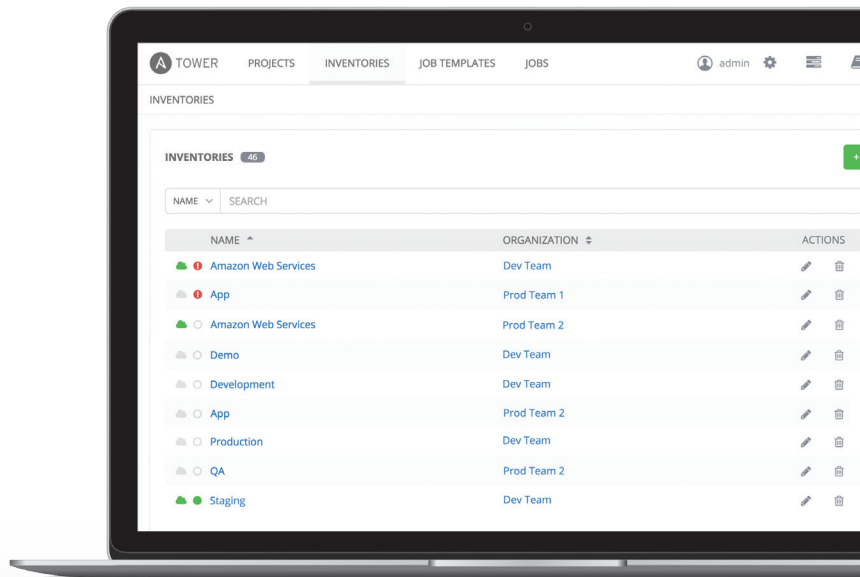


初心者向けガイド

RED HAT ANSIBLE TOWER による 管理/制御



目次

管理/制御を維持するという
課題 2

より優れたAnsible の
実行方法 3

大企業における Ansible Tower と
統合アプローチ 4

Ansible Tower で
インフラストラクチャを
管理する 3 つの方法 4-8

Ansible Tower による管理/制御の
ベストプラクティス 9

まとめ 10

はじめに

本書では、IT 組織が企業規模の自動化を促進し、Ansible 環境に新しいレベルの制御、セキュリティ、および権限委任の機能を導入するための方法を説明します。また、製品の説明にとどまらず、これらの機能が現場でどのような意味を持つのかについても説明します。最後に、当社チームでの実例やヒントを交えつつ、組織で稼働中の Ansible でミッションコントロールのアプローチを導入することがいかに簡単かを示します。

IT はイノベーションの原動力です。ソフトウェアを迅速に提供できなければ、組織は職務を果たせません。イノベーションの実現を阻む最大の障壁は、複雑性です。この課題を克服するため、多くの組織が自動化と DevOps 的なツールやプラクティスの導入を検討しています。従来、DevOps やその他のアジャイル手法の導入を実現するには、独自のプログラミングスキルが必要でした。

しかし、オープンソースの Ansible 自動化エンジンを使用することで、組織はより迅速にアプリケーションを導入し、システムをより効率的に管理し、複雑性を解消できるようになりました。また、それにより DevOps の強力な基盤を構築し、さまざまな部署において「自動化」を身近なものとしています。

Ansible のユーザーは、このことを既に理解しています。なぜなら、Ansible のシンプルかつパワフルでエージェントレスな（そして習得も容易な）自動化機能を、日々実感しているためです。ソフトウェアのデプロイは、もう面倒な作業ではありません。なぜなら、人間にとっては苦痛な反復作業を、Ansible が肩代わりしてくれるからです。Ansible を使用するためには、特別なコーディングスキルは必要ありません。このことは、IT 全体を自動化する際の主要な障壁を取り除いてくれます。Ansible は自動化を通じて、いくらあっても足りないことのない、人間が使うことのできる「時間」を提供します。

