SAS® Event Stream Manager 6.2 for Linux: Deployment Guide
Contents

Chapter 1 / Introduction ................................................................. 1
  Steps for a Successful Deployment ............................................ 1
  Contact SAS Technical Support ................................................. 2

Chapter 2 / System Requirements .................................................. 3
  Hardware and Operating System Requirements .......................... 3
  Server Software Requirements ................................................. 4
  Security Requirements ............................................................ 5
  Client Requirements .............................................................. 7
  Deployment Tools ................................................................. 8

Chapter 3 / Pre-installation Tasks .................................................. 9
  Prepare for Deployment .......................................................... 10
  Installing from a Mirror Repository ......................................... 10
  Enable Required Ports ........................................................... 12
  Configure SELinux .................................................................. 14
  Perform Linux Tuning ............................................................. 16
  Configure a Proxy Server ....................................................... 19
  Enable the Yum Cache ............................................................ 20
  Install Ansible ...................................................................... 20
  (Optional) Enable Key-Based SSH Authentication ..................... 22
  Create a Playbook ................................................................ 23

Chapter 4 / Installing SAS Event Stream Manager ............................ 27
  Overview .............................................................................. 27
  Edit the Playbook .................................................................. 27
  Install the Software .............................................................. 32

Chapter 5 / Post-installation Tasks .................................................. 35
  Complete SAS Event Stream Manager Setup ............................. 35

Chapter 6 / Validating the Deployment ............................................. 39
  Verify SAS Event Stream Manager Status ............................... 39
  Troubleshoot Service Status .................................................... 39
  Access Log Files .................................................................. 40
  Verify SAS Message Broker ................................................... 40
  Verify SAS Infrastructure Data Server ..................................... 41

Chapter 7 / Completing the Deployment ........................................... 43
  Product Documentation .......................................................... 43

Chapter 8 / Managing Your Software .............................................. 45
  Support for Upgrades ............................................................ 45
  What Is an Update? ............................................................... 45
  What Is an Upgrade? ............................................................... 46
  Updating Your SAS Event Stream Manager Software ............... 46
  Upgrading Your Software ....................................................... 51
Chapter 9 / Uninstalling SAS Event Stream Manager

What deploy-cleanup Does .................................................. 55
Uninstall the Software ...................................................... 55
Introduction

Steps for a Successful Deployment

Before You Begin

- Because the contents of this guide are subject to continual updates, make sure that you have the latest guide. You can always access the latest release of this guide from the following site:

  SAS Viya Deployment Guides

  If you accessed this guide directly from the Software Order Email (SOE), you are viewing the latest guide. If you are viewing a saved copy of the PDF version of this guide, the content might be outdated.

- To use this guide successfully, you should have a working knowledge of Ansible and the Linux operating system.

- SAS Event Stream Manager is a separately licensed web-based client that enables you to manage SAS Event Stream Processing environments. SAS recommends installing SAS Event Stream Manager on a separate machine from the SAS Event Stream Processing server. This guide also provides instructions for installing this optional product.

Step 1 — Prepare for the Deployment

1. Perform one of the following tasks:

   - To update, upgrade, or add software to an existing deployment, go directly to Chapter 8, “Managing Your Software,” on page 45.
To deploy a new instance of the software, continue following these steps.

1. Go to Chapter 2, “System Requirements,” on page 3 to learn about requirements for hardware, software, security, and more.

2. Go to Chapter 3, “Pre-installation Tasks,” on page 9 to prepare the environment before you deploy the software.

### Step 2 — Perform the Deployment

1. Go to Chapter 4, “Installing SAS Event Stream Manager,” on page 27 to edit the playbook and install the software.

2. Go to Chapter 5, “Post-installation Tasks,” on page 35 to perform post-installation configuration.

### Step 3 — Validate and Complete the Deployment

1. Go to Chapter 6, “Validating the Deployment,” on page 39 to verify that the servers were deployed correctly and to locate the log files.

2. Go to Chapter 7, “Completing the Deployment,” on page 43 for best practices after deployment, including where to find additional documentation.

### Contact SAS Technical Support

Technical support is available to all customers who license SAS software. However, you are encouraged to engage your designated on-site SAS support personnel as your first support contact. If your on-site SAS support personnel cannot resolve your issue, have them contact SAS Technical Support to report your problem.

Before you contact SAS Technical Support, explore the SAS Support website at support.sas.com/techsup/. This site offers access to the SAS Knowledge Base, as well as SAS communities, Technical Support contact options, and other support materials that might answer your questions.

When you contact SAS Technical Support, you are required to provide information, such as your SAS site number, company name, email address, and phone number, that identifies you as a licensed SAS software customer.
Hardware and Operating System Requirements

Hardware Requirements

Table 2.1 Minimum Hardware Requirements

<table>
<thead>
<tr>
<th>Item</th>
<th>Recommended Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>2 cores (x86 architecture)</td>
</tr>
<tr>
<td></td>
<td>Intel Xeon chip set with a minimum speed of 2.6 GHz</td>
</tr>
<tr>
<td>Memory</td>
<td>16 GB of RAM</td>
</tr>
<tr>
<td></td>
<td>Memory clock speed of 1600 MHz</td>
</tr>
</tbody>
</table>
### Supported Operating Systems

SAS Event Stream Manager runs on 64-bit Linux operating systems.

For the full list of supported platforms, see: https://support.sas.com/en/documentation/third-party-software-reference/viya/35/support-for-operating-systems.html.

### Server Software Requirements

#### Java

A Java Runtime Environment (JRE) must be installed on every machine in your deployment. The playbook checks for a pre-installed version of Java that meets or exceeds the requirements. If one is found, it is used. Otherwise, the playbook attempts to install a recent version of OpenJDK and to set the path in a system configuration file. You can also specify the path to an existing JRE in the vars.yml file before you run your playbook.

Java 1.8 is required for both SAS Event Stream Processing and SAS Event Stream Manager.

#### Apache httpd

The deployment process automatically installs Apache httpd on the machines that you designate as targets for the HTTP proxy installation unless it has already been installed. Apache httpd with the mod_ssl module is required in order to create the Apache HTTP Server, which provides security and load balancing for multiple SAS Viya components. This server is also referred to as the *reverse proxy server* in this guide.

SAS recommends that you install Apache httpd and configure the Apache HTTP Server to use certificates that comply with the security policies at your enterprise before you start the deployment process. The playbook will automatically configure the certificates to secure the server.

The Apache HTTP Server must be dedicated to a single SAS Viya deployment.
Security Requirements

Verify that the following prerequisites have been met before you start the deployment:

- An LDAP server, for user authentication.
  
  The SAS Logon Manager component provides logon services for SAS Event Stream Manager. It requires LDAP.

- Administrator privileges for the Linux machine where you are launching the SAS software deployment.

- Super user (sudo) access for the user account that you are using for the deployment. To verify that the user ID is included in the sudoers file, run the following command:

  ```
sudo -v
  ```

  To verify your sudoers privileges, run the following command:

  ```
sudo -l
  ```

  **Note:** The ability to start a shell (with the `!SHELL` entry in some sudoers files) as root is not required.

After the installation has completed, administrator privileges are not required in order to run SAS Event Stream Manager. User accounts are managed in LDAP.

LDAP Requirements

Read access to your LDAP provider is required for SAS Event Stream Manager.

SAS Viya requires a userDN and password in order to bind to the LDAP server. Anonymous binding is supported for clients that are authenticating to the LDAP server.

