Red Hat Ansible Automation Platform 2.4

Ansible Playbook のスタートガイド

Ansible Playbook の使用

Last Updated: 2024-07-31

Red Hat Ansible Automation Platform 2.4 Ansible Playbook のスタートガイド

Ansible Playbook の使用

法律上の通知

概要

このガイドでは、自動化の要件に対応するために Playbook を作成して使用する方法を説明します。 このドキュメントには、GNU GENERAL PUBLIC LICENSE v3.0 の対象であるアップストリームの docs.ansible.com ドキュメントのコンテンツが含まれています。

目次

はじめに

Red Hat Ansible Automation Platform に興味をお持ちいただきありがとうございます。Ansible Automation Platform は、Ansible を装備した環境に、制御、ナレッジ、委譲の機能を追加して、チームが複雑かつ複数層のデプロイメントを管理できるように支援する商用サービスです。

このガイドでは、Ansible Playbook の使用方法について説明します。

RED HAT ドキュメントへのフィードバック (英語のみ)

このドキュメントを改善するための提案がある場合、またはエラーを見つけた場合は、テクニカルサポート (https://access.redhat.com) に連絡し、**docs-product** コンポーネントを使用して Ansible Automation Platform Jira プロジェクトで Issue を作成してください。

第1章 概要

Ansible Playbook は自動化タスクの青写真で、自動化タスクとは、ソリューションのインベントリー全体で、手作業を抑えて実行するアクションのことです。Playbook は Ansible に、どのデバイスで実行するかを指示します。IT 環境全体で何百、何千もの同様のテクノロジーに同じアクションを手動で適用する代わりに、Playbook を実行すると、ルーターの設定など、指定された種類のインベントリーに対して同じアクションが自動的に完了されます。

Playbook は、オペレーティングシステムや Kubernetes プラットフォーム、ネットワーク、セキュリティーシステム、GitHub などのコードリポジトリーなどの IT インフラストラクチャーを自動化するために定期的に使用されます。Playbook を使用すると、最初からすべて作成する手動のオーバーヘッドなしで、アプリケーション、サービス、サーバーノード、その他のデバイスをプログラムできます。Playbook とその中の条件、変数、タスクは、無期限に保存、共有、再利用できます。こうすることで、運用上の知識を体系化し、同じアクションを一貫して実行することが容易になります。

1.1. ANSIBLE PLAYBOOK の仕組み

Ansible Playbook は、指定したインベントリーまたはホストのグループに自動的に実行されるタスクの一覧です。1つ以上の Ansible タスクを組み合わせてプレイ、つまり、特定のホストにマッピングされたタスクを順序付けてグループ化できます。