管理/制御を維持するという課題

すべての人が自動化の恩恵を受けられるとしたら、素晴らしいことです。しかし、それを可能とするための拡張性は、特にチーム単位で自動化を利用する場合、課題を伴います。

たとえば、次のようなシナリオを想定してみましょう。Ansible を使って作業を始めた当初、そのユーザーであるチームはまだ小規模でした。Ansible は完璧に機能し、自動化によって時間を節約できました。しかし、より多くのユーザーが Ansible を採用し始めるにつれ、状況が変わりました。今ではさまざまなユーザーが同時に独自の Playbook を作成したり、組織のインフラストラクチャ全体を構成したりしようとしています。同時に作業を試みるユーザーの数は 5 人から 20 人、あるいはそれ以上かもしれません。決まった人数があるわけではありませんが、複数のユーザーが、統合されていない、または並行的な自動化を実行している場合、複雑な事態に発展することがあります。

小規模なチーム環境において、メンバー全員が Ansible に精通し、インフラストラクチャに対する制御を維持し、Playbook のベストプラクティスを遵守しているような場合、セキュリティと権限委任の管理は難しくありません。しかし、多くの組織では Ansible の使用について、Ansible のコマンドライン・インタフェース (CLI) の限界を超えた、チーム単位でのニーズが存在します。

具体的に組織が必要としているものは、以下のとおりです。

管理/制御

ユーザーやチームに権限を委任したり、特定のプロジェクトやリソースに対するアクセスをロックしたりする機能

スケジューリング

ジョブのスケジュールや繰り返しの設定

可視性

ジョブステータスの更新、Playbook の実行、Ansible 環境で何が実行中で何が実行中でないかなど、管理者が Ansible の最新の状態をいつでもリアルタイムで確認できる機能