If the mail attribute is specified for LDAP accounts, it must have a non-null value that is unique for each user.

LDAPS is supported, but the required certificates are not configured automatically by the deployment process.

To configure LDAP to enable access to SAS Event Stream Manager, follow the steps in “Configure LDAP Settings” on page 32 before you run the playbook.
Transport Layer Security

Transport Layer Security (TLS) is applied to many of the network connections in your deployment. These connections are secured by SAS Secret Manager, which is provided by HashiCorp Vault. In a full deployment that is also fully compliant with SAS security standards, the certificates are all signed by a Vault-generated root CA and an intermediate certificate.

The deployment process provides a default level of data encryption. However, you should perform several additional actions to increase the level of security on your systems.

How Default Security Is Applied

An Apache HTTP server acts as a reverse proxy server to secure your environment. Default security settings use the Apache mod_ssl module to secure the server with self-signed certificates.

The playbook can automatically install Apache httpd with the mod_ssl module. This option uses default Apache security settings and self-signed certificates. These settings are reasonably secure, but they are not compliant with SAS security standards.

The playbook also inspects any existing certificates and the CA chain to determine whether they comply with SAS security requirements. If compliant certificates are found, they are used without changes. If only the default mod_ssl is found, the playbook generates a self-signed certificate and configures mod_ssl to use it.

You can add your own certificates after the completion of the deployment process, which will require a brief outage. If you do not add compliant certificates and instead keep the default security settings and certificates, end users will see a standard web browser warning message. SAS recommends replacing the certificates before giving end users access to the software.

The deployment process provides a default level of encryption for data in motion (transmitted data). SAS Viya components attempt to use the highest level of the TLS protocol that the operating system library supports, up to TLS 1.2. The OpenSSL implementation is used for TLS protocols. SAS Viya components attempt to use cipher suites that ensure Perfect Forward Secrecy and that provide the highest level of security that the host can support. For more information, see TLS Encryption.

Enhance Default Security Settings

SAS recommends that you enhance the default security that is applied by the playbook. As a best practice, follow these steps before you start the deployment process:

1. Install the Apache httpd module and the Apache mod_ssl module on all the web servers in your environment.

2. Add certificates that conform to the policies at your enterprise.
Specify the location of the intermediate certificates and the root CA when you edit the playbook. For more information, see “Specify the Path to Certificates” on page 30.

The playbook can then enhance the security of your SAS software deployment automatically. It detects the CA chain that is configured for mod_ssl and incorporates it into the truststores for all other machines in your deployment. On machines that are targets for Consul deployment, the playbook performs additional security configuration.

(Optional) You can also perform these actions after the playbook has been run:
- Block external connections to port 80.
- Use HTTPS for access to SAS user interfaces from a web browser.
- Add custom certificates to the self-signed certificates that a full deployment provides on all machines.
- Upgrade the security protocol and ciphers that are enabled by default using the sas-ssl.conf file.

For more information about setting up the Apache HTTP Server and configuring additional security settings, see *Encryption in SAS Viya: Data in Motion*.

---

**Client Requirements**

**Web Browsers**

End users can access the product user interfaces for SAS Viya applications from a desktop computer, using a supported web browser. Because SAS software is not installed on this machine, the requirements are minimal. UNIX and 64-bit Windows operating systems are supported.

Some SAS Viya user interfaces include some advanced features that require recent versions of popular web browsers. For information about supported web browsers and the corresponding platforms to access SAS user interfaces, see: [https://support.sas.com/en/documentation/third-party-software-reference/viya/35/support-for-web-browsers.html](https://support.sas.com/en/documentation/third-party-software-reference/viya/35/support-for-web-browsers.html).

**Screen Resolution**

The minimum screen resolution for each client machine that will access the SAS Viya user interfaces is 1280 x 1024.
Deployment Tools

Ansible Controller Requirements

A typical Ansible deployment consists of at least one control machine (the Ansible controller) and multiple Ansible managed nodes (the machines where SAS software is installed). In a single-machine deployment, Ansible and all SAS software are installed on the Ansible controller. For more information, see “Install Ansible” on page 20.

In a distributed deployment, the managed nodes use a secure shell (SSH) framework for connections to the Ansible controller. Verify network connectivity between the controller and the managed nodes. Connectivity is also required among all machines in the deployment and from the controller to the SAS yum repositories.

For information about supported Ansible versions and other requirements, see: https://support.sas.com/en/documentation/third-party-software-reference/viya/35/support-for-operating-systems.html#ansible.
Pre-installation Tasks

Prepare for Deployment ................................................................. 10
Installing from a Mirror Repository ........................................... 10
   Create a Mirror Repository .................................................. 10
Enable Required Ports ................................................................. 12
Configure SELinux ................................................................. 14
   Options for Deploying SAS Viya with SELinux ......................... 14
   Disable SELinux and Deploy SAS Viya .................................. 15
   Configure the Environment to Accommodate SELinux ............... 15
   Disable SELinux, Deploy SAS Viya, then Re-enable SELinux ....... 16
Perform Linux Tuning ................................................................. 16
   Set the ulimit Values ......................................................... 16
   Set the Semaphore Values ................................................. 18
   Change the Default Time-outs ............................................ 18
   (SUSE Linux Only) Change the Maximum Number of Operating System Tasks 18
Configure a Proxy Server .......................................................... 19
   Overview ........................................................................... 19
   Using curl ......................................................................... 19
   Using yum ......................................................................... 19
Enable the Yum Cache ................................................................. 20
Install Ansible ............................................................................ 20
   Standard Ansible Installation .............................................. 20
   Streamlined Ansible Installation for Red Hat Enterprise Linux and Equivalent Distributions 20
   Streamlined Ansible Installation for SUSE Linux .................... 21
   Test Your Ansible Installation ............................................. 22
(Optional) Enable Key-Based SSH Authentication .......................... 22
Create a Playbook ....................................................................... 23
   Download the SAS Orchestration CLI ................................... 23
   Create a Playbook with the SAS Orchestration CLI ............... 24
   Store the Playbook ............................................................. 25
Prepare for Deployment

Be sure to complete the tasks that are described in this chapter before you run the playbook.

Installing from a Mirror Repository

A mirror repository is required for all deployments on SUSE Linux. For Red Hat Enterprise Linux, a mirror repository is optional and should be used only if your machine target does not have access to the internet, or if you must always deploy the same version of software (such as for regulatory reasons).

Create a Mirror Repository

Standard Mirror Repository Creation

SAS Mirror Manager is a command-line utility for synchronizing a collection of SAS software repositories. Its primary use is to create and manage mirror repositories for software deployment.

SAS Mirror Manager downloads the software that you ordered and creates a mirror repository. It can create the mirror repository in a specified location, such as a shared NFS mount point or a web server that serves the files by HTTP. The default location for the download is the sas_repos directory that is created in the installation user’s home directory. Make sure that the default location for the download and the destination for the mirror repository have adequate space.

This guide refers to the default location as sas_repos. If you want to specify the mirror destination, use the --path option, followed by the full directory path. In addition, replace instances of sas_repos that are used in this guide with the actual location that you select.

The sas_repos directories are explained as follows:

- The entitlements.json is a list of the repositories to which you are entitled.
- The location_group_declarations.json file and the sasmd directory contain data that is used by the SAS Orchestration CLI to create the order-specific tools for your deployment.
- Any remaining directories are the software repositories, organized by native deployment tools:
  - repos contains yum files for Linux.
  - win contains MSI files for Windows.
  - deb contains APT files for Debian.
To create a mirror repository with SAS Mirror Manager:

1. Download SAS Mirror Manager from the SAS Mirror Manager download site to the machine where you want to create your mirror repository. If you use Internet Explorer to download the Linux or Macintosh version, save the file as a .tgz file instead of a .gz file.
   