タスクは、書き込まれた順序で実行されます。

Playbook には1つ以上のプレイを含めることができます。

Playbook は、1つ以上の **プレイ** を順序付けしたリストまとめ、構成されます。

Playbook と **プレイ** という用語はスポーツに例えられます。

各プレイは、Playbook の全体的な目標の一部を実行し、1つ以上のタスクを実行します。

各タスクは Ansible モジュールを呼び出します。

Playbook

全体的な目標を達成するために、Ansible が上から下に操作を実行する順序を定義するプレイのリスト。

プレイ

インベントリーのマネージドノードにマップするタスクの順序付きリスト。

タスク

Ansible が実行する操作を定義する単一モジュールへの参照。

ロール

ロールは、必要に応じて任意の Playbook で使用できる「ライブラリー」に機能を組み込むことで、Playbook 内のコードを再利用可能にする方法です。

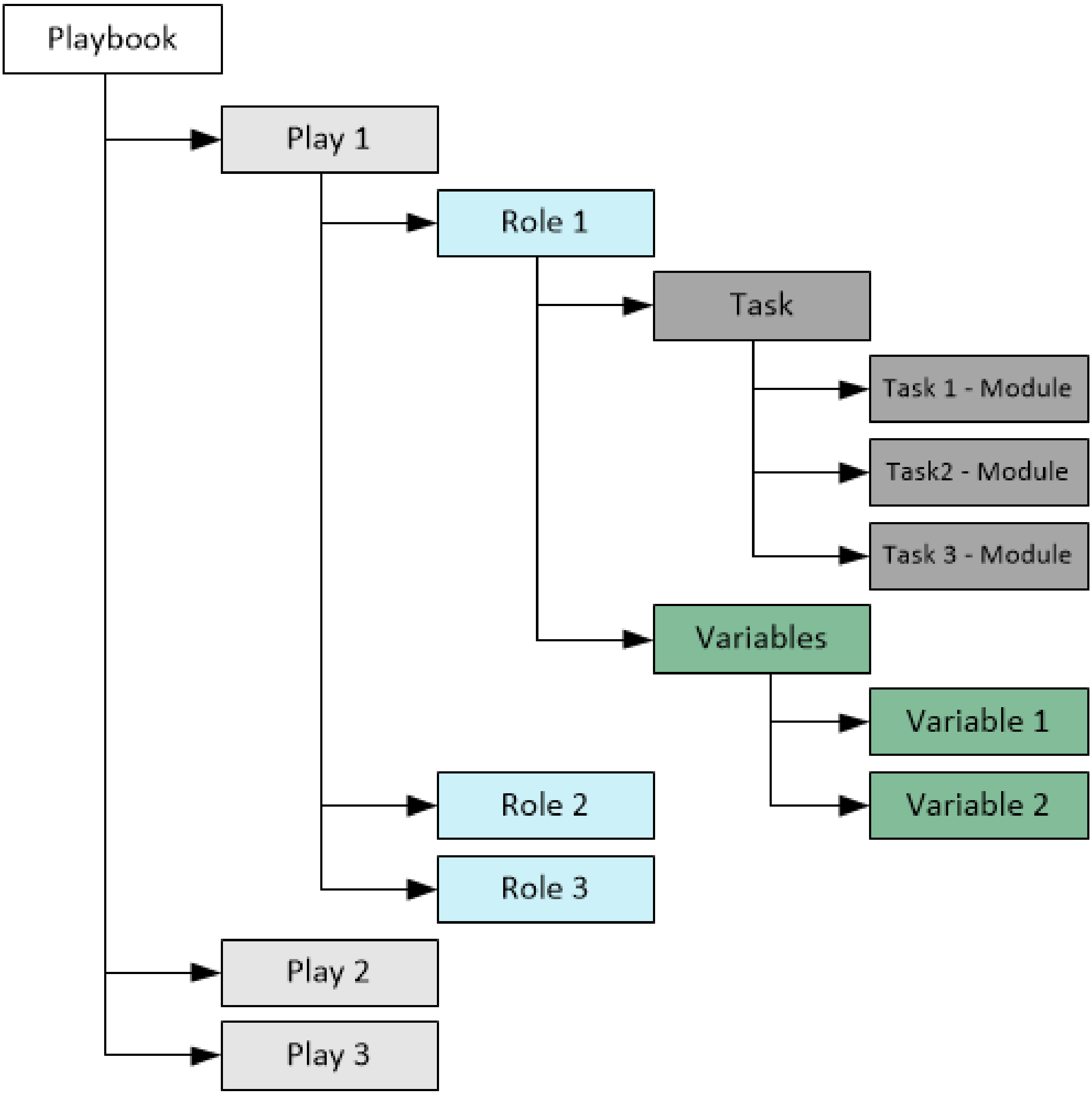
モジュール

Ansible が管理ノードで実行するコードまたはバイナリーの単位。

Ansible モジュールは、各モジュールの **完全修飾コレクション名** (FQCN) を持つコレクションにグループ化されます。タスクはモジュールによって実行され、それぞれが Playbook 内の特定のタスクを実行します。モジュールには、タスクが実行される時期と場所、およびタスクを実行するユーザーを決定するメタデータが含まれています。次のようなあらゆる種類の IT タスクを実行する Ansible モジュールが何千もあります。

