Quick Starts are automated reference deployments that use AWS CloudFormation templates to deploy key technologies on Amazon Web Services (AWS), following AWS best practices.

Quick Links

The links in this section are for your convenience. Before you launch the Quick Start, please review the architecture, security, and other considerations discussed in this guide.

- If you have an AWS account, and you’re already familiar with AWS services and AWX, you can launch the Quick Start to build the architecture shown in Figure 1 in a new or existing virtual private cloud (VPC). The deployment takes approximately 45 minutes. If you’re new to AWS or to AWX, please review the implementation details and follow the step-by-step instructions provided later in this guide.

- If you want to take a look under the covers, you can view the AWS CloudFormation templates that automate the deployment.

Overview

This Quick Start reference deployment guide provides step-by-step instructions for deploying AWX on the AWS Cloud. It is for users who want to run and manage an AWX infrastructure on AWS.

AWX is an open-source community project that provides software for managing Ansible projects. AWX is hosted on GitHub and provides a web-based user interface, REST API, and task engine for Ansible. The visual AWX dashboard lets you schedule and deploy Ansible playbooks, and provides centralized logging, auditing, and system tracking. AWX provides the source code for Ansible Tower, which is the commercial version of AWX.

This Quick Start deploys code from the master branch of the GitHub repository for AWX. You can also choose to deploy code from a fork of the repository.
Ansible is an IT DevOps tool that automates provisioning, configuration management, application deployment, intra-service orchestration, continuous delivery, and many other IT processes. Ansible is designed for multi-tier deployments. Instead of managing systems individually, it models your IT infrastructure by describing the inter-relationships among all your systems.

A key advantage to Ansible over other automation engines is that it uses no agents and no additional custom security infrastructure, which simplifies deployment. Ansible uses a very simple, human-readable language called YAML for Ansible playbooks, to manage configuration, deployment, and orchestration tasks. Ansible works by connecting to your nodes and running small programs, called Ansible modules, to configure the resource for your system. Ansible executes these modules over Secure Shell (SSH) by default, and removes them when finished.

Costs and Licenses

You are responsible for the cost of the AWS services used while running this Quick Start reference deployment. There is no additional cost for using the Quick Start.

The AWS CloudFormation template for this Quick Start includes configuration parameters that you can customize. Some of these settings, such as instance type, will affect the cost of deployment. For cost estimates, see the pricing pages for each AWS service you will be using. Prices are subject to change.

Ansible, which is installed as part of the deployment, is licensed under the GNU General Public License version 3.

AWX is open-source software. It is distributed under the Apache version 2.0 license and is free to use.

Architecture

Deploying this Quick Start for a new virtual private cloud (VPC) with default parameters builds the following AWX environment in the AWS Cloud.
The Quick Start sets up the following:

- A highly available architecture that spans two Availability Zones.*
- A VPC configured with public and private subnets according to AWS best practices, to provide you with your own virtual network on AWS.*
- An internet gateway to allow access to the internet.*
- In the public subnets, managed NAT gateways to allow outbound internet access for resources in the private subnets.*
- In the private subnets, an Amazon Elastic Container Service (Amazon ECS) cluster in an Auto Scaling group across the two Availability Zones.
- In the private subnets, an Amazon Relational Database Service (Amazon RDS) PostgreSQL database.
An AWS CodeBuild project to build the AWX project from the official AWX Github repository.

* The template that deploys the Quick Start into an existing VPC skips the tasks marked by asterisks and prompts you for your existing VPC configuration.

**Prerequisites**

**Specialized Knowledge**

Before you deploy this Quick Start, we recommend that you become familiar with the following AWS services. (If you are new to AWS, see [Getting Started with AWS](https://aws.amazon.com)).

- Amazon EC2
- Amazon ECS
- Amazon RDS
- Amazon VPC
- AWS CloudFormation
- AWS CodeBuild

**Deployment Options**

This Quick Start provides two deployment options:

- **Deploy AWX into a new VPC** (end-to-end deployment). This option builds a new AWS environment consisting of the VPC, subnets, NAT gateways, security groups, and other infrastructure components, and then deploys AWX into this new VPC.