   Note: This step requires internet connectivity. If you receive warnings or errors regarding connectivity, check your firewall settings.

2. Uncompress the downloaded file.

3. Uncompress the downloaded file.

4. (Optional) Add the location of SAS Mirror Manager to your PATH environment variable.
   
   Note: This step is not required. However, the example SAS Mirror Manager commands in this section assume that you have added the recommended location to your PATH.
   
   ```bash
   export PATH=/opt/sas/viya/home/bin:$PATH
   ```

5. Run the following basic command to create the mirror repository in the default location:
   
   Note: All the software to which your order entitles you is downloaded if you use the basic command in the previous step. To download software for selected target platforms, skip to the next step.
   
   ```bash
   mirrormgr mirror --deployment-data path-to-SAS_Viya_deployment_data.zip
   ```
   
   By default, the repositories are placed in the `sas_repos` directory in the installation user’s home directory. Use the `--path` option, followed by the full directory location of the mirror destination, to change this location.

6. (Optional) Run the following command to see a list of the platforms that you can select for the download operation:
   
   ```bash
   mirrormgr list remote platforms --deployment-data path-to-SAS_Viya_deployment_data.zip
   ```

7. (Optional) Use the `--platform` option and one of the values that were returned by the `list remote platforms` command to download software only for a selected target platform:
   
   Important: Use `x64-redhat-linux-6` for all supported versions of Red Hat Enterprise Linux and its equivalent, such as Oracle Linux. The `x64-redhat-linux-7` value indicates a package type that is only compatible with a different type of deployment.
   
   ```bash
   mirrormgr mirror --deployment-data path-to-SAS_Viya_deployment_data.zip --path location-of-mirror-repository --platform platform --latest
   ```

8. (Optional) Use the `--latest` option to exclude any obsolete packages from the mirror repository that is being created.
   
   If you use this option, be sure to use it with any subsequent `mirrormgr` commands. For example, to compare the contents of your mirror with the contents of SAS repositories, use the `diff` command with `--latest` if you used this option with the `mirror` command.
9. (Optional) After the initial download is complete, move the file structure to a web server or shared NFS mount point. Internet connectivity is not required for the destination machine.

Depending on your platform, you can use tools like `rsync` and `scp` to move the files. Here is a typical command for `rsync`:

```
rsync -av --progress sas_repos target_machine:/var/www/html/pulp/
```

**Specify a Log Location**

The default location for the logs for SAS Mirror Manager is `user-home-directory/.local/share/mirrormgr/mirrormgr.log`. To specify an alternative log location:

```
mirrormgr mirror --deployment-data path-to-SAS_Viya_deployment_data.zip --path location-of-mirror-repository --log-file location-of-mirror-repository/mirrormgr.log --platform linux-distribution --latest
```

**Enable Required Ports**

The following ports should be available before you begin to deploy your software. The same ports should also be available for any firewalls that are configured on the operating system or the network.

<table>
<thead>
<tr>
<th>Process</th>
<th>Required Port</th>
<th>Must Allow Inbound Traffic From</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event Stream Manager</td>
<td>80 (external)</td>
<td>anywhere</td>
<td></td>
</tr>
<tr>
<td></td>
<td>80 (internal)</td>
<td>anywhere</td>
<td></td>
</tr>
<tr>
<td></td>
<td>443 (external)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTTPD</td>
<td>5430–5439</td>
<td>SAS Viya servers only</td>
<td>For a single server deployment with no failover, ports 5430-5432 must be opened. Additional standby nodes each get the next available port number sequentially up to 5439.</td>
</tr>
<tr>
<td>Process</td>
<td>Required Port</td>
<td>Must Allow Inbound Traffic From</td>
<td>Notes</td>
</tr>
<tr>
<td>---------</td>
<td>--------------</td>
<td>---------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>default SAS Messaging Broker AMQP client access port</td>
<td>5672</td>
<td>SAS Viya servers only</td>
<td></td>
</tr>
<tr>
<td>SAS Configuration Server</td>
<td>8300–8309, 8500</td>
<td>SAS Viya servers only</td>
<td>SAS uses HashiCorp Consul as its configuration server. Ports should be open to both UDP and TCP traffic.</td>
</tr>
<tr>
<td>default SAS Messaging Broker management web console port</td>
<td>15672</td>
<td>SAS Viya servers only</td>
<td></td>
</tr>
<tr>
<td>default SAS Messaging Broker clustering port</td>
<td>25672</td>
<td>SAS Viya servers only</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** To enable the machines in your deployment to communicate, port 80 on the machine where the Apache HTTP server is installed must be reachable by any machine where SAS software is installed. However, in order to secure web access to your SAS software, only port 443 (HTTPS) should be open externally.

The Linux operating system defines a specific series of network service ports as an ephemeral port range. These ports are designed for use as short-lived IP communications and are allocated automatically from within this range. If a required port is within the range of the ephemeral ports for a host, another application can attempt to claim it and cause services to fail to start. Therefore, you must exclude the required ports from the ports that can be allocated from within the ephemeral port range.

1. To determine the active ephemeral port range, run the following command on your host:
   ```bash
   sudo sysctl net.ipv4.ip_local_port_range
   ```
   The results contain two numbers:
   ```plaintext
   net.ipv4.ip_local_port_range = inclusive-lower-limit inclusive-upper-limit
   ```

2. To list any existing reserved ports, run the following command:
   ```bash
   sudo sysctl net.ipv4.ip_local_reserved_ports
   ```
   Here is an example of the results:
   ```plaintext
   net.ipv4.ip_local_reserved_ports = 23, 25, 53
   ```
If no ports are reserved, no ports are listed in the results:

```
net.ipv4.ip_local_reserved_ports =
```

3. After you determine the limits of the ephemeral port range, you must add any required ports that are included in the ephemeral port range to the Linux system reserved ports list. Add ports to the reserved list as comma-separated values or as a range within quotation marks:

```
sudo sysctl -w net.ipv4.ip_local_reserved_ports="ports-or-port-range"
```

Here is an example:

```
sudo sysctl -w net.ipv4.ip_local_reserved_ports="5672,15672,25672"
```

Note: The sysctl command numerically sorts the port numbers regardless of the order that you specify.

4. Add an entry to the `/etc/sysctl.conf` file to make your changes permanent. Here is an example:

```
net.ipv4.ip_local_reserved_ports = 4369,5672,9200,15672,25672
```

---

**Configure SELinux**

If Security-Enhanced Linux (SELinux) is enabled in your environment, it must be disabled or accommodated before you can use Ansible to deploy SAS Viya.

---

**Options for Deploying SAS Viya with SELinux**

If SELinux is required in your environment, it is supported on Red Hat Enterprise Linux if you perform some additional tasks. On SUSE Linux, SELinux is not supported.

To determine the present status of SELinux in your environment:

```
sudo sestatus -v
```

- If you see a message that the command is not enabled, SELinux is not active in your environment. You can use Ansible to deploy SAS Viya.
- If a mode that is not `permissive` is returned, SELinux is enabled. In order to deploy SAS Viya, you must select one of the following options:
  - Disable SELinux and deploy SAS Viya.
  - Configure the environment to accommodate SELinux.
  - Disable SELinux, deploy SAS Viya, and then re-enable SELinux after the deployment has completed.
Enable SELinux and Deploy SAS Viya

If SELinux is active in your environment, one option is to configure permissive mode for SELinux on all the target machines in your deployment.