- クラウド管理

- ユーザー管理
- ネットワーク
- セキュリティー
- 設定管理
- 通信



1.2. ANSIBLE PLAYBOOK の使用方法

Ansible は YAML 構文を使用します。YAML は、人間が読める言語で、複雑なコーディング言語を学ばなくても Playbook を作成できます。

YAML の詳細は、YAML 構文を参照し、テキストエディターのアドオンのインストールを検討してください。Playbook で明確な YAML 構文を記述するのに役立つ その他のツールとプログラム を参照してください。

Ansible Playbook を使用する方法は 2 つあります。

- **コマンドラインインターフェイス (CLI) の使用**
- Red Hat Ansible Automation Platform のプッシュボタンデプロイメントの使用

1.2.1. CLI からの操作

オープンソースの Ansible プロジェクトまたは Red Hat Ansible Automation Platform のインストール後に、Red Hat Enterprise Linux CLI で以下を使用して、

```
$ sudo dnf install ansible
```

ansible-playbook コマンドで Ansible Playbook を実行できます。

1.2.2. プラットフォーム内からの操作

Red Hat Ansible Automation Platform ユーザーインターフェイスは、大規模なジョブまたはジョブテンプレートの一部として使用できる、プッシュボタンの Ansible Playbook デプロイメントを提供します。これらのデプロイメントには、IT 自動化に慣れていない場合や、CLI での作業経験があまりない場合に、特に役立つ追加の保護手段が含まれています。

1.3. ANSIBLE を使用した自動化の開始

自動化プロジェクトを作成し、インベントリーを構築し、**Hello World** Playbook を作成して、Ansible の使用を開始します。

前提条件

- Ansible パッケージがインストールされている。

手順

- ファイルシステムにプロジェクトフォルダーを作成します。

```
mkdir ansible_quickstart
cd ansible_quickstart
```

単一のディレクトリー構造を使用すると、ソース管理への追加や自動化コンテンツの再利用と共有が容易になります。

1.4. インベントリーのビルド

インベントリーは、Ansible にシステム情報とネットワークの場所を提供する集中ファイルにマネージドノードを編成します。インベントリーファイルを使用すると、Ansible は 1 つのコマンドで多数のホストを管理できます。次の手順を完了するには、少なくとも 1 つのホストシステムの IP アドレスまたは完全修飾ドメイン名 (FQDN) が必要です。デモの目的で、ホストはコンテナーまたは仮想マシン内でローカルに実行できます。

また、公開 SSH 鍵が各ホストの **authorized_keys** ファイルに追加されていることを確認する必要があります。インベントリーを構築するには、以下の手順を使用します。

手順

作成した **ansible_quickstart** ディレクトリーに **inventory.ini** という名前のファイルを作成します。新しい **[myhosts]** グループを inventory.ini ファイルに追加し、各ホストシステムの IP アドレスまたは完全修飾ドメイン名 (FQDN) を指定します。

```
[myhosts]
192.0.2.50
192.0.2.51
192.0.2.52
```

以下を使用してインベントリーを確認します。

```
ansible-inventory -i inventory.ini --list
```

以下を使用して、インベントリー内の **myhosts** グループに ping します。

```
ansible myhosts -m ping -i inventory.ini
```

コントロールノードと管理ノードでユーザー名が異なる場合は、Ansible コマンドに **-u** オプションを渡します。

```
192.0.2.50 | SUCCESS => {
   "ansible_facts": {
      "discovered_interpreter_python": "/usr/bin/python3"
   },
   "changed": false,
   "ping": "pong"
}
192.0.2.51 | SUCCESS => {
   "ansible_facts": {
      "discovered_interpreter_python": "/usr/bin/python3"
   },
   "changed": false,
   "ping": "pong"
}
192.0.2.52 | SUCCESS => {
   "ansible_facts": {
      "discovered_interpreter_python": "/usr/bin/python3"
   },
   "changed": false,
   "ping": "pong"
}
```

