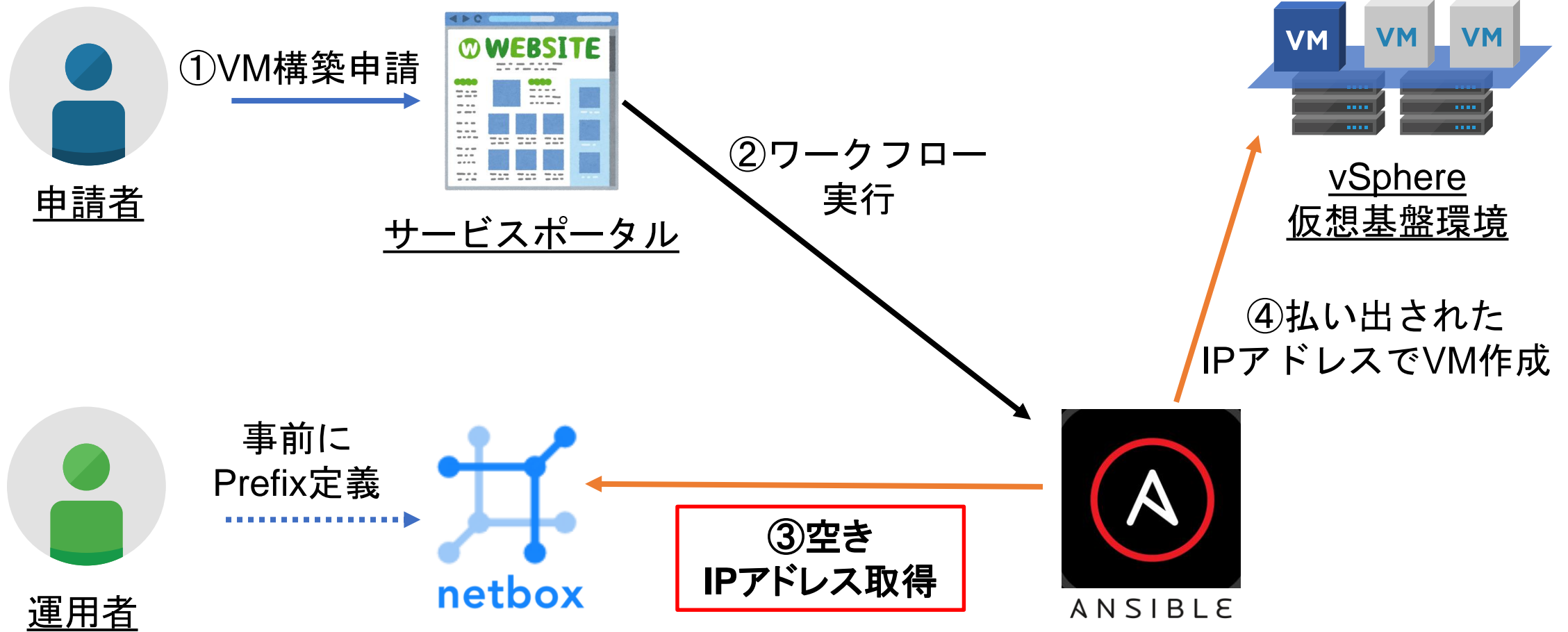
NetBoxとは

- IPAM/DCIMの機能を持つDjangoベースのOSS
- 標準でAPI対応しており、Ansibleとの連携に強い
- プラグイン・カスタムフィールド等で拡張可能
- Docker版もあり、すぐに検証利用可能
- 最新バージョンはv3.5.6(2023/7/25時点)



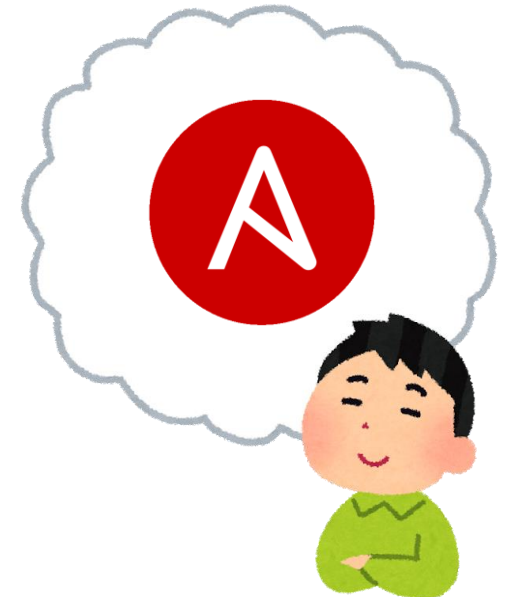
実装

VM構築自動化の全体アーキテクチャ



IPアドレス採番での検討ポイント

- IPアドレスが重複していないかをpingを送信して調査する
- IPアドレスと一緒にVM情報もNetBoxに保管する
- IP採番部分とVM構築部分のPlaybook(ジョブテンプレート)を分離



Playbookの流れ

1. プレフィックスを指定して空きIPアドレスの若番を仮取得する
2. 自動採番されたアドレスが本当に使われていないかを確認する
3. 採番したIPアドレス情報をNetBoxに保管
4. 採番したIPアドレスを次のジョブテンプレートでも利用する

Playbookに渡すパラメータ

```
---  
__hostname: "TESTSERVER" # Netboxに登録するホスト名  
__cluster: "test-cluster" # ホストに登録する仮想基盤クラスタ  
__nic: "eth0" # IPアドレスを紐づけるNIC名  
__prefix: "192.168.0.0/24" # IPアドレスを払い出し元プレフィックス名
```