Note: Permissive mode effectively disables SELinux.

To change the mode value to permissive on all target machines in your deployment:

```
sudo setenforce 0
```
```
sudo sed -i.bak -e 's/SELINUX=enforcing/SELINUX=permissive/g' /etc/selinux/config
```

Configure the Environment to Accommodate SELinux

To deploy SAS Viya with SELinux enabled on all target machine in your deployment, perform all the following tasks:

1. Disable the pre-installation check that determines whether SELinux is active. For more information, see.

2. Configure SELinux to enable the Apache HTTP Server. By default, SELinux does not allow the Apache httpd component to access the network.

   Run the following command on any machines that are deployTargets for the [httpproxy] host group in the inventory.ini file:

   ```
sudo setsebool -P httpd_can_network_connect 1
```

3. Make sure that the SELinux Policy deny_unknown status is set to allowed. Perform the following steps:

   a. Run the following command to determine the current SELinux settings:

      ```
sudo sestatus -v
```

   b. Check the value of Policy deny_unknown status in the output. If the value is not allowed, you must change the policy setting.

   c. As root, edit the /etc/selinux/semange.conf file.

   d. Add the following line:

      ```
handle-unknown=allow
```

   e. As root, run the following command to rebuild and reload the policy:

      ```
semodule -B
```
Disable SELinux, Deploy SAS Viya, then Re-enable SELinux

A final option is to disable SELinux during the deployment and then re-enable it as soon as the deployment has completed. If you select this option, perform tasks 2 and 3, as described in “Configure the Environment to Accommodate SELinux”.

Perform Linux Tuning

This section describes tuning that should be performed on your Linux machines before you deploy your software.

Set the ulimit Values

Overview

The Linux operating system provides mechanisms that enable you to set the maximum limit for the amount of resources that a process can consume. Here are some of the resource types:

- open file descriptors
- stack size
- processes available to a user ID

Each resource type with limits is stored in the appropriate file on each machine in your deployment.

Here is the format of the `/etc/security/limits.conf` file for setting the maximum number of open file descriptors:

```
*     -     nofile    value
```

The asterisk (*) indicates all user accounts.

For a single user account, * can be replaced with the user ID for that account. Here is an example:

```
account-name    -     nofile    value
```

This line is duplicated in the file for each user ID.

For a group, * can be replaced with the at symbol (@) followed by the group name. Here is an example:

```
@group-name    -     nofile    value
```
Set the Maximum Number of Open File Descriptors and Stack Size

For each machine in your deployment:

1 Open the `/etc/security/limits.conf` file.

2 Set the limit for open file descriptors as follows:
   - If PostgreSQL will be deployed on the machine, set the limit (using the nofile item) to 150000 for the sas user account.
     ```
     sas     -     nofile     150000
     ```
   - For all other machines in the deployment, set the limit for the sas user to at least 48000.
     ```
     *     -     nofile     48000
     ```
   
   Note: If you are performing a single-machine deployment, use the highest limit (described in step 2) for all users.
   ```
   *     -     nofile     150000
   ```

3 For machines on which PostgreSQL will be deployed, set the limit for the stack size (using the stack item) to 10240 for the sas user.
   ```
   sas     -     stack     10240
   ```
   For machines that will not have PostgreSQL deployed on them, do not set a limit for the stack size.

4 Save and close the `/etc/security/limits.conf` file.

Set the Maximum Number of Processes Available

For each machine in your deployment:

1 Open the appropriate file. For Red Hat Enterprise Linux 6.7 or an equivalent distribution, open `/etc/security/limits.d/90-nproc.conf`. For Red Hat Enterprise Linux 7.1 and greater or an equivalent distribution, open `/etc/security/limits.d/20-nproc.conf`. For SUSE Linux, open `/etc/security/limits.conf`.

2 Set the limit for the number of processes as follows:
   - If PostgreSQL will be deployed on the machine, set the limit (using the nproc item) to 100000 for the sas user account.
     ```
     sas     -     nproc     100000
     ```
   - For all other machines in the deployment, set the sas user to at least 65536.
     ```
     *     -     nproc     65536
     ```
   
   Note: If you are performing a single-machine deployment, use the highest limit (described in step 2) for all users.
   ```
   *     -     nproc     100000
   ```

3 Save and close the `*-nproc.conf` file.
Set the Semaphore Values

For each machine on which PostgreSQL will be deployed.

1. Open the `/etc/sysctl.conf` file.

2. Add the following lines or modify existing values as follows:
   
   ```
   kernel.sem=512 32000 256 1024
   net.core.somaxconn=2048
   ```

3. Save and close the `/etc/sysctl.conf` file.

4. Refresh the revised settings from the `/etc/sysctl.conf` file:
   ```
   sudo sysctl -p
   ```

Change the Default Time-outs

Note: The information in this section applies only to systems running Red Hat Enterprise Linux 7.1 or later or equivalent distributions. If you are using a Linux distribution earlier than Red Hat Enterprise Linux 7.1, you should skip this section.

To change the default time-out values:

1. Open the `/etc/systemd/system.conf` file.

2. Find the two variables that control time-outs: DefaultTimeoutStartSec and DefaultTimeoutStopSec.

3. If the lines that contain these variables are not already uncommented, uncomment each line by removing the number sign (#).

4. Assign both the DefaultTimeoutStartSec and DefaultTimeoutStopSec variables a value of 1800s.
   ```
   DefaultTimeoutStartSec=1800s
   DefaultTimeoutStopSec=1800s
   ```

5. Save and close the `/etc/systemd/system.conf` file.

(SUSE Linux Only) Change the Maximum Number of Operating System Tasks

If you are deploying on SUSE Linux, run the following commands to change the maximum number of operating system (OS) tasks that each user can run concurrently.

Note: Run these commands as a root or sudoer user.

```bash
sudo sed -i 's#.*UserTasks.*#UserTasksMax=50000#g' /etc/systemd/logind.conf
sudo systemctl restart systemd-logind
```
Configure a Proxy Server

Overview

The SAS Viya deployment process uses both curl and yum to download RPM packages from SAS repositories. If your organization uses a forward HTTP proxy server, both curl and yum on each target deployment machine must be configured for forward proxy servers.

Refer to the Linux man pages for yum.conf and curl for more information about proxy settings.

Using curl

Curl uses the https_proxy and http_proxy environment variables to send requests to proxy servers. You can export these variables in a new shell profile script such as /etc/profile.d/httpproxy.sh. Here is an example of the /etc/profile.d/httpproxy.sh script:

```
export https_proxy=http://user-name:password@internet-proxy-server-FQDN:8080/
export http_proxy=http://user-name:password@internet-proxy-server-FQDN:8080/
```

In addition, ensure that HTTP requests between machines in the deployment are not routed through the proxy server during deployment by adding the IP addresses, host names, or domains for the SAS Viya machines to the no_proxy variable in your profile.d script. For example, if the SAS Viya machines are using the IP addresses, 10.255.47.131 and 10.255.47.132, and the host names, machine1.example.com and machine2.example.com, you can configure no_proxy as follows:

```
export no_proxy="localhost,127.0.0.1,.example.com,10.255.47.131,10.255.47.132"
```

If the profile script is properly configured, these environment variables are set at login for all users. Curl requests for HTTP or HTTPS resources should use the connection information from these variables.

Using yum

Forward proxy server settings for yum can be configured in /etc/yum.conf. Here is an example of the /etc/yum.conf script:

```
proxy=internet-proxy-server-FQDN:8080/
proxy_username=user-name
proxy_password=password
```
Enable the Yum Cache

**Note:** SUSE Linux does not use yum as a deployment tool. If you are using SUSE Linux or installing from a local mirror repository, skip this section.

By default, yum deletes downloaded files after a successful operation when they are no longer needed, minimizing the amount of storage space that yum uses. However, you can enable caching so that the files that yum downloads remain in cache directories. By using cached data, you can perform certain operations without a network connection.

In order to enable caching, add the following text to the `[main]` section of `/etc/yum.conf`.