インベントリーが正常に作成されました。

1.4.1. INI または YAML 形式のインベントリー

インベントリーは、INI ファイルまたは YAML のいずれかで作成できます。前述の例のように、ほとんどの場合、INI ファイルは単純で、マネージドノードが少数の場合は簡単に確認できます。マネージドノードの数が増えると、YAML 形式でインベントリーを作成すると適切なオプションになります。

たとえば、次のコードは、マネージドノードの一意の名前を宣言し、**ansible_host** フィールドを使用する **inventory.ini** と同等です。

```
myhosts:
  hosts:
    my_host_01:
      ansible_host: 192.0.2.50
    my_host_02:
      ansible_host: 192.0.2.51
    my_host_03:
      ansible_host: 192.0.2.52
```

1.4.2. インベントリーのビルドに関するヒント

- グループ名が一意で、わかりやすいものであることを確認してください。
- グループ名も大文字と小文字が区別されます。
- グループ名にはスペース、ハイフン、または前に数字を付けないでください (19th_floor ではなく、floor_19 を使用してください)。
- What (内容)、Where (場所)、および When (タイミング) に従って、インベントリー内のホストを論理的にグループ化します。
  - What: トポロジーに従ってホストを分類します (例: db、web、leaf、spine)。
  - Where: 地理的な場所でホストを分類します (例: datacenter、region、floor、build)。
  - When: ステージごとにホストを分類します (例: development、test、staging、production)。

1.4.3. メタグループの使用

以下の構文を使用して、インベントリー内に複数のグループを整理するメタグループを作成します。

```
metagroupname:
  children:
```

次のリスト表は、データセンターの基本構造を示しています。以下のインベントリーは、データセンターの基本構造を示しています。このインベントリーの例には、すべてのネットワークデバイスを含む network メタグループと、network グループとすべての Web サーバー含む datacenter メタグループが含まれます。

```
leafs:
  hosts:
    leaf01:
      ansible_host: 192.0.2.100
    leaf02:
      ansible_host: 192.0.2.110

spines:
  hosts:
    spine01:
      ansible_host: 192.0.2.120
    spine02:
      ansible_host: 192.0.2.130

network:
  children:
    leafs:
    spines:

webservers:
  hosts:
    webserver01:
      ansible_host: 192.0.2.140
    webserver02:
      ansible_host: 192.0.2.150

datacenter:
  children:
    network:
    webservers:
```

1.5. 変数の作成

変数は、IP アドレス、FQDN、オペレーティングシステム、SSH ユーザーなどのマネージドノードの値を設定するため、Ansible コマンドの実行時に渡す必要はありません。

変数は特定のホストに適用できます。

```
webservers:
  hosts:
    webserver01:
      ansible_host: 192.0.2.140
      http_port: 80
    webserver02:
      ansible_host: 192.0.2.150
      http_port: 443
```

変数は、グループ内のすべてのホストにも適用できます。

```
webservers:
  hosts:
    webserver01:
      ansible_host: 192.0.2.140
      http_port: 80
    webserver02:
      ansible_host: 192.0.2.150
      http_port: 443
  vars:
    ansible_user: my_server_user
```

インベントリーと Ansible インベントリー変数の詳細は、インストーラインベントリーファイル と インベントリーファイル変数 を参照してください。

1.6. 最初の PLAYBOOK の作成

以下の手順を使用して、ホストに ping を実行し、“Hello world” メッセージを出力する Playbook を作成します。

手順

1. **ansible_quickstart** ディレクトリーに次の内容の **playbook.yaml** という名前のファイルを作成します。

```
- name: My first play
  hosts: myhosts
  tasks:
   - name: Ping my hosts
     ansible.builtin.ping:

   - name: Print message
     ansible.builtin.debug:
      msg: Hello world
```