インベントリー

複雑なハイブリッド仮想化環境やクラウド環境においても、インベントリー全体を管理して追跡するための優れた方法

システムトラッキング

マシンがコンプライアンスに準拠し、正確に構成されていることの確認

エンタープライズ統合

既存の環境と企業のツールセットに Ansible を統合

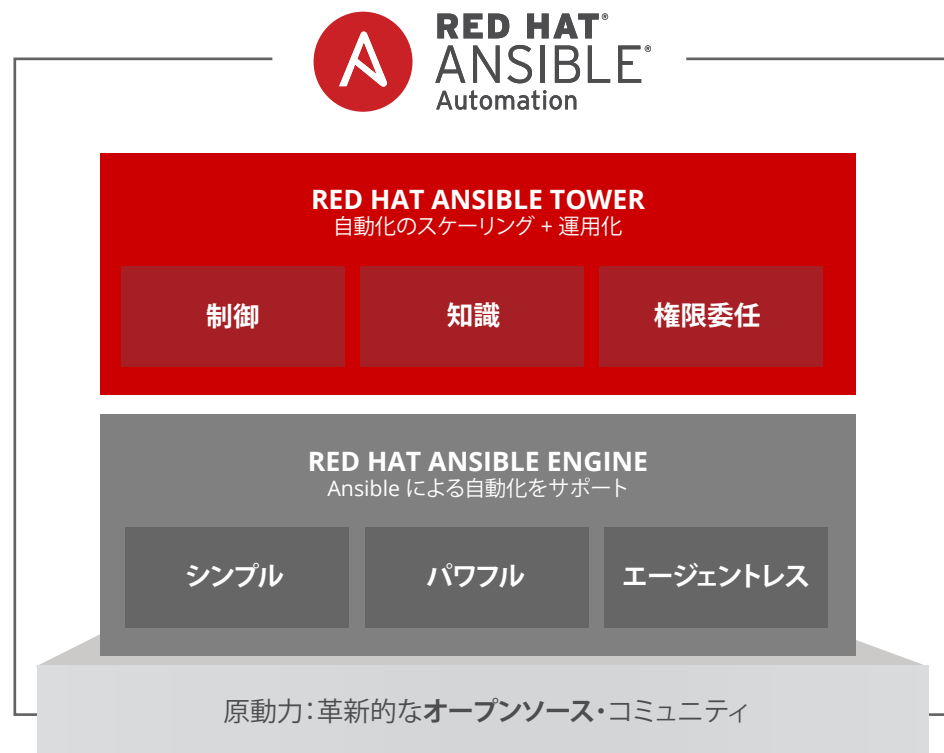
セルフサービス式の IT

時間を解放し、自動化ジョブを他のユーザーに委任できる柔軟性を提供

より優れた **ANSIBLE** の実行方法

Red Hat の Red Hat® Ansible® Tower は、これらの条件の多くを満たしています。Ansible Tower は、UI と RESTful API を使用して Ansible による自動化を制御、保護、および管理するためのエンタープライズ向けフレームワークです。Ansible Tower は、Ansible のシンプルでパワフルな自動化エンジンに加えて制御、知識、および権限委任を実現する、組織内で Ansible を実行するための最善の手段です。

Ansible Tower ではミッションコントロールとして、Ansible のインフラストラクチャをビジュアル・ダッシュボードで一元的に制御します。このダッシュボードは Ansible 環境で起こっていることすべてについて、変更があった際に通知を表示し、ロールベースのアクセス制御、ジョブのスケジューリング、グラフィカルなインベントリ管理機能を提供します。Ansible Tower は Ansible の実行を一元化するため、継続的インテグレーションおよび継続的デリバリー (CI/CD) や DevOps プロセスに必要なシステムやワークフローへ Ansible を統合することを容易にします。



大企業における ANSIBLE TOWER と統合アプローチ

Ansible Tower は Ansible による IT 自動化の取り組みを効率化および一元化するため、企業またはチームレベルで Ansible を使用する際に特に役立ちます。また、これらの Ansible 自動化の取り組みは、権限を委任された組織内のユーザーにより、Ansible の専門知識なしに実行可能です。

Ansible Tower は、従来型の構成管理から、カスタム・アプリケーションのデプロイ、ダウンタイムなしで実現するローリング・アップデートのオーケストレーションに至るまで、さまざまな方法で使用されています。Amelco などの企業では、Ansible をインフラストラクチャのデプロイに一貫して、繰り返し使用しています。NASA はセキュリティ脆弱性のアップデートや nasa.gov の修正と管理のために、毎週 Ansible Tower を使用しています。Web 経由でアプリケーションを配信して収入を得ている企業は、Ansible Tower が IT のボトルネックを解消し、反復的なタスクを自動化し、アプリケーションの市場投入を加速させる上で役立つことに気づいています。

ANSIBLE TOWER でインフラストラクチャを管理する 3 つの方法

オープンソースのパワフルな Ansible 自動化エンジンは、業務に欠かせないものとなりつつあります。そこに組み合わせることで、Ansible Tower は自動化をさらに容易にする数多くの強力な自動化ツールを提供し、Ansible にユーティリティを追加し、Ansible 環境の制御をサポートします。しかし、それは現場では具体的にどのような働きをするのでしょうか？

ここでは Ansible と Ansible Tower によりインフラストラクチャを制御する方法について、Playbook による自動実行よりも高度な例をいくつか紹介します。また、ベストプラクティスや、すぐに利用可能なサンプルの Playbook もご確認いただけます。

1. 構成管理の自動化

構成ファイルの管理とデプロイの一元化は、Ansible のユースケースとして一般的なものです。これは、多くのパワーユーザーが最初に Ansible の自動化プラットフォームに関心を示した理由でもあります。実際 Ansible は、現在利用可能な構成管理の手段のうち、最もシンプルなソリューションです。Ansible は必要最小限の簡素さや一貫性、セキュリティ、および優れた信頼性重視するとともに、管理者や開発者、IT マネージャーによる習得が容易であるように設計されています。

たとえば、Ansible の構成はインフラストラクチャに関する単純なデータの記述 (人間が理解できるものと機械で解析可能なものの両方) でできしており、チーム全員が各構成タスクの意味を理解できるようになっています。このため、新しいチームメンバーも速やかに作業に参加し、実力を発揮できます。既存のチームメンバーは作業をより迅速に完了することができ、構成管理よりも重要で戦略的な作業に取り組むための時間を得られます。