- **Deploy AWX into an existing VPC**. This option provisions AWX in your existing AWS infrastructure.

The Quick Start provides separate templates for these options. It also lets you configure CIDR blocks, instance types, and AWX settings, as discussed later in this guide.

**Deployment Steps**

**Step 1. Prepare Your AWS Account**

1. If you don’t already have an AWS account, create one at [https://aws.amazon.com](https://aws.amazon.com) by following the on-screen instructions.

2. Use the region selector in the navigation bar to choose the AWS Region where you want to deploy AWX on AWS.
Important This Quick Start includes AWS CodeBuild, which isn’t available in all AWS Regions. See the list of supported regions.

3. Create a key pair in your preferred region.

4. If necessary, request a service limit increase for the EC2 instance type that you’re planning to use for the deployment. You might need to do this if you already have an existing deployment that uses this instance type, and you think you might exceed the default limit with this deployment.

Step 2. Create an OAuth Token for CodeBuild to Access GitHub

An OAuth token is required to build the AWX project.

1. Open the AWS CodeBuild console at https://console.aws.amazon.com/codebuild/, and then choose Get started.

   ![AWS CodeBuild console](image)

   Figure 2: AWS CodeBuild console

2. On the Configure your project page, type a project name.

3. In Source: What to build, for source provider, choose GitHub. For repository, choose Connect to GitHub.
Figure 3: Connecting to GitHub

4. In the **Authorize AWS CodeBuild** prompt, choose **Authorize aws-codesuite**. This authorizes CodeBuild to access your GitHub account. You will be prompted to confirm your GitHub password.

Figure 4: Authorizing CodeBuild to access your GitHub account

5. In the CodeBuild console, close the **Create project** page and continue with step 3.
Step 3. Launch the Quick Start

**Note** You are responsible for the cost of the AWS services used while running this Quick Start reference deployment. There is no additional cost for using this Quick Start. For full details, see the pricing pages for each AWS service you will be using in this Quick Start. Prices are subject to change.

1. Choose one of the following options to launch the AWS CloudFormation template into your AWS account. For help choosing an option, see deployment options earlier in this guide.

<table>
<thead>
<tr>
<th>Option 1</th>
<th>Option 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deploy AWX into a new VPC on AWS</td>
<td>Deploy AWX into an existing VPC on AWS</td>
</tr>
<tr>
<td><img src="Launch" alt="Launch" /></td>
<td><img src="Launch" alt="Launch" /></td>
</tr>
</tbody>
</table>

**Important** If you’re deploying AWS into an existing VPC, make sure that your VPC has two private subnets in different Availability Zones for the database instances. These subnets require NAT gateways or NAT instances in their route tables, to allow the instances to download packages and software without exposing them to the internet. You will also need the domain name option configured in the DHCP options as explained in the Amazon VPC documentation. You will be prompted for your VPC settings when you launch the Quick Start.

The deployment takes about 45 minutes to complete.

2. Check the region that’s displayed in the upper-right corner of the navigation bar, and change it if necessary. This is where the network infrastructure for AWX will be built. The template is launched in the US East (Ohio) Region by default.

**Important** This Quick Start includes AWS CodeBuild, which isn’t available in all AWS Regions. See the list of supported regions.

3. On the Select Template page, keep the default setting for the template URL, and then choose Next.


4. On the **Specify Details** page, change the stack name if needed. Review the parameters for the template. Provide values for the parameters that require input. For all other parameters, review the default settings and customize them as necessary. When you finish reviewing and customizing the parameters, choose **Next**.

In the following tables, parameters are listed by category and described separately for the two deployment options. You can also download the AWS CloudFormation template that automates the deployment and customize it for your specific scenario.