2. 以下のコマンドを使用して Playbook を実行します。

```
ansible-playbook -i inventory.ini playbook.yaml
```

3. Ansible は次の出力を返します。

```
PLAY [My first play] ****************************************************************************

TASK [Gathering Facts] **************************************************************************
ok: [192.0.2.50]
ok: [192.0.2.51]
ok: [192.0.2.52]

TASK [Ping my hosts] ****************************************************************************
ok: [192.0.2.50]
ok: [192.0.2.51]
ok: [192.0.2.52]

TASK [Print message] ****************************************************************************
ok: [192.0.2.50] => {
    "msg": "Hello world"
}
ok: [192.0.2.51] => {
    "msg": "Hello world"
}
ok: [192.0.2.52] => {
    "msg": "Hello world"
}

PLAY RECAP ****************************************************************************************
192.0.2.50: ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
192.0.2.51: ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
192.0.2.52: ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

この出力で、以下を確認できます。

- プレイおよび各タスクに付けた名前。Playbook の検証とトラブルシューティングが容易になるように、常にわかりやすい名前を使用してください。
- Gather Facts タスクは暗黙的に実行されます。デフォルトでは、Ansible は Playbook で使用できるインベントリーに関する情報を収集します。

- 各タスクのステータス。各タスクのステータスは **ok** で、正常に実行されたことを意味します。
- Playbook 内のすべてのタスクの結果をホストごとにまとめたプレイの要約。この例では、3 つのタスクがあるため、**ok=3** は各タスクが正常に実行されたことを示します。

第2章 PLAYBOOK を使用した管理対象ノードへの接続確立

認証情報を確認するには、手動でネットワークデバイスに接続し、その設定を取得します。サンプルユーザー名とデバイス名は、実際の認証情報に置き換えます。

たとえば、VyOS ルーターの場合は以下のようになります。

```
ssh my_vyos_user@vyos.example.net
show config
exit
```

2.1. ネットワークの ANSIBLE コマンドの実行

ネットワークデバイスに手動で接続してコマンドを実行する代わりに、単一の Ansible コマンドでその設定を取得できます。

```
ansible all -i vyos.example.net, -c ansible.netcommon.network_cli -u \
my_vyos_user -k -m vyos.vyos.vyos_facts -e \
ansible_network_os=vyos.vyos.vyos
```

このコマンドのフラグは、7 つの値を設定します。

- コマンドを適用するホストグループ (この場合は **all**)
- インベントリー (**-i**、ターゲットに設定するデバイス (最後のコンマがない **-i** はインベントリーファイルの指定がありません))
- 接続方法 (**-c**、ansible の接続方法および実行方法)
- ユーザー (**-u**、SSH 接続のユーザー名)
- SSH 接続方法 (**-k**、パスワードの入力を求めるプロンプト)
- モジュール (**-m**、完全修飾コレクション名 (FQCN) を使用した、実行する Ansible モジュール)
- 追加変数 (**-e**、この場合はネットワーク OS 値の設定)

注記

ssh 鍵で **ssh-agent** を使用する場合は、Ansible が自動的にそれらを読み込みます。**-k** フラグは省略できます。

仮想環境で Ansible を実行している場合は、変数 **ansible_python_interpreter=/path/to/venv/bin/python** も追加する必要があります。

2.2. ネットワーク ANSIBLE PLAYBOOK の実行

毎日特定のコマンドを実行する場合は、Playbook に保存し、ansible ではなく **ansible-playbook** を使用して実行できます。Playbook は、コマンドラインでフラグを使用して指定したパラメーターを多数保存し、コマンドラインで入力するようにできます。これには、Playbook とインベントリーファイルという 2 つのファイルが必要です。

前提条件

ここ から **first_playbook.yml** をダウンロードします。

Playbook は以下のようになります。

```
---

- name: Network Getting Started First Playbook
  connection: ansible.netcommon.network_cli
  gather_facts: false            1
  hosts: all
  tasks:

    - name: Get config for VyOS devices
      vyos.vyos.vyos_facts:
        gather_subset: all

    - name: Display the config
      debug:
        msg: "The hostname is {{ ansible_net_hostname }} and the OS is {{ ansible_net_version }}"
```

ラベル	説明
gather_facts	Playbook はこのネットワークコレクション内のプラットフォーム固有のモジュール (**vyos.vyos.vyos_facts**) によって提供されるファクトに依存しているため、ここでは Ansible のネイティブファクト収集 (**ansible.builtin.setup**) が無効になっています。

Playbook は、上記のコマンドラインから 7 つの値のうち 3 つを設定します。

- グループ (**ホスト: すべて**)
- 接続方法 (**connection: ansible.netcommon.network_cli**) と
- モジュール (各タスク内)。

Playbook でこれらの値を設定すると、コマンドラインで値を省略できます。この Playbook では、設定出力を表示する 2 番目のタスクも追加します。

vyos.vyos.vyos_facts や **ansible.builtin.setup** などのコレクション固有のファクトモジュールを通じてシステムからファクトが収集されると、収集されたデータはコンソールに書き込まれるのではなく、今後のタスクで使用できるようにメモリーに保持されます。

モジュールが Playbook で実行されると、出力はコンソールに書き込まれるのではなく、今後のタスクで使用できるようにメモリーに保持されます。他のほとんどのモジュールでは、モジュールまたはタスクの出力を保存して再利用するには変数を明示的に登録する必要があります。

ファクトの詳細は、**Ansiible Playbook リファレンスガイド** の [Ansible ファクト] を参照してください。

次のデバッグタスクを使用すると、シェルで結果を確認できます。

手順

1. 以下のコマンドで Playbook を実行します。

```
ansible-playbook -i vyos.example.net, -u ansible -k -e ansible_network_os=vyos.vyos.vyos
first_playbook.yml
```

Playbook には 2 つのタスクを含むプレイが 1 つ含まれており、次のような出力を生成します。

```
$ ansible-playbook -i vyos.example.net, -u ansible -k -e ansible_network_os=vyos.vyos.vyos
first_playbook.yml

PLAY [Network Getting Started First Playbook]
**************************************************************************************************************
**********

TASK [Get config for VyOS devices]
**************************************************************************************************************
**********
ok: [vyos.example.net]

TASK [Display the config]
**************************************************************************************************************
**********
ok: [vyos.example.net] => {
    "msg": "The hostname is vyos and the OS is VyOS 1.1.8"
}
```

2. デバイス設定を取得できるようになったので、Ansible を使用して更新してみることができます。

3. ここ から **first_playbook_ext.yml** をダウンロードします。これは、最初の Playbook の拡張バージョンです。
   Playbook は以下のようになります。

```
---

- name: Network Getting Started First Playbook Extended
  connection: ansible.netcommon.network_cli
  gather_facts: false
  hosts: all
  tasks:

    - name: Get config for VyOS devices
      vyos.vyos.vyos_facts:
        gather_subset: all

    - name: Display the config
      debug:
        msg: "The hostname is {{ ansible_net_hostname }} and the OS is {{ ansible_net_version
}}"

    - name: Update the hostname
      vyos.vyos.vyos_config:
        backup: yes
        lines:
          - set system host-name vyos-changed

    - name: Get changed config for VyOS devices
```

```
      vyos.vyos.vyos_facts:
        gather_subset: all

    - name: Display the changed config
      debug:
        msg: "The new hostname is {{ ansible_net_hostname }} and the OS is {{
ansible_net_version }}"
```

4. 拡張された最初の Playbook には、1 回のプレイに 5 つのタスクがあります。

5. 以下のコマンドで Playbook を実行します。

```
$ ansible-playbook -i vyos.example.net, -u ansible -k -e ansible_network_os=vyos.vyos.vyos
first_playbook_ext.yml
```

6. 出力には、Ansible が設定に加えた変更が表示されます。