Ansible Tower が提供する機能には、自動構成、プロビジョニング・コールバック、ジョブのスケジューリング、継続的な修復などが含まれ、構成の制御を新たなレベルへと引き上げます。

自動構成

Ansible の Playbook はいつでも任意のマシン上で実行して構成を適用できますが、さらに Ansible Tower を使用することで、環境内で起動したすべてのマシンが適切かつ自動的に構成されるようにすることが可能です。

利用可能なアップデートを適用したい場合でも、自社インフラを定義する構成が詳細な場合でも、Ansible Tower は完全な、自動化された制御を提供します。

ご自身の Ansible 環境で使用できる構成 PlayBook の[例をご覧ください](#)。

プロビジョニング・コールバック

Ansible Tower を使用することで、他のシステム (AWS オートスケーリングや、Kickstart や Preseed などの OS プロビジョニング・システム) によってプロビジョニングされたシステムを自動的に構成したり、Ansible Tower のユーザー・インタフェースを直接使用せずにプログラムでジョブを呼び出すことも可能となります。Ansible Tower のプロビジョニング・コールバック機能を使用すると、ユーザーが Ansible Tower のコンソールを使用してホスト管理のジョブを起動するのを待つことなく、Ansible Tower の Playbook をマシン上で実行するよう、Ansible Tower の RESTful API 経由でトリガーできます。

プロビジョニング・コールバックを設定し、サンプルの EC2 プロビジョニング Playbook を表示するには、[こちらのヒントをご覧ください](#)。

ジョブのスケジューリング

手動スクリプトやその場しのぎのアドホックなプラクティスを卒業し、一貫性、信頼性とセキュリティに優れた方法で環境を管理しましょう。Playbook の実行、クラウド・インベントリへのアップデート、およびソース管理のアップデートは、すべて Tower 内でスケジュール設定が可能です。スケジュールには、1 回限りまたは繰り返しの実施 (たとえばメンテナンス期間中など) を設定できます。管理リクエストの数が多い場合、ビルトインのキューシステムが、ジョブの実行を効率化します。

スケジューリング機能を使用することで、定期的な修復や継続的なデプロイメントを支援できるほか、毎晩の夜間バックアップなども設定が可能です。スケジュールの設定は簡単です。ジョブテンプレートを編集する際に、スケジュール・メニューの下に追加するだけです。または、ジョブテンプレートのリストへ移動してから、スケジュールアイコンと「+」をクリックしても、新しいスケジュールを追加できます。このジョブはスケジュールに従って自動的にアップデートを適用します。設定済みのスケジュールは、必要に応じて一時停止ないし停止することができます。

The screenshot shows the 'SCHEDULES' page in Ansible Tower, specifically the configuration for a schedule named 'Tuesday update window'. The breadcrumb trail at the top is 'JOB TEMPLATES / APPLY UPDATES / SCHEDULES / TUESDAY UPDATE WINDOW'. The configuration form includes the following fields:

- * NAME:** Tuesday update window
- * START DATE:** 10/13/2016
- * START TIME (HH24:MM:SS):** 04:00:00
- * LOCAL TIME ZONE:** America/New_York
- * REPEAT FREQUENCY:** Week
- FREQUENCY DETAILS:**
 - * EVERY:** 1 WEEKS
 - * ON DAYS:** SUN MON **TUE** WED THU FRI SAT
 - * END:** Never

継続的な修復

構成管理の仕事はほとんどの場合、マシンの起動時に構成を適用するだけでは終わりません。その後、必ず何らかの変更が必要となります。システム管理者によるオペレーティング・システムのアップデートや、アプリケーションの変更、ローカルの変更などは、すべて構成のゆらぎにつながります。ここから、「継続的な修復」という概念が生まれます。継続的な修復の機能は、ベースラインからの逸脱を軽減するため、指定の構成を定期的かつ自動的に適用します。この作業は Ansible により効率化され、Ansible Tower のジョブ・スケジューリング機能により簡単になります。