- **Parameters for deploying AWX into a new VPC**
- **Parameters for deploying AWX into an existing VPC**

**Option 1: Parameters for deploying AWX into a new VPC**

**View template**

**Network Configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability Zones (AvailabilityZones)</td>
<td>Requires input</td>
<td>The list of Availability Zones to use for the subnets in the VPC. The Quick Start uses two Availability Zones from your list preserves the logical order you specify.</td>
</tr>
<tr>
<td>VPC CIDR (VPCCIDR)</td>
<td>10.0.0.0/16</td>
<td>The CIDR block for the VPC.</td>
</tr>
<tr>
<td>Public Subnet 1 CIDR (PublicSubnet1CIDR)</td>
<td>10.0.128.0/20</td>
<td>The CIDR block for the public (DMZ) subnet located in Availability Zone 1.</td>
</tr>
<tr>
<td>Public Subnet 2 CIDR (PublicSubnet2CIDR)</td>
<td>10.0.144.0/20</td>
<td>The CIDR block for the public (DMZ) subnet located in Availability Zone 2.</td>
</tr>
<tr>
<td>Private Subnet 1 CIDR (PrivateSubnet1CIDR)</td>
<td>10.0.0.0/19</td>
<td>The CIDR block for the private subnet located in Availability Zone 1.</td>
</tr>
<tr>
<td>Private Subnet 2 CIDR (PrivateSubnet2CIDR)</td>
<td>10.0.32.0/19</td>
<td>The CIDR block for the private subnet located in Availability Zone 2.</td>
</tr>
<tr>
<td>Allowed Bastion External Access CIDR (RemoteAccessCIDR)</td>
<td>Requires input</td>
<td>The CIDR IP range that is permitted to access AWX. We recommend that you set this value to a trusted IP range. For example, you might want to grant only your corporate network access to the software.</td>
</tr>
</tbody>
</table>

**Amazon EC2 Configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key Pair Name (KeyPairName)</td>
<td>Requires input</td>
<td>A public/private key pair, which allows you to connect securely to your instance after it launches. When you created an AWS</td>
</tr>
<tr>
<td>Parameter label (name)</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>account, this is the key pair you created in your preferred region.</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Cluster Size (ClusterSize)</strong></td>
<td>2</td>
<td>The number of Amazon ECS hosts to deploy initially.</td>
</tr>
<tr>
<td><strong>DBInstance Class (DBInstanceClass)</strong></td>
<td>db.t2.medium</td>
<td>The compute and memory capacity of the database instance.</td>
</tr>
<tr>
<td><strong>Instance Type (InstanceType)</strong></td>
<td>m4.large</td>
<td>The EC2 instance type to use to build the ECS cluster.</td>
</tr>
</tbody>
</table>

**Amazon RDS Database Configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Master DB Username (MasterUsername)</strong></td>
<td>Requires input</td>
<td>The user name for the master PostgreSQL database.</td>
</tr>
<tr>
<td><strong>Master DB Password (MasterUserPassword)</strong></td>
<td>Requires input</td>
<td>The password for the master PostgreSQL database.</td>
</tr>
<tr>
<td><strong>Daily Backup Window (PreferredBackup Window)</strong></td>
<td>00:00-02:00</td>
<td>The daily time range, in UTC, during which automated backups of the PostgreSQL database are created (if automated backups are enabled in Amazon RDS). This parameter must not overlap with the time you set for the preferred maintenance window.</td>
</tr>
<tr>
<td><strong>Maintenance period - day of week (PreferredMaintenance WindowDay)</strong></td>
<td>Mon</td>
<td>The day of the week when maintenance will be performed on the PostgreSQL database.</td>
</tr>
<tr>
<td><strong>Maintenance period - start time (PreferredMaintenance WindowStartTime)</strong></td>
<td>04:00</td>
<td>The weekly start time, in UTC, for PostgreSQL database maintenance. This must be before the time you set for the <strong>Maintenance period – end time</strong> parameter, and it must not overlap with the <strong>Daily Backup Window</strong> parameter setting.</td>
</tr>
<tr>
<td><strong>Maintenance period - end time (PreferredMaintenance WindowEndTime)</strong></td>
<td>06:00</td>
<td>The weekly end time, in UTC, for the PostgreSQL database maintenance. This must be after the time you set for the <strong>Maintenance period – start time</strong> parameter, and it must not overlap with the <strong>Daily Backup Window</strong> parameter setting.</td>
</tr>
</tbody>
</table>
### AWX Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWX Admin Username (AWXAdminUsername)</td>
<td>admin</td>
<td>The administrator user name for accessing AWX.</td>
</tr>
<tr>
<td>AWX Admin Password (AWXAdminPassword)</td>
<td>Requires input</td>
<td>The administrator password for accessing AWX.</td>
</tr>
<tr>
<td>AWX GitHub Repo (AWXGitHubRepo)</td>
<td><a href="https://github.com/ansible/awx.git">https://github.com/ansible/awx.git</a></td>
<td>The GitHub repository to use as the source for the build. By default, the code is installed from the master branch. Change this parameter if you want to install code from a fork.</td>
</tr>
<tr>
<td>AWX Version (AWXVersion)</td>
<td>1.0.1</td>
<td>The version of AWX to deploy.</td>
</tr>
</tbody>
</table>