```
$ ansible-playbook -i vyos.example.net, -u ansible -k -e ansible_network_os=vyos.vyos.vyos
first_playbook_ext.yml

PLAY [Network Getting Started First Playbook Extended]
***************************************************************************************************************
********************


TASK [Get config for VyOS devices]
***************************************************************************************************************
******************
ok: [vyos.example.net]

TASK [Display the config]
***************************************************************************************************************
*********************
ok: [vyos.example.net] => {
    "msg": "The hostname is vyos and the OS is VyOS 1.1.8"
}

TASK [Update the hostname]
***************************************************************************************************************
*********************
changed: [vyos.example.net]

TASK [Get changed config for VyOS devices]
***************************************************************************************************************
*********************
ok: [vyos.example.net]

TASK [Display the changed config]
***************************************************************************************************************
*********************
ok: [vyos.example.net] => {
    "msg": "The new hostname is vyos-changed and the OS is VyOS 1.1.8"
}

PLAY RECAP
```

```
*******************************************************************************************************************
*******************
vyos.example.net          : ok=5    changed=1    unreachable=0    failed=0
```

2.3. ネットワークデバイスからのファクトの収集

Gather_facts キーワードは、標準化されたキーと値のペアで、ネットワークデバイスファクトを収集しやすくなります。これらのネットワークファクトをさらなるタスクにフィードして、ネットワークデバイスを管理できます。以下に示すように、ネットワーク ***_facts** モジュール (**arista.eos.eos_facts** など) で **Gather_network_resources** パラメーターを使用して、デバイス設定のサブセットを返すこともできます。

```
- hosts: arista
  gather_facts: True
  gather_subset: interfaces
  module_defaults:
    arista.eos.eos_facts:
      gather_network_resources: interfaces
```

Playbook は以下のインターフェイスのファクトを返します。

```
"network_resources": {
      "interfaces": [
          {
              "description": "test-interface",
              "enabled": true,
              "mtu": "512",
              "name": "Ethernet1"
          },
          {
              "enabled": true,
              "mtu": "3000",
              "name": "Ethernet2"
          },
          {
              "enabled": true,
              "name": "Ethernet3"
          },
          {
              "enabled": true,
              "name": "Ethernet4"
          },
          {
              "enabled": true,
              "name": "Ethernet5"
          },
          {
              "enabled": true,
              "name": "Ethernet6"
          },
      ]
  }
```

注記

Gather_network_resources は、サポートされているすべてのリソース (**interfaces/bgp/ospf/etc**`) のファクトとして設定データをレンダリングしますが、**gather_subset** は主に運用データをフェッチするために使用されます。

これらのファクトを保存し、eos_interfaces リソースモジュールなどの別のタスクで直接使用できます。

第3章 ANSIBLE PLAYBOOK の実用的な例

Ansible は、クラウドベースの REST API から Linux および Windows システム、ネットワークハードウェアなど、さまざまなデバイス区分と通信できます。

以下は、2 種類のサーバーを自動更新する 2 つの Ansible モジュールのサンプルです。

3.1. PLAYBOOK 実行

Playbook は、上から下へ順番に実行されます。各プレイ内では、タスクは上から下に順番に実行されます。複数のプレイを含む Playbook は、Web サーバー上で 1 つのプレイを実行し、次にデータベースサーバー上で別のプレイを実行し、次にネットワークインフラストラクチャー上で 3 番目のプレイを実行するなど、複数マシンのデプロイメントを調整できます。

少なくとも、各プレイは 2 つのことを定義します。

- ターゲットに設定する管理対象ノード (パターンの使用)
- 実行すべき 1 つ以上のタスク

注記

Ansible 2.10 以降では、複数のコレクションに同じ名前のモジュールが含まれていることがあるため (例: **user**)、正しいモジュールが選択されるように、Playbook で完全修飾コレクション名を使用します。

詳細は、Playbook でのコレクションの使用 を参照してください。

この例では、第 1 のプレイは Web サーバーを対象とし、第 2 のプレイはデータベースサーバーを対象としています。