修復の実行を行う間隔は、適宜スケジューリングすることができます。構成の修復の実行後は、実行結果を解釈しましょう。指定の構成を一貫して適用することも重要ですが、構成のリセットが繰り返される場合には、どこかに問題がある可能性もあります。次に取るべきアクションを決定するには、まずどのような変更が加えられたのかを把握する必要があります。Ansible を使用する重要なメリットとして、必要な場合にのみ変更が加えられることが挙げられます。変更の必要がなかった場合、タスクは「OK」とのみ報告されます。このような挙動は、しばしば「理想の構成状態」または「冪等性」と呼ばれます。この特徴に加えて、Ansible Tower の監査機能や、Ansible 実行結果のロギング機能を組み合わせることで、構成のゆらぎを簡単に見つけられるようになります。[詳細はこちら](#)。

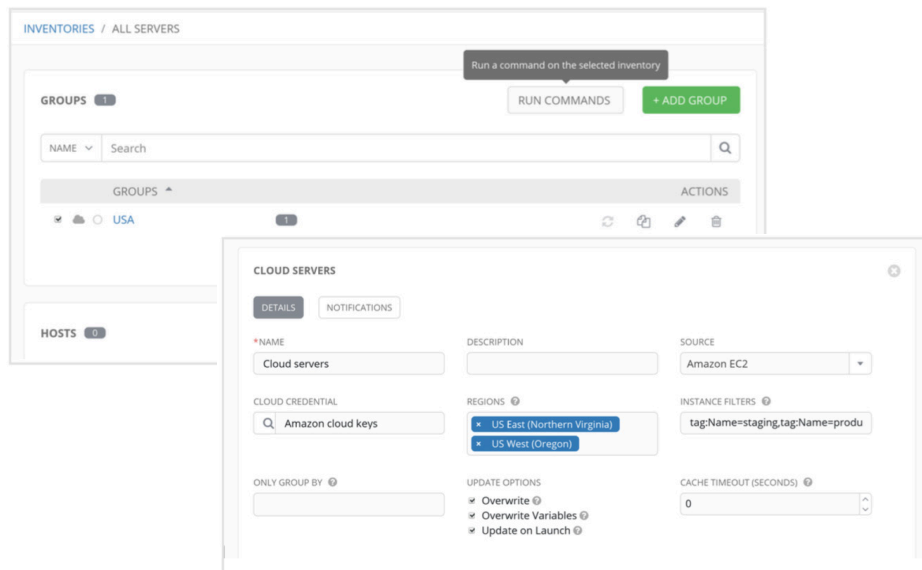
2. インベントリー全体を管理して追跡する

前のセクションでは、Ansible Tower を使用した構成定義や継続的修復により、インフラストラクチャの構成制御が容易になることを説明しました。しかし、インフラストラクチャの構成を制御することは、ほんの一步にすぎません。インベントリーを Ansible Tower 内で適切に制御できるようにするためには、一元化されたソースが必要です。

Ansible Tower が提供する機能を利用することで、利用中のクラウドが AWS、Rackspace、OpenStack、Google Compute Engine、Azure、VMware のいずれであっても、インベントリーの定義と管理を簡単に行うことができます。Ansible Tower はクラウドのインベントリーの同期を助けるだけでなく、強力なプロビジョニング・コールバック機能を提供することにより、ノードがオンデマンドで構成を要求できるようになるため、オートスケーリングが可能になります。

インベントリーの基本

Ansible を使用したことがあるならば、インベントリーの基礎知識もあるはずです。Ansible の静的インベントリーは、ホストとグループを記述する INI 形式のファイルにすぎず、オプションとしてホストおよびグループに変数を適用することができます。[静的インベントリーの例は、こちらで確認できます](#)。以下の画像からも分かるように、同じようなインベントリーを Ansible でも簡単に入力することが可能です。また、Ansible Tower の RESTful API を使用して簡単にインベントリーを入力できる柔軟性もあります。Ansible Tower は複数のインベントリーに対応しており、開発、テスト、およびプロダクション環境用のよく似たインベントリーを複数、簡単に作成することができます。RESTful API を使用してインベントリーを作成する方法については、[こちらの例を参照](#)してください。