```
keepcache = 1
```

This task should be performed on each machine in the deployment.

Install Ansible

Ansible is third-party software that provides automation and flexibility for deploying software to multiple machines. You must install a supported version of Ansible.

Standard Ansible Installation

The Ansible installation process is documented at [http://docs.ansible.com/ansible/latest/intro_installation.html](http://docs.ansible.com/ansible/latest/intro_installation.html). You should always follow the Ansible documentation and choose the installation method that works best for your IT environment.

Not all versions of Ansible that are available for installation are supported by SAS Viya. For a list of supported Ansible versions, see: [https://support.sas.com/en/documentation/third-party-software-reference/viya/35/support-for-operating-systems.html#ansible](https://support.sas.com/en/documentation/third-party-software-reference/viya/35/support-for-operating-systems.html#ansible). On that same page, SAS provides a list of supported versions of Python. Python support is determined by the release of Ansible that you install.

Streamlined Ansible Installation for Red Hat Enterprise Linux and Equivalent Distributions

**Note:** Even though you are advised to follow the instructions in the Ansible documentation, streamlined installation instructions are provided here as a convenience. Before performing these instructions, ensure that they are appropriate for your site and that they comply with the IT policies in your organization.
These steps assume that you have sudo access to the machine where you are installing Ansible.

1 Run the following commands to attach the EPEL repository to your server. You can copy and paste this entire block of text for convenience.

```bash
## find out which release (6 or 7)
if   grep -q -i "release 6" /etc/redhat-release ; then
    majversion=6
elif grep -q -i "release 7" /etc/redhat-release ; then
    majversion=7
else
    echo "Apparently, running neither release 6.x nor 7.x "
fi
## Attach EPEL
# Display the available repositories
sudo yum repolist
```

2 To Install Python PIP and related packages:

```bash
sudo yum install -y python python-setuptools python-devel openssl-devel
```

3 Because EPEL will no longer be required, you can remove it with the following command:

```bash
sudo yum remove -y epel-release
```

4 Upgrade PIP and setuptools using one of the following methods, based on the version of Python you are running.

Specific versions of Python modules are required. Here are some examples:

```bash
sudo pip install --upgrade pip==9.0.3
sudo pip install pycparser==2.14
sudo pip install idna==2.7
```

For Python 2.7 (and later within 2.7.x):

```bash
sudo pip install --upgrade pip setuptools
```

5 To install a specific version of Ansible through PIP:

```bash
sudo pip install ansible==2.7.2
```

---

Streamlined Ansible Installation for SUSE Linux

**Note:** Even though you are advised to follow the instructions in the Ansible documentation, streamlined installation instructions are provided here as a convenience. Before performing these instructions, ensure that they are appropriate for your site and that they comply with the IT policies in your organization.

These steps assume that you have sudo access to the machine where you are installing Ansible.

1 To install Python’s setup tools:

```bash
sudo zypper install python-setuptools
```

2 To Install Python PIP:
sudo easy_install pip

3 To install a specific version of Ansible through PIP:
   sudo pip install ansible==2.7.2

Test Your Ansible Installation

1 To test the Ansible version:
   ansible --version

Here is an example of successful output:

   ansible 2.7.2
   config file =
   configured module search path = Default w/o overrides
   python version = 2.7.15 (default, May 14 2019, 07:55:04)
   [GCC 4.8.5 20150623 (Red Hat 4.8.5-14)]

2 To perform a basic ping test:
   ansible localhost -m ping

Here is an example of successful output:

   [WARNING]: Host file not found: /etc/ansible/hosts
   [WARNING]: provided hosts list is empty, only localhost is available
   localhost | SUCCESS => {
             "changed": false,
             "ping": "pong"
   }

(Optional) Enable Key-Based SSH Authentication

Note: Key-based SSH authentication is optional, but it is recommended.

In order to run Ansible tasks on multiple hosts without being prompted for a password, you can create an SSH key pair and distribute the public key to the machines where SAS software will be installed. Performing this task provides a secure authentication mechanism for SSH logins and avoids the need for SSH password options when performing Ansible tasks.

Here is an example of setting up an SSH key pair. However, there are many methods for creating and propagating SSH keys.

Note: These steps assume that the PasswordAuthentication keyword has been enabled in the SSH daemon configuration file. It is also assumed that the user has a password that can be used for ssh-copy-id authentication.
1. Create an SSH key pair without a passphrase. The following example specifies the RSA key type. However, you can specify any key type that is supported by your SSH installation. Refer to the ssh-keygen man page for more information.

   ```
   ssh-keygen -t rsa -N "" -f ~/.ssh/id_rsa
   ```

2. Copy the public key to each target host. Here is an example:

   ```
   ssh-copy-id target0.example.com
   ssh-copy-id target1.example.com
   ```

   If the machine where Ansible is installed is also a target host for installing SAS software, run `ssh-copy-id` against the Ansible host.

3. Verify that you can authenticate to all target hosts without being prompted for a password.

   If you enable SELinux by setting it to enforcing after the deployment has completed, you must restore the context of any SSH keys that were generated while SELinux was running in permissive mode. Here is an example of the command:

   ```
   restorecon -R -v /path-to-keys/.ssh
   ```

   For `path-to-keys`, substitute the location of your SSH key files.

---

### Create a Playbook

If you are installing on SUSE Linux, be sure to complete the steps in “Create a Mirror Repository” on page 10 before you create a playbook.

The SAS Orchestration Command Line Interface (CLI) uses the order information that was included in your Software Order Email (SOE) to create a playbook for deploying your SAS software. Before you use the SAS Orchestration CLI, ensure that the SAS_Viya_deployment_data.zip file attachment from your SOE is copied to a directory on a machine that runs the Linux, Macintosh, or Windows operating system.

---

### Download the SAS Orchestration CLI

The SAS Orchestration CLI can be run on Linux or Windows and it requires the Java Runtime Environment 1.8.x. It also requires access to the internet.

1. The SOE indicated that you should save the SAS_Viya_deployment_data.zip file attachment. If you have not already done so, save that file now.

2. Go to [SAS Viya Install Center](#), and download the SAS Orchestration CLI for the operating system where you stored the ZIP file.

   The SOE recommended that you save the ZIP file to a machine that runs Linux, which is where you install the SAS software that you purchased. However, you can also store it on a machine that runs Macintosh or Windows. If you use Internet Explorer to download the Linux or Macintosh version, save the file as a .tgz file instead of a .gz file.

   **Note:** This step requires internet connectivity.
Uncompress the TGZ file (Linux and Macintosh) or ZIP file (Windows) in the same location where you downloaded it. The result is a file named sas-orchestration on Linux or Macintosh or a file named sas-orchestration.exe on Windows.

Create a Playbook with the SAS Orchestration CLI

Basic Command

To create a playbook, use the command that is appropriate for the operating system where the SAS Orchestration CLI is located.

Note: The following commands are organized by the operating system where the SAS Orchestration CLI runs, rather than by the operating system where your SAS Viya software is deployed. After you create the playbook, you can move it to the machine where you deploy your software. Enter each command on a single line. Multiple lines are used here to improve readability.

Linux or Macintosh