```
---
- name: Update web servers
  hosts: webservers
  become: true

  tasks:
    - name: Ensure apache is at the latest version
      ansible.builtin.yum:
        name: httpd
        state: latest
    - name: Write the apache config file
      ansible.builtin.template:
        src: /srv/httpd.j2
        dest: /etc/httpd.conf
        mode: "0644"

- name: Update db servers
  hosts: databases
  become: true

  tasks:
    - name: Ensure postgresql is at the latest version
      ansible.builtin.yum:
```

```
      name: postgresql
      state: latest
  - name: Ensure that postgresql is started
    ansible.builtin.service:
      name: postgresql
      state: started
```

Playbook には 2 つのプレイが含まれます。

- 最初に、Web サーバーソフトウェアが最新かどうかを確認し、必要に応じて更新を実行します。
- 次に、データベースサーバーソフトウェアが最新の状態かどうかを確認し、必要に応じて更新を実行します。

Playbook には、ホスト行とタスク以外のものを含めることができます。

たとえば、このサンプル Playbook では、プレイごとに Remote_user を設定します。SSH 接続用のユーザーアカウントです。他の Playbook キーワードを Playbook、Play、またはタスクレベルで追加して、Ansible の動作を操作できます。Playbook キーワードは、接続プラグイン、権限昇格を使用するかどうか、エラーの処理方法などを制御できます。

さまざまな環境をサポートするために、Ansible では、Ansible 設定またはインベントリーでこれらのパラメーターの多くをコマンドラインフラグとして設定できます。これらのデータソースの優先ルールを学習すると、Ansible エコシステムを拡張するときに役立ちます。

第4章 オープンソースライセンス

GNU GENERAL PUBLIC LICENSE

バージョン 3、2007 年 6 月 29 日

Preamble
The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works.対照的に、GNU General Public License は、プログラムのすべてのバージョンを共有および変更する自由を保証することを目的としています。これにより、プログラムがすべてのユーザーにとって自由なソフトウェアであり続けることが保証されます。We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors.You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price.Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights.Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received.You must make sure that they, too, receive or can get the source code.And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software.For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so.This is fundamentally incompatible with the aim of protecting users' freedom to change the software.The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable.Therefore, we have designed this version of the GPL to prohibit the practice for those products.If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents.States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary.To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License.Each licensee is addressed as “you”.“Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy.The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy.Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies.Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License.If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it.“Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form.A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities.However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work.For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met.This License explicitly affirms your unlimited permission to run the unmodified Program.The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work.This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force.You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright.Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below.Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7.This requirement modifies the requirement in section 4 to “keep intact all notices”.
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy.This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged.This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation’s users beyond what the individual works permit.Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
- c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source.This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
- d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge.You need not require recipients to copy the Corresponding Source along with the object code.If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source.Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

- e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling.In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage.For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product.A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source.The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information.But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed.Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions.Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law.If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it.(Additional permissions may be written to require their own removal in certain cases when you modify the work.)You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10.If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term.If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License.Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License.If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program.Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance.However, nothing other than this License grants you permission to propagate or modify any covered work.These actions infringe copyright if you do not accept this License.Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License.You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations.If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License.For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based.The work thus licensed is called the contributor’s “contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version.For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement).To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients.“Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License.You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License.If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all.For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work.The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time.Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number.If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation.If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions.However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU.SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program.It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program.  If not, see <https://www.gnu.org/licenses/>.
```

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program>  Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License.Of course, your program’s commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary.For more information on this, and how to apply and follow the GNU GPL, see <https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs.If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library.If this is what you want to do, use the GNU Lesser General Public License instead of this License.But first, please read <https://www.gnu.org/licenses/why-not-lgpl.html>.