動的インベントリー

上記の通り、Ansible Tower はインベントリーの一元的な管理場所として利用することが可能です。しかしほとんどの環境におけるインベントリーは、極めて動的な性質を帯びています。なぜなら、多くのマシンがプロビジョニングおよびリタイアされている中、それらのマシンのグループ、ファクト、変数群は、クラウドのプロバイダー、プロビジョニング・システム、構成管理データベースなど、さまざまなソースから提供されている場合があるためです。

Ansible と Ansible Tower は、「動的インベントリー」という概念を通じてこれらのソースと連携します。たとえば、インベントリースourceとして AWS を使用している場合、AWS ホスト用のグループを作成し、インベントリーは Amazon EC2 をインベントリースourceとして使用するよう構成します。このインベントリーは、地域やイメージタグ、その他の Amazon メタデータを使用して、さまざまな方法でフィルタリングが可能です。このインベントリーグループが作成されると、インベントリーのアップデートは、オンデマンドで行うことも、スケジュールに従って行うことも可能です。また、このインベントリーを参照する Playbook を実行する都度、自動的にインベントリーをアップデートすることも可能です。また、従来通り API を使用して動的インベントリーを設定することもできます。

カスタム動的インベントリー

Ansible Tower には Amazon、Microsoft Azure、OpenStack などの主要なパブリッククラウドおよびプライベートクラウドのプロバイダーを対象としたインベントリースクリプトが付属しているほか、独自の動的インベントリーも簡単に追加できます。Ansible Tower の設定メニューには、カスタム・インベントリースクリプトをアップロードできる「インベントリースクリプト」の項目があります。

高度なカスタム動的インベントリー

さらに、インベントリーを Playbook と一緒にソース管理で保存している場合は、PlayBook と連携して一律にアップデートできます。手動で Ansible Tower に同期させる必要はありません。動的インベントリー、カスタムインベントリー、カスタム動的インベントリーのスクリプトの[例をご覧ください](#)。

3. 日常の管理タスクを簡素化

構成管理と継続的な修復のワークフローを作成し、インベントリーのソースを設定したら、システムの制御が完了したと思うかもしれません。しかしほとんどの場合、日常的なシステム管理にも多くの時間が必要となります。時には、サービスやマシンを再起動したり、一度限りのパッチを適用したりする必要もあります。Ansible Tower を使用すると、これらの日常的なタスクを制御し、簡単に簡素化することができます。

ロールベースのアクセス制御と監査

これは、Ansible 環境において誰が何をするかを管理する機能です。Ansible Tower を使用すると、さまざまなチームや明示的ユーザーに対して、特定の権限を委任できます。また、特定のプロジェクトを非公開にできるほか、一部のユーザーだけにインベントリーの編集権限を与え、その他のユーザーには特定のシステム (確認用のドライランモードまたはライブモードのいずれか) のみで Playbook を実行できるようにもできます。資格情報を開示せず、特定のユーザーが資格情報を使えるようにすることも可能です。Ansible Tower は、オブジェクトの編集やジョブの起動を含め、ユーザーが行うあらゆる操作の履歴と作業者を記録します。

「ジャストインタイム」管理の制御

現場ではしばしば、ユーザー 1 人の追加、セキュリティの脆弱性 1 つに対するアップデート、異常の発生したサービス 1 つの再起動といった、いくつかのホストでの単純な作業が必要とされます。これをサポートするのが、Ansible Tower のリモートコマンド実行機能です。Ansible で個別の play として記述できるタスクはすべて、インベントリー内のホストまたはホストグループ上で実行可能です。このため、システムの起動と管理が迅速かつ容易になります。さらに、Ansible Tower が提供するロールベースのアクセス制御エンジンと詳細な監査ログ機能により、誰がどのマシンに何をしたのかも明確に記録されています。

