

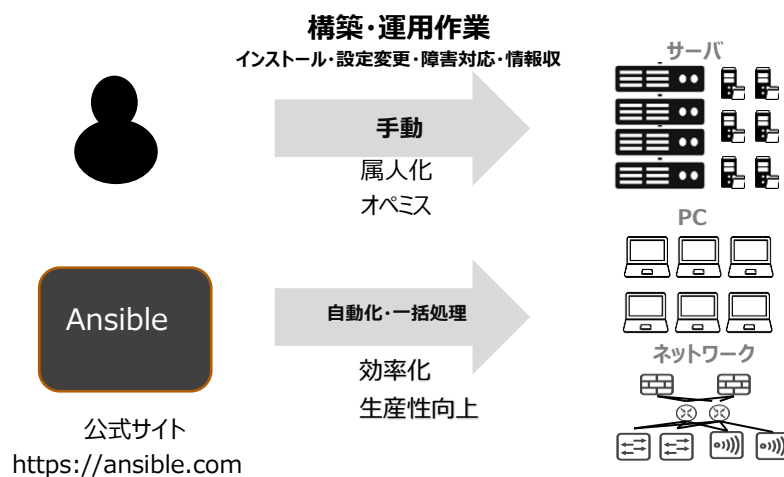
Ansible チュートリアル

船生 英治

アドバンスクラウドエンジニアリング事業部

はじめに

Ansible は、オープンソースの IT 構成管理、オートメーションツールです。本ツールを利用して、大規模環境におけるインフラ構築・設定を自動化することができ、生産性を大幅に向上させることができます。



構成管理というと、IT 機器・ソフトウェア・ライセンスといった資産の情報を採集・整理するデータベースが想起されますが、Ansible においては、大規模環境におけるサーバ、ネットワーク機器に対する、インストール、プロビジョニング、設定変更を効率よく行う「ソフトウェア導入・設定ツール」と理解するとイメージしやすいかと思います。

本書では、Ansible のアーキテクチャを説明し、CUI によるファーストステップとして、

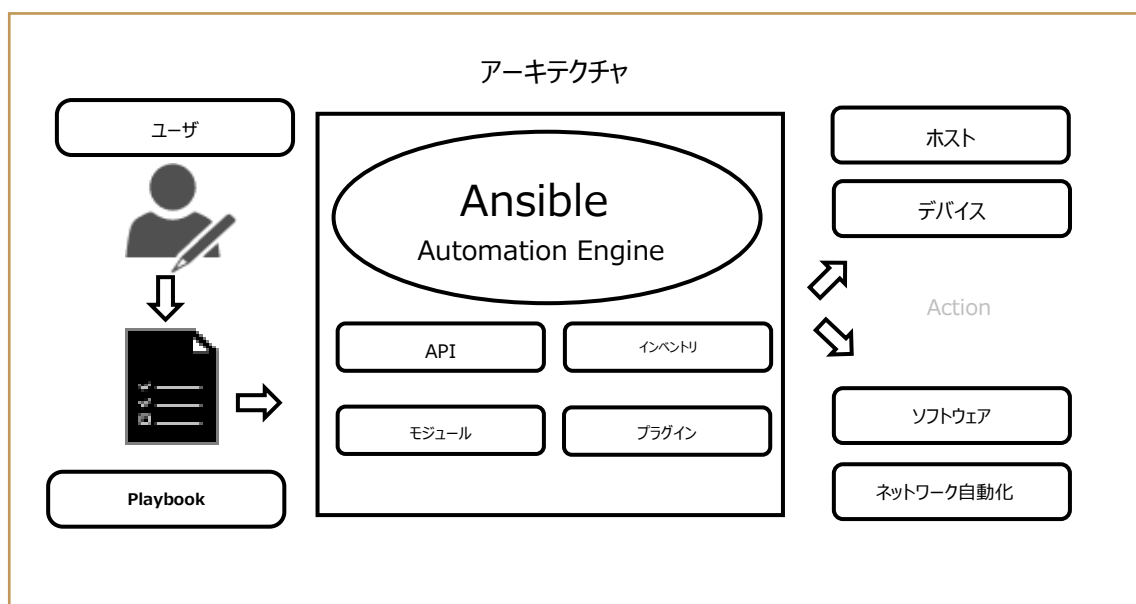
“Hello World” と書かれたテキストファイルをクライアントにコピーする方法を説明します。

同種のオープンソースツールは、20 種類くらいありますが、それらを比較した一覧表¹から見出される Ansible の特徴は以下の通りです。

- 特徴
 - ✓ GUI - 画面操作によるインタラクティブな設定も可能²。
 - ✓ エージェントレス。サーバ管理・監視ツールを利用する際、専用のエージェント(サービス)を必要としない。Ssh/WinRM を利用 (リモートシェル)
 - ✓ マルチプラットフォーム対応。AIX、HP-UX といった、運用管理が必要とされる比較的大規模な環境ではしばし利用されるプラットフォームがサポートされている。