### AWS Quick Start Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quick Start S3 Bucket Name (QSS3BucketName)</td>
<td>aws-quickstart</td>
<td>The S3 bucket you have created for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. The bucket name can include numbers, lowercase letters, uppercase letters, and hyphens, but should not start or end with a hyphen.</td>
</tr>
<tr>
<td>Quick Start S3 Key Prefix (QSS3KeyPrefix)</td>
<td>quickstart-awx/</td>
<td>The S3 key name prefix used to simulate a folder for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. This prefix can include numbers, lowercase letters, uppercase letters, hyphens, and forward slashes.</td>
</tr>
</tbody>
</table>

- **Option 2:** Parameters for deploying AWX into an existing VPC

  View template

### Network Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>The ID of your existing VPC (VPC)</td>
<td>Requires input</td>
<td>The ID of your existing VPC (e.g., vpc-0343606e).</td>
</tr>
<tr>
<td>Public DMZ subnet 1 in Availability Zone 1 (PublicSubnet1ID)</td>
<td>Requires input</td>
<td>The ID of the public subnet in Availability Zone 1 in your existing VPC (e.g., subnet-a0246dcd).</td>
</tr>
<tr>
<td>Parameter label (name)</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------------------------------------------------</td>
<td>------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Public DMZ subnet 2 in Availability Zone 2 (PublicSubnet2ID)</td>
<td>Requires input</td>
<td>The ID of the public subnet in Availability Zone 2 in your existing VPC (e.g., subnet-b58c3d67).</td>
</tr>
<tr>
<td>Private subnet 1 in Availability Zone 1 (PrivateSubnet1ID)</td>
<td>Requires input</td>
<td>The ID of the private subnet in Availability Zone 1 in your existing VPC (e.g., subnet-a0246dcd).</td>
</tr>
<tr>
<td>Private subnet 2 in Availability Zone 2 (PrivateSubnet2ID)</td>
<td>Requires input</td>
<td>The ID of the private subnet in Availability Zone 2 in your existing VPC (e.g., subnet-b58c3d67).</td>
</tr>
<tr>
<td>Allowed Bastion External Access CIDR (RemoteAccessCIDR)</td>
<td>Requires input</td>
<td>The CIDR IP range that is permitted to access AWX. We recommend that you set this value to a trusted IP range. For example, you might want to grant only your corporate network access to the software.</td>
</tr>
</tbody>
</table>

**Amazon EC2 Configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key Pair Name (KeyPairName)</td>
<td>Requires input</td>
<td>A public/private key pair, which allows you to connect securely to your instance after it launches. When you created an AWS account, this is the key pair you created in your preferred region.</td>
</tr>
<tr>
<td>Cluster Size (ClusterSize)</td>
<td>2</td>
<td>The number of Amazon ECS hosts to deploy initially.</td>
</tr>
<tr>
<td>Instance Type (InstanceType)</td>
<td>m4.large</td>
<td>The EC2 instance type to use to build the ECS cluster.</td>
</tr>
</tbody>
</table>

**Amazon RDS Database Configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon RDS Access CIDR (RDSAccessCidr)</td>
<td>Requires input</td>
<td>The CIDR IP range that is permitted to access the Amazon RDS database.</td>
</tr>
<tr>
<td>Master DB Username (MasterUsername)</td>
<td>Requires input</td>
<td>The user name for the master PostgreSQL database.</td>
</tr>
<tr>
<td>Master DB Password (MasterUserPassword)</td>
<td>Requires input</td>
<td>The password for the master PostgreSQL database.</td>
</tr>
<tr>
<td>Daily Backup Window (PreferredBackup Window)</td>
<td>00:00-02:00</td>
<td>The daily time range, in UTC, during which automated backups of the PostgreSQL database are created (if automated backups are enabled in Amazon RDS). This parameter must</td>
</tr>
</tbody>
</table>
### Parameter label (name) | Default | Description
--- | --- | ---
not overlap with the time you set for the preferred maintenance window.

**Maintenance period - day of week**
(PreferredMaintenanceWindowDay)

- **Default**: Mon
- **Description**: The day of the week when maintenance will be performed on the PostgreSQL database.

**Maintenance period - start time**
(PreferredMaintenanceWindowStartTime)

- **Default**: 04:00
- **Description**: The weekly start time, in UTC, for PostgreSQL database maintenance. This must be before the time you set for the **Maintenance period – end time** parameter, and it must not overlap with the **Daily Backup Window** parameter setting.

**Maintenance period - end time**
(PreferredMaintenanceWindowEndTime)

- **Default**: 06:00
- **Description**: The weekly end time, in UTC, for the PostgreSQL database maintenance. This must be after the time you set for the **Maintenance period – start time** parameter, and it must not overlap with the **Daily Backup Window** parameter setting.

**DBInstance Class**
(DBInstanceClass)

- **Default**: db.t2.medium
- **Description**: The compute and memory capacity of the database instance.

### AWX Configuration:

| Parameter label (name) | Default | Description |
--- | --- | ---
**AWX Admin Username**
(AWXAdminUsername)

- **Default**: admin
- **Description**: The administrator user name for accessing AWX.

**AWX Admin Password**
(AWXAdminPassword)

- **Default**: Requires input
- **Description**: The administrator password for accessing AWX.

**AWX GitHub Repo**
(AWXGitHubRepo)

- **Default**: https://github.com/ansible/awx.git
- **Description**: The GitHub repository to use as the source for the build. By default, the code is installed from the master branch. Change this parameter if you want to install code from a fork.

**AWX Version**
(AWXVersion)

- **Default**: 1.0.1
- **Description**: The version of AWX to deploy.

### AWS Quick Start Configuration:

| Parameter label (name) | Default | Description |
--- | --- | ---
**Quick Start S3 Bucket Name**
(QSS3BucketName)

- **Default**: aws-quickstart
- **Description**: The S3 bucket you have created for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. The bucket name can include numbers,
<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quick Start S3 Key Prefix (QSS3KeyPrefix)</td>
<td>quickstart-awx/</td>
<td>The S3 key name prefix used to simulate a folder for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. This prefix can include numbers, lowercase letters, uppercase letters, hyphens, and forward slashes.</td>
</tr>
</tbody>
</table>

5. On the **Options** page, you can specify tags (key-value pairs) for resources in your stack and set advanced options. When you’re done, choose **Next**.

6. On the **Review** page, review and confirm the template settings. Under **Capabilities**, select the check box to acknowledge that the template will create IAM resources.

7. Choose **Create** to deploy the stack.

8. Monitor the status of the stack. When the status is **CREATE_COMPLETE**, the AWX cluster is ready.

9. Use the URLs displayed in the **Outputs** tab for the stack to view the resources that were created.

**Step 4. Test the Deployment**

To access the AWX environment, you access the AWX endpoint created during the deployment. When the migration step is complete, you can use the AWX credentials you defined during your deployment to log in to the AWX dashboard.