The screenshot shows the Ansible Tower web interface. At the top, there is a breadcrumb navigation: INVENTORIES / ALL SERVERS / USA / SOUTHEAST. Below this, the 'SOUTHEAST' host configuration page is displayed. It has two tabs: 'DETAILS' (selected) and 'NOTIFICATIONS'. Under 'DETAILS', there are three input fields: 'NAME' (containing 'Southeast'), 'DESCRIPTION' (empty), and 'SOURCE' (a dropdown menu showing 'Manual'). Below these fields, there is a section for 'VARIABLES' with a toggle for 'YAML' (selected) and 'JSON'. The YAML editor shows three lines of code: 1. '---', 2. 'escape_pods: 2', and 3. 'halon system timeout: 30'.

システムトラッキング

Ansible Tower のシステムトラッキングは、インフラストラクチャに新しいレベルの可視性をもたらします。システムの現状を過去の状態やクラスタ内の他のシステムと比較することで、システム上で起こっていることを正確に把握できるようにします。このことは、コンプライアンスへの準拠を確実化する上でも役立ちます。システムトラッキングのデータストアは豊富かつ拡張可能で、Ansible Tower の RESTful API を介してアクセスできるため、他のツールやシステムに提供することができます。

安全性確保の自動化

Ansible を使用すれば、マシンが仕様を満たしていないインスタンスへの対応も容易です。このような場合、誰かが手動で変更を加えたか、ソフトウェアが異常な動作をしている可能性があります。原因が何であれ調査が必要です。Ansible の柔軟性と Ansible Tower のブロックサポートにより、タスクの論理グループ化と、実行中のエラー処理が可能になります。Playbook のスケジューリングを行うだけで、Ansible Tower は仕様から大幅に外れたシステムを自動的にリフレッシュします。加えて、新しいインスタンスがスピンアップされた後には、基本的な設定を適用するため、Ansible Tower へのコールバックも行います。

ANSIBLE TOWER による管理/制御のベストプラクティス

Ansible Tower が Ansible 環境にもたらす制御の機能は、強力であるがゆえに、ある種の責任を伴います。Ansible および Ansible Tower を最大限に活用するためには、いくつかのベストプラクティスに従うことが役立ちます。この点について理解を深めるには、Ansible による自動化の性質を理解することが重要です。

Ansible は以下のものを定義します。

インフラストラクチャ: Playbook で定義

構成: マシンリーダブルなデータやコードで定義

これにより、手動作業でマシンを構成するプロセスを排除し、インフラストラクチャとアプリケーションをプログラムによって表現するビルドプロセスで置き換えます。

Ansible 環境ではコードとしてのインフラストラクチャをマシンの管理に使用するので、ユーザーもインフラストラクチャをコードのように扱う必要があります。このため、採用すべきベストプラクティスもコードと同様のものとなり、構成や Playbook をプロダクション環境にデプロイする前には、適切なテストを行う必要があります。

1. ソースコントロールの使用

Ansible Tower は Ansible Tower サーバーに直接格納されている Playbook を使用することができますが、Playbook、ルール、および関連する詳細はすべて、[ソースコントロール](#)に保存することをお勧めします。そうすることで、インフラストラクチャを自動化するルールを変更した時期と理由を記述した監査証跡を得ることができます。

また、チームやインフラストラクチャ内の別の部分と Playbook を共有することも簡単になります。これにより、たとえば一度解決した問題であれば、解決方法を Ansible で自動化してから Ansible Tower で共有することができます。Ansible Tower では手動で Playbook をアップロードすることも可能ですが、ソースコントロールの使用を強くお勧めします。