1. プレフィックスを指定して空きIPアドレスの若番を仮取得する

name: ユーザから指定されたプレフィックス名からIDを取得

set_fact:

__prefix_id: >-

{{

query(

 'netbox.netbox.nb_lookup',

 'prefixes',

 api_filter='prefix=' + __prefix,

 api_endpoint=netbox_url,

 token=netbox_token,

 validate_certs=netbox_validate_certs

)[0].value.id

}}

- name: 次の空IPアドレスを取得

uri:

url: "{{ netbox_url }}/api/ipam/prefixes/{{ __prefix_id }}/available-ips/?limit=1"

method: GET

headers:

 Authorization: "Token {{ netbox_token }}"

url: "{{ netbox_validate_certs }}"

register: __next_available_address

- name: 仮IPアドレスを変数定義

set_fact:

__assigned_address: "{{ __next_available_address.json[0].address | ansible.utils.ipaddr('address') }}"

2. 自動採番されたアドレスが本当に使われていないかを確認する

```
- name: アドレスが使用中かチェック
  command: "ping -c 1 {{ __assigned_address }}"
  changed_when: false
  failed_when: __ping_command.rc not in [0, 1]
  register: __ping_command

- name: アドレスが使用中の場合、失敗
  fail:
    msg: "{{ __assigned_address }} is already used."
  when: __ping_command.rc == 0
```

3. 採番したIPアドレス情報をNetBoxに保管(1/2)

```
- name: VMオブジェクト作成
  netbox.netbox.netbox_virtual_machine:
    netbox_url: "{{ netbox_url }}"
    netbox_token: "{{ netbox_token }}"
    validate_certs: "{{ netbox_validate_certs }}"
    state: present
    data:
      name: "{{ __hostname }}"
      cluster: "{{ __cluster }}"

- name: VMインターフェースオブジェクト作成
  netbox.netbox.netbox_vm_interface:
    netbox_url: "{{ netbox_url }}"
    netbox_token: "{{ netbox_token }}"
    validate_certs: "{{ netbox_validate_certs }}"
    state: present
    data:
      name: "{{ __nic }}"
      virtual_machine: "{{ __hostname }}"
      enabled: true
```

3. 採番したIPアドレス情報をNetBoxに保管(2/2)

- name: アドレスオブジェクト作成

```
netbox.netbox.netbox_ip_address:
```

```
netbox_url: "{{ netbox_url }}"
```

```
netbox_token: "{{ netbox_token }}"
```

```
validate_certs: "{{ netbox_validate_certs }}"
```

```
state: present
```

```
data:
```

```
address: "{{ __assigned_address }}/{{ __prefix | ansible.utils.ipaddr('prefix') }}"
```

```
description: "アドレス自動採番"
```

```
assigned_object:
```

```
virtual_machine: "{{ __hostname }}"
```

```
name: "{{ __nic }}"
```

- name: 作成したアドレスをプライマリIPアドレスに指定

```
netbox.netbox.netbox_virtual_machine:
```

```
netbox_url: "{{ netbox_url }}"
```

```
netbox_token: "{{ netbox_token }}"
```

```
validate_certs: "{{ netbox_validate_certs }}"
```

```
state: present
```

```
data:
```

```
name: "{{ __hostname }}"
```

```
primary_ip4:
```

```
address: "{{ __assigned_address }}/{{ __prefix | ansible.utils.ipaddr('prefix') }}"
```

4.採番したIPアドレスを次のジョブテンプレートでも利用する

```
- name: 自動採番したIPアドレスを次のジョブテンプレートで利用するためにartificatsに保存
  set_stats:
    data:
      __assigned_address: "{{ __assigned_address }}"
```

デモ

- Prefix(192.168.0.0/24)から若番IPアドレスを採番

IP Addresses

Results 4Filters

Quick search

<input type="checkbox"/> IP Address	VRF	Status	Role	Tenant	Assigned	DNS Name	Description
<input type="checkbox"/> 192.168.0.1/24	Global	Active	—	—	✓	—	アドレス自動採番
<input type="checkbox"/> 192.168.0.2/24	Global	Active	—	—	✓	—	アドレス自動採番
<input type="checkbox"/> 192.168.0.3/24	Global	Active	—	—	✓	—	アドレス自動採番
<input type="checkbox"/> 192.168.0.4/24	Global	Active	—	—	✓	—	アドレス自動採番

導入効果と新たな課題

得られた効果

- VM構築にかかる稼働時間、リードタイムの削減
- オペレーションミスがなくなり、作業品質の向上
- VM情報も紐づけて管理することができVMの棚卸が容易



自動化によって生じた課題

- NetBoxの管理コスト増加
⇒運用者への周知や手順書の作成で対応
- 別システムでIPアドレスを管理している箇所もあり
二重管理となってしまう
⇒現状はVM構築に必要なIPアドレス帯域のみをNetBox管理
⇒将来的にはどちらかに寄せる必要がある



まとめ

- NetBoxをIPAMとして使い
VM構築自動化時のIPアドレス払い出しを自動化
- IPアドレス払い出しを自動化することで以下達成
 - VM構築のリードタイム短縮
 - 作業品質向上

さいごに

- 昨今、クラウドサービスやAPIを連携することが前提となっているがそれらをつなぐ手段が用意されていない
- Ansibleをソリューション連携する中心として利用することで無理なく自動化できると考えている

ご清聴ありがとうございました