アーキテクチャ

Ansible のアーキテクチャは、非常にシンプルです。イメージは下図の通りとなります。



ユーザは、Playbook (手順の記載) を介して Automation Engine を実行します。Automation Engine は、Playbook 内に定義された内容にしたがって、各種デバイス設定・ネットワーク(設定)を行います。

¹ WIKIPEDIA "[Comparison of open-source configuration management software](#)"

² Qiita [構成管理 Ansible の GUI ツール「AWX」を触ってみた](#)

Automation Engine の構成要素は以下の通りです。

- i. インベントリ：管理する必要のあるホスト（ノード）とその IP アドレス、サーバのリストです。
- ii. API：クラウド・オンプレミス(DC)など IT フィールドとの連携に使用するインターフェイスです。本 API を用いてオーケストレーションを実施することが可能です。
- iii. モジュール：パッケージ・サービス・プログラムなど ホスト上で実行されるライブラリです。OS・ソフトウェア・クラウド環境など 各種プラットフォーム上で、サービス、パッケージ、ファイル などのシステムリソースを制御し、また、システムコマンドを実行することができます。
- iv. プラグイン：Automation Engine の機能を強化します。プラグインにより以下の機能を実現が可能です。
 - (1) アクションプラグイン：Automation Engine 上でモジュールを呼び出す
 - (2) キャッシュプラグイン：インベントリのデータ、収集データを効率的に行う
 - (3) コールバックプラグイン：表示またはログ記録の目的で Ansible イベントにフックしたイベントドリブンな操作を実施する

HelloWorld

本項では、Ansible の簡単な使い方を示すために、クライアントホストに、“HelloWorld”と書かれた ファイルをコピーする手順を紹介します。

1. Playbook の作成

テキストエディタを使い Playbook (手順書) を作成します。

```
[foobar@ansible ~]$ # vi HelloWorld.yml
---
- name: HelloWorld example
  hosts: Client.local
  tasks:
    - src: Making Hello world.
      copy:
        content: HelloWorld
        dest: /var/tmp/HelloWorld.txt
```

本 Playbook は、コピーモジュールを使用して、クライアント Client.local に、`/var/tmp/HelloWorld.txt` を作成します。

Playbook は、可読性の高い yaml³ 形式で書かれているため、内容を理解しやすいです。

2. Playbook の実行

次に、前項で作成した Playbook を、**ansible-playbook** コマンドにより実行します。

```
[foobar@ansible ~]$ # ansible-playbook HelloWorld.yml
PLAY [This is a HelloWorld example] *****

TASK [setup] *****
ok: [Client.local]

TASK [Create a file called '/var/tmp/HelloWorld.txt' with the content 'HelloWorld'.]
changed: [Client.local]
```

3. クライアントのファイルを確認

クライアントにログインし、`HelloWorld.txt` が作成されていることを確認します。

```
[foobar@Client.local ~]$ cat /var/tmp/HelloWorld.txt
HelloWorld
```

³ [Red Hat Customer Portal 第 5 章 YAML の概要](#)

✚おわりに

以上、簡単に、Ansible の紹介をしました。筆者が従事する運用管理プロジェクトでも、数百台のサーバ、ネットワーク機器に対し、本ツールを利用しています。

本書は Ansible 「入門編」 のため、本格的な実例を取り上げることはできませんでした。

今後の記事では、事例を用いて Ansible の特徴を具体的に見ていきたいと思えます。

引用・参考文献

- 1 WIKIPEDIA "Comparison of open-source configuration management software"
https://en.wikipedia.org/wiki/Comparison_of_open-source_configuration_management_software
- 2 Qiita 構成管理 Ansible の GUI ツール「AWX」を試してみた
<https://qiita.com/bindingpry/items/960149b0a1eabba6d6ff>
Red Hat Customer Portal 第 5 章 YAML の概要
- 3 <https://access.redhat.com/obsolete-documentation/ja/red-hat-enterprise-linux-atomic-host/7/paged/getting-started-with-containers/chapter-5-yaml-in-a-nutshell>

GSLetterNeo Vol.168

2022年7月20日発行

発行者 株式会社 SRA 技術本部 先端技術研究室

編集者 熊澤努 方学芬

バックナンバー <https://www.sra.co.jp/public/sra/gsletter/>

お問い合わせ gsneo@sra.co.jp



〒171-8513 東京都豊島区南池袋 2-32-8

夢を。



夢を。Yawaraka Innovation
やわらかいのべーしょん