2. テスト

コード環境と同じように、構成と Playbook はプロダクションへとプッシュする前に必ずテストしましょう。また、Playbook を本稼働させる前にテストできるよう、開発環境を構築するべきです。デプロイメントのワークフローにある程度のテストを組み込むことで、コードが本稼働した際に発生しうる予期せぬ自体を減らすことができます。また、多くの場合、プロダクション環境でもテストを行うことで、問題のあるアップデートがインストール全体に及んでしまうリスクを軽減できます。

Ansible はプッシュ型であるため、このようなステップをローカルホストやテストサーバー上で実行することも極めて簡単です。Ansible を使用すれば、アップグレードを行う際のワークフローに、チェック項目やテストを必要なだけ組み込むことができます。テスト作業を Ansible Playbook へ統合するための最適な方法については、[こちらで詳細](#)をご確認ください。

まとめ

Red Hat Ansible Tower を使用することで、構成作業の制御や、インベントリ管理、日常的な管理作業のためにオンデマンドで行うリモートコマンドの実行など、大部分のプロセスを自動化することが可能です。Red Hat Ansible Tower は、企業チームが Ansible による IT 自動化の取り組みを集中管理し、簡素化する上での助けとなります。

Ansible Tower は使いやすく、エージェントレスで、Ansible 環境の全体を一覧できるビューを提供します。さらにロールベースのアクセス制御機能と、環境内で起こっているすべてに関する情報が監査に適した形式のログで提供されるため、システムの構成状況をリアルタイムで監視することができます。

また、Ansible Tower は、Ansible の自動化エンジンに強力なセルフサービス機能をもたらします。これにより、Ansible の力を組織全体で活用することが可能となります。たとえば、開発者や QA 部門であれば、独自の開発環境やテスト環境をプロビジョニングすることができます。カスタマーサービス・エージェントならば、新しいデモ環境をプロビジョニングできます。駆け出しの管理者であれば、パスワード変更作業などの簡単なジョブを、ボタン 1 つで実行できるでしょう。複雑さの軽減、反復的なタスクやエラーの排除、生産性の向上、仕事のコラボレーションと満足度向上など、Ansible Tower が提供する数々の利点を通じて、成功を約束する優れた企業文化を実現しましょう。

RED HAT ANSIBLE TOWER について

Ansible は、Red Hat をスポンサーとするオープンソースのコミュニティ・プロジェクトであり、IT 自動化を実現するための最もシンプルな手段であるとともに、システム管理者やネットワーク管理者から開発者および経営陣に至るまで、さまざまな IT チームが使用することのできる唯一の自動化言語です。Red Hat® Ansible® Automation は、アプリケーション・ライフサイクル全体を自動化するためのエンタープライズに対応したソリューションを提供し、サーバーからクラウドやコンテナ、そしてその間に存在するすべてのものを自動化します。Red Hat® Ansible® Tower は、複数の層で構成される複雑なデプロイ作業をサポートする商用製品であり、Ansible 導入環境に管理/制御、知識、そして権限委任の機能を追加します。

RED HAT について

オープンソースソリューションのプロバイダーとして世界をリードする Red Hat は、コミュニティとの協業により高い信頼性と性能を備えるクラウド、Linux、ミドルウェア、ストレージおよび仮想化テクノロジーを提供、さらにサポート、トレーニング、コンサルティングサービスも提供しています。Red Hat は、お客様とパートナー、そしてオープンソース・コミュニティのグローバル・ネットワークをつなぐ役割を担い、成長のためのリソースの解放と、お客様が将来の IT を実現するための革新的な技術の構築を支援します。

北米	ヨーロッパ、中東、 アフリカ	アジア太平洋地域	中南米
1 888 REDHAT1	00800 7334 2835	+65 6490 4200	+54 11 4329 7300
www.redhat.com	europe@redhat.com	apac@redhat.com	info-latam@redhat.com

日本
03 5798 8510



Copyright ©2017 Red Hat, Inc. Red Hat, Shadowman ロゴ、および Ansible は米国およびその他の国における Red Hat, Inc. またはその関連子会社の商標または登録商標です。