1. Locate the AWX endpoint address from the **Outputs** tab of the AWS CloudFormation console.

   ![Figure 5: AWX endpoint](image)

2. Use your preferred web browser to open the URL. You will see the AWX migration screen during setup.
When setup is complete, the login page will be displayed.

3. Use the AWX user name and password you specified during the Quick Start launch. The default AWX user name is `admin`.

   This will bring up the AWX dashboard, as shown in Figure 7.

To read about AWX and ways you can use it, see [The Inside Playbook](https://example.com) on the Ansible website.
Troubleshooting

**Q.** I encountered a CREATE FAILED error when I launched the Quick Start.

**A.** If AWS CloudFormation fails to create the stack, we recommend that you relaunch the template with **Rollback on failure** set to **No**. (This setting is under **Advanced** in the AWS CloudFormation console, **Options** page.) With this setting, the stack’s state will be retained and the instance will be left running, so you can troubleshoot the issue. (Look at the log files in %ProgramFiles%\Amazon\EC2ConfigService and C:\cfn\log.)

**Important** When you set **Rollback on failure** to **No**, you will continue to incur AWS charges for this stack. Please make sure to delete the stack when you finish troubleshooting.

For additional information, see [Troubleshooting AWS CloudFormation](https://aws.amazon.com/documentation/cloudformation/troubleshooting/) on the AWS website.

**Q.** I encountered a size limitation error when I deployed the AWS Cloudformation templates.

**A.** We recommend that you launch the Quick Start templates from the links in this guide or from another S3 bucket. If you deploy the templates from a local copy on your computer or from a non-S3 location, you might encounter template size limitations when you create the stack. For more information about AWS CloudFormation limits, see the [AWS documentation](https://aws.amazon.com/documentation/cloudformation/).

**Q.** The Quick Start displayed the following errors for the infrastructure stack and failed to bootstrap the Amazon ECS host instances:

```
Failed to receive 1 resource signal(s) for the current batch. Each resource signal timeout is counted as a FAILURE.
Received 0 SUCCESS signal(s) out of 1. Unable to satisfy 100% MinSuccessfulInstancesPercent requirement.
```

**A.** You might encounter these errors if the private subnets don’t have outbound connectivity. As a result, the Amazon ECS host instances can’t connect to the Amazon ECS service, and the Quick Start doesn’t create the **ECSAutoScaling** resource in the infrastructure stack. If you’re deploying the Quick Start for a new VPC, the Quick Start sets up NAT gateways to ensure connectivity. If you’re deploying the Quick Start into an existing VPC, make sure that the private subnets in your existing AWS environment include NAT gateways or NAT instances in their route tables.
GitHub Repository

You can visit our GitHub repository to download the templates and scripts for this Quick Start, to post your comments, and to share your customizations with others.

Additional Resources

AWS services

- Amazon EC2
  https://aws.amazon.com/documentation/ec2/
- Amazon ECS
  https://aws.amazon.com/documentation/ecs/
- Amazon RDS
  https://aws.amazon.com/documentation/rds/
- Amazon VPC
  https://aws.amazon.com/documentation/vpc/
- AWS CloudFormation
  https://aws.amazon.com/documentation/cloudformation/

AWX documentation

- Ansible documentation
  http://docs.ansible.com/
- The Ansible Project FAQ
  https://www.ansible.com/products/awx-project/faq

Quick Start reference deployments

- AWS Quick Start home page
  https://aws.amazon.com/quickstart/

Document Revisions

<table>
<thead>
<tr>
<th>Date</th>
<th>Change</th>
<th>In sections</th>
</tr>
</thead>
</table>
| October 2018 | Added ability for users to deploy the Quick Start into their own VPC | Deployment Options
|             |                                                       | Parameters for existing VPC                     |
| May 2018    | Initial publication                                    | —                                               |