```
./sas-orchestration build --input location-of-ZIP-file-including-file-name --platform deployment-platform-tag --architecture deployment-architecture-tag
```

Windows

```
\sas-orchestration.exe build --input location-of-ZIP-file-including-file-name --platform deployment-platform-tag --architecture deployment-architecture-tag
```

For `deployment-platform-tag`, specify the target operating system, one of the following:

- `redhat` for Red Hat Enterprise Linux or an equivalent distribution, such as Oracle Linux
- `suse` for SUSE Linux Enterprise Server

For `deployment-architecture-tag`, specify the target chip for the deployment: `x64` for 64-bit chips.

Using the SAS Orchestration CLI creates a new file named `SAS_Viya_playbook.tgz`.

Options

Use a Proxy Server

If you use an unauthenticated proxy to reach the internet, you must add the following option to the run command in order to make an outgoing connection:

```
--java-option "-Dhttps.proxyHost=proxy-server-IP-address-or-host-name"
```

```
--java-option "-Dhttps.proxyUser=user-name"
```

```
--java-option "-Dhttps.proxyPassword=user-password"
```

In addition, if the proxy server is not using the default proxy port of 80, you must also add the following option:
For normal usage, when communicating with SAS, the options must start with Dhttps. If you have set up a mirror behind an HTTP server without TLS, the options must start with Dhttp.

The Java options should not be combined into a single option. Here is an example of using the options on a Linux machine:

```
./sas-orchestration --java-option "-Dhttps.proxyHost=my.proxy.com --java-option "-Dhttps.proxyPort=1111" build --input /tmp/SAS_Viya_deployment_data.zip
```

The --java-option tags must come before the build command.

Use a Mirror Repository

If you created a mirror repository with SAS Mirror Manager, you must include its location with the --repository-warehouse option.

```
./sas-orchestration build --input /sas/install/ SAS_Viya_deployment_data.zip --platform redhat --repository-warehouse "URL-to-mirror-repository-content"
```

Note: The repository warehouse URL must be available to all hosts that will participate in the deployment because the hosts are going to use that address to retrieve packages from the repositories. For example, if the repository warehouse is file-based, that location should be shared across hosts and should be shared at the same path on each of those hosts.

For more information about SAS Mirror Manager, see "Create a Mirror Repository" on page 10.

Help with the Options

The SAS Orchestration CLI includes several options. To learn about all the options for the SAS Orchestration CLI, use the appropriate command:

**Linux or Macintosh**

```
./sas-orchestration build --help
```

**Windows**

```
\sas-orchestration.exe build --help
```

Store the Playbook

1. If necessary, move the SAS_Viya_playbook.tgz file to a directory on your Ansible controller that can be read by other users. The recommended location is /sas/install.

2. In the same directory where you have saved the playbook, uncompress it.

```
tar xf SAS_Viya_playbook.tgz
```

In addition, SAS recommends that you create a directory on each machine in your deployment for storing files that are used to deploy and maintain your software. The best practice is to use the same directory location on each machine. SAS recommends using /sas/install. This guide assumes that you will use /sas/
install. However, if you do not use it, replace those instances in this guide with the actual location that you select.
Overview

This chapter describes the initial deployment of your SAS Event Stream Manager software. For information about modifying an existing deployment with updated software or adding new software to an existing deployment, see Chapter 8, “Managing Your Software,” on page 45.

When you order SAS software, SAS sends a Software Order Email (SOE) to your business or organization. Your SOE includes information about the software order, including several file attachments and instructions for generating a deployment playbook using the SAS Orchestration CLI.

If you have not already done so, be sure to uncompress the file that is attached to your SOE, as instructed in the email text.

Edit the Playbook

About the Inventory File

Ansible uses an inventory file to define the machines to be included in a deployment and the software to be installed on them. The file sas_viya_playbook/
inventory.ini is used as the inventory file. If you used the recommended location for uncompressing your playbook, the file is located at /sas/install/sas_viya_playbook/inventory.ini.

However, if you do not want to manually complete the default inventory.ini file, you can copy an existing template from the sas_viya_playbook/samples subdirectory instead. This directory contains templates for different types of deployments, including a single-machine deployment, which is described later in this chapter. Copy the template that you want to use, rename it inventory.ini, and place it in the sas_viya_playbook directory. It replaces the existing inventory.ini file.

The first line of the inventory file is a deployment target reference. It defines the machine on which the SAS Viya software is being deployed and the user account that controls Ansible. Make sure that the user account has root or sudo permissions. In a typical SAS Event Stream Processing deployment, Ansible is used locally (on the same machine where you are deploying SAS software), and the inventory.ini file should be used without modification.

Note: If you are instead using Ansible remotely from a separate machine, modify the first line in the inventory file to include the location of the machine where SAS software is being deployed. Use the following format:

```
deployTarget ansible_ssh_host=host1.example.com
```

Each inventory file consists of two parts:

**deployment target definition**
A specification of each machine where SAS Viya software is deployed.

**host group assignment list**
A mapping of the installable groups of software and the machines on which they will be deployed. SAS Viya software is deployed as host groups, which are identified by square brackets ([ ]) in the inventory file. Each host group is preceded by comments that describe the purpose of the software in the host group. The comments also contain any requirements, constraints, and recommendations relating to host groups.

Specify the machines where a host group is deployed by listing them under the host group name. A machine can have more than one host group deployed on it.

Here is an example of a host group assignment list:

```
# The CommandLine host group contains command line interfaces for remote interaction with services.
[CommandLine]
deployTarget
deployTarget2
```

Note: Inventory files are generated for a specific software order. Do not copy files from one playbook and attempt to use them with another playbook.

---

**Modify the vars.yml File**

As its name suggests, the vars.yml file contains deployment variables that enable you to customize your deployment to meet your requirements. Note that all entries in the vars.yml file are case-sensitive.
Open the file from the `sas_viya_playbook` directory, and use your preferred text editor to modify it. If you used the recommended location for uncompressing your playbook, the file is located at `/sas/install/sas_viya_playbook/vars.yml`.

### Set the Deployment Label

The `DEPLOYMENT_LABEL` is a unique name used to identify the deployment. A default value for `DEPLOYMENT_LABEL` is set by the playbook.

If you want to use a customized `DEPLOYMENT_LABEL`, replace the default entry with another name, within double quotation marks, that is appropriate for your deployment. The name can contain only lowercase alphabetic characters, numbers, and hyphens. Nonalphanumeric characters, including a space, are not allowed.

Here is an example of a valid name:

```
DEPLOYMENT_LABEL: "esm-2019"
```

### Set the Pre-deployment Validation Parameters

The setting of the `VERIFY_DEPLOYMENT` variable determines the extent of the pre-deployment validation that the playbook performs. If the variable is set to true (the default), all of the following actions take place. If the variable is set to false, only the Ansible version check is performed.

### Check the Ansible Version

The playbook checks the installed Ansible version to determine whether it is at least the minimum supported version. If not, the playbook stops with a message.

**Note:** For information about supported Ansible versions, see [https://support.sas.com/en/documentation/third-party-software-reference/viya/35/support-for-operating-systems.html#ansible](https://support.sas.com/en/documentation/third-party-software-reference/viya/35/support-for-operating-systems.html#ansible).

### Verify Machine Properties

The playbook checks each machine in the deployment to ensure that the necessary conditions for deployment are met. If any of these conditions is not met, a warning is given and the playbook stops the deployment.

1. Use your preferred text editor to open the `vars.yml` file. If you uncompressed your playbook in the recommended location, the file is in `/sas/install/sas_viya_playbook/`.
2. Verify that the `DEPLOYMENT_LABEL` variable has content and contains only lowercase alphabetic characters, numbers, and hyphens.
3. Verify that the target machine’s fully qualified domain name contains less than or equal to 64 characters.

### Verify System Requirements

The playbook ensures that some system requirements are met. If any of the following requirement checks fail, a warning is provided and the playbook stops execution.

1. Verify that SELinux mode for each machine is disabled. The value should be set to `permissive`, which disables SELinux.
This check is performed by default. You can deploy SAS Viya in an environment where SELinux is enabled. However, some additional configuration is required. For more information about the requirements to run with SELinux enabled, see “Configure SELinux” on page 14.

You can temporarily set SELinux to permissive until the deployment has completed. However, if you instead deploy with SELinux enabled (that is, set to enforcing), you must add a key-value pair that disables playbook checks for SELinux. For more information, see “(Optional) Disable SELinux Verification” on page 31.

2 For each machine, verify the nofile and nproc settings for the install user. For more information, see “Perform Linux Tuning” on page 16.

Specify the Path to Certificates

By default, when SAS Event Stream Manager is deployed, it installs Apache httpd with a self-signed certificate for use across the deployment. If you want to accept the default, you should skip this section. If, however, you already have httpd set up and configured, you must provide a value for the HTTPD_CERT_PATH variable as described here.

The SSLCertificateChainFile is a variable set in the security configuration file for Apache httpd at /etc/httpd/conf.d/ssl.conf. It is a location on your system containing certificate information. SAS recommends that the file at the location that SSLCertificateChainFile represents contain the root certificate authority (CA) and all intermediate certificates in the chain.

To set HTTPD_CERT_PATH:

1 Open the vars.yml file.

2 Set the value of HTTPD_CERT_PATH based on the following conditions. Ensure that any value you use is enclosed in single quotation marks (’).

   - If your SSLCertificateChainFile contains the root certificate authority (CA) and all intermediate certificates, remove the existing value for HTTPD_CERT_PATH. Ensure that all browsers and clients have the root CA in their truststore.

     Here is an example of the modified variable:

     HTTPD_CERT_PATH:

   - If your SSLCertificateChainFile contains the intermediate links but not the root CA, HTTPD_CERT_PATH should be the path to the file on the machine in the [httpproxy] host group in the inventory file that contains the root CA.

     Here is an example of the HTTPD_CERT_PATH variable with a value:

     HTTPD_CERT_PATH: '/etc/pki/tls/certs/my-ca-chain.crt'

   - If your SSLCertificateChainFile contains no certificates and no root CA, HTTPD_CERT_PATH should be the path to the file on the machine in the [httpproxy] host group in the inventory file that contains the intermediate certificates and the root CA. Ensure that all the intermediate certificates are in the truststore of all browsers and clients.

     Here is an example of the HTTPD_CERT_PATH variable with a value:

     HTTPD_CERT_PATH: '/etc/pki/tls/certs/my-ca-chain.crt'

   - Note: The default value for HTTPD_CERT_PATH in the vars.yml file is the most likely location for the necessary file for Red Hat Enterprise Linux and equivalent distributions. If that file is in the default location, you do not need
to make any changes. The default location for SUSE Linux is `/etc/apache2/ssl.crt/localhost.crt`.

3 Save and close the vars.yml file.

**Change the Repository Warehouse**

When you generate the playbook with the SAS Orchestration CLI, the `REPOSITORY_WAREHOUSE` variable in the vars.yml file is set to the default repository warehouse or to the repository warehouse that was specified in the command-line option. If you are using a mirror repository, the value for `REPOSITORY_WAREHOUSE` should be the location of that mirror. If the target deployment systems use a different address to the mirror repository or if the mirror repository is moved after the initial deployment, you should change the mirror location by revising the `REPOSITORY_WAREHOUSE` value.

**Note:** If you are using a Red Hat Satellite Server, use a value of `none` to prevent the deployment from adding more repositories to the server.

```yaml
REPOSITORY_WAREHOUSE: "URL-to-mirror-repository-content"
```

**(Optional) Specify JRE**

The Java Runtime Environment (JRE) must be installed on the target machine to enable SAS Event Stream Manager. By default, the playbook attempts to install a recent version of OpenJDK and to set the path in a system configuration file. You can instead supply the path to an existing JRE before you run the playbook. To use a preinstalled version of the JRE:

1 Use your preferred text editor to open the vars.yml file.

2 Set the value of `sas_install_java` to false. For example:

```yaml
sas_install_java: false
```

3 Add the file path to the JRE as the value of `sasenv_java_home`. Be sure to include “jre” in the file path. For example:

```yaml
sasenv_java_home: /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.101-3.b13.el6_8.x86_64/jre
```

4 Save and close the vars.yml file.


**(Optional) Disable SELinux Verification**

If SELinux is active in your environment, and you want to deploy SAS Viya with SELinux enabled, add a key-value pair.

1 Open the vars.yml file if it is not already open.

2 Add the following statement at the bottom of the file:

```yaml
VERIFY_SELINUX: false
```

3 Save and close the vars.yml file.

For more information about SELinux, see “Configure SELinux” on page 14.
Configure LDAP Settings

The sitedefault.yml file, which is in the /roles/consul/files directory in the playbook, is used to configure authentication for SAS Event Stream Manager. After the initial deployment, you cannot simply modify sitedefault.yml to change an existing value and deploy the software again. You can modify sitedefault.yml only to set property values that have not already been set.

For more information about using the sitedefault.yml file, see Configuration Properties: Concepts in SAS Viya Administration.

Take these steps to enable the playbook to configure the LDAP server so that SAS Logon Manager can provide access to SAS Event Stream Manager:

1. If you have not already copied and renamed the sitedefault.yml file, locate the sitedefault_sample.yml file on the Ansible controller machine. If you used the recommended location for uncompressing your playbook, the file is located at /sas/install/sas_viya_playbook/roles/consul/files/sitedefault_sample.yml. Make a copy of sitedefault_sample.yml in the same folder, and name the copy sitedefault.yml.

2. Use your preferred text editor to open sitedefault.yml.

3. Add values that are valid for your site, and save the file.

When you run your Ansible playbook using the site.yml option, the updated sitedefault.yml file is used automatically.

Install the Software

Deploy the Software

Assessment Test

Before you deploy the software, SAS recommends that you run the following command to assess the readiness of your system for deployment.

    ansible-playbook system-assessment.yml

Add an option based on the password requirements for the user ID that performs the command:

<table>
<thead>
<tr>
<th>Password Requirements</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Does not require passwords</td>
<td>ansible-playbook site.yml</td>
</tr>
</tbody>
</table>
Password Requirements | Command
---|---
Requires a sudo password only | `ansible-playbook site.yml --ask-become-pass`

Requires an SSH password only | `ansible-playbook site.yml --ask-pass`

Requires both a sudo and an SSH password | `ansible-playbook site.yml --ask-pass --ask-become-pass`

If you receive an unexpected error, run the following command to ensure that you are using a supported version of Ansible.

`ansible-playbook --version`

**Note:** For information about supported Ansible versions, see "Ansible Controller Requirements" on page 8.

If you are using a supported version of Ansible and still receive errors from the system assessment, fix those errors before you run the deployment command.

**Deployment Command**

Ensure that you are at the top level of the playbook in the `sas_viya_playbook` directory.

Use the appropriate command to run the playbook, according to the password requirements for the user ID that performs the deployment:

You deploy the software by running the playbook. Here is the basic syntax for the command to run the playbook and deploy the software:

```bash
ansible-playbook site.yml [ option ]
```

The command that you select is determined by your deployment and password requirements. You can select an option to specify the interface to the software to be installed in your environment. To specify whether you want to perform only an installation or configuration, see “Options” on page 33.

In addition, SAS recommends adding a `-vvv` option to enable verbose logging. This option will assist SAS Technical Support in diagnosing any issues you might need to contact them about.

**Options**

To install, but not configure the software, use the basic command, but replace `site.yml` with `install-only.yml`. Here is an example:

```bash
ansible-playbook install-only.yml --ask-pass --ask-become-pass
```

To configure software that has been installed only, use the basic command:

```bash
ansible-playbook site.yml [ option ]
```

**Run from a Directory Other Than the Default**

The playbook runs the commands from the top-level `sas_viya_playbook` directory by default. If you want to run the playbook from another directory, modify the `ansible.cfg` configuration file with the appropriate SAS Viya configuration options.
Refer to the Ansible documentation to find the appropriate ansible.cfg file and add those options.

**Successful Playbook Execution**

Here is an example of the output from a successful playbook execution:

```
PLAY RECAP ********************************************************************
deployTarget               : ok=81   changed=65   unreachable=0    failed=0
```

The most important indicator of success from this message is `failed=0`. If the deployment is successful, the software is deployed to the `/opt/sas` directory.

**Retry a Failed Deployment**

If your deployment fails, and you are able to respond to the error message and can recover from the error, you must restart the deployment using the appropriate deployment commands described in “Deployment Command” on page 33 and any appropriate options.

**Deployment Logs**

Logs for Ansible deployments are stored in `sas_viya_playbook/deployment.log`. If you used the recommended location for uncompressing your playbook, the file is located at `/sas/install/sas_viya_playbook/deployment.log`.

To view the logs from the yum installation commands that are used in your deployment, run the following commands:

```
sudo yum history
sudo less /var/log/yum.log
```
Post-installation Tasks

Complete SAS Event Stream Manager Setup

Take a few steps after the installation has completed to prepare the environment.

Configure the ESP Server for SAS Event Stream Manager

In order to manage SAS Event Stream Processing instances with SAS Event Stream Manager, you must locate and define ESP servers that are running in your environment. SAS recommends that you start your ESP servers with some additional instructions that enable secure, persistent sockets between SAS Event Stream Manager and ESP servers. SAS Event Stream Manager can then locate and manage ESP servers automatically. Use of SAS Event Stream Manager is optional.

To start an ESP server with a connection to SAS Event Stream Manager:

1. Provide SAS Logon Manager with a client ID and client secret for SAS Event Stream Manager. First, obtain the value of the SAS Configuration Server (Consul) token for your environment:

   **Note:** Specify the command on a single line. Multiple lines are used here for improved readability.

   ```bash
   sudo cat /opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/default/client.token
   ```

2. Run a curl command to request a registration token for a new client. In this example, the client is named app:

   ```bash
   curl -X POST "http://localhost/SASLogon/oauth/clients/consul?callback=false&serviceId=app" -H "X-Consul-Token: X-Consul-Token-Value"
   ```

   For `X-Consul-Token-value`, substitute the value for the Consul token, which you obtained from the previous step.
Note: Specify the command on a single line. This request must pass a callback=false query string parameter and authenticate directly by passing a Consul token. If the Consul token that you specified in the command is valid, SAS Logon Manager returns the OAuth access token for registration in the response.

3 Use the registration token to register the client ID. This step establishes the ESP server as a new client of SAS Logon Manager. Run the following curl command:

```bash
    "client_id": "client-id",
    "client_secret": "client-secret",
    "scope": ["openid", "*"]
}
''
```

Note: You can find more information about the required steps to configure a new client for SAS Logon Manager in Obtain an Access Token Using Password Credentials in SAS Viya Administration: Authentication.

4 Create an XML file with filename esm.xml. Make sure that it uses the required syntax.

Here is an example:

```xml
<esm>
  <server name="SAS-Event-Stream-Manager-host">
    <url>http://fully-qualified-host-name</url>
    <port>port-number</port>
    <context-path>context-path-to-SAS-Event-Stream-Manager</context-path>
    <auth>
      <clientId>client-id</clientId>
      <clientSecret>client-secret</clientSecret>
      <user>user-name</user>
      <password>password</password>
    </auth>
  </server>
</esm>
```

a For SAS-Event-Stream-Manager-host, substitute the host name of the machine where SAS Event Stream Manager is running.

b For fully-qualified-host-name, substitute the fully qualified domain name of the machine where the ESP server is running.

c (Optional) For port-number, substitute the port where SAS Event Stream Manager, running without a proxy server, is listening.

d (Optional) For context-path-to-SAS-Event-Stream-Manager, substitute the context path to your instance of SAS Event Stream Manager that is deployed without SAS Viya services. If nothing is specified, the default context path (/SASEventStreamManager) is used.

e For client-id, substitute the client ID that you provided to SAS Logon Manager for the SAS Event Stream Manager instance.
For client-secret, substitute the client secret that you provided to SAS Logon Manager for SAS Event Stream Manager.

For user-name, substitute a user name for an LDAP user account that is valid for use with SAS Logon Manager.

For password, substitute the password that corresponds to the user account that you specified.

Repeat the <server></server> section of the file as many times as required to accommodate all SAS Event Stream Manager servers.

5 Save the file in a network-accessible directory.

6 Change directories:
   
   cd /opt/sas/viya/home/SASEventStreamProcessingEngine/version/bin
   
   For version, specify the version of the SAS Event Stream Processing software.

7 Start the ESP server. If you have an esm.xml configuration file, use this command:

   ./dfesp_xml_server -esm file://full-path-to-file/esm.xml

   The -esm file://esm.xml argument instructs the ESP server to read the contents of the esm.xml file. Other values that provide server start-up instructions are defined in the esp-properties.yml configuration file. For more information, see Server Configuration Properties.

   When it is started with the optional -esm file://esm.xml argument, the ESP server automatically registers with SAS Event Stream Manager, which can then manage it. The esm.xml file instructs the ESP server where to locate SAS Event Stream Manager. The ESP server registers itself with SAS Logon Manager as a new client with a new secret. SAS Logon Manager can then provide a token that enables the ESP server to set up a persistent web socket for secure communications with SAS Event Stream Manager.

   You can also configure the ESP server to start and run as a system daemon. You can then use init.d scripts to start and stop it. For more information, see Setting Up the ESP Server as a Daemon.

---

Log On to SAS Event Stream Manager

SAS Event Stream Manager uses SAS Logon Manager for logon functionality. SAS Logon Manager uses LDAP for user authentication.

1 Open the following URL:

   http://host:port/SASEventStreamManager

   The host is the system on which SAS Event Stream Manager is installed. The port is the port number used by the system that hosts SAS Event Stream Manager. The default port is 80.

   The Sign In to SAS window is displayed.

2 Enter your user ID and password, and click Sign In.

   If you are a member of the SASAdministrators group, the Assumable Groups window is displayed. Group membership is not required.
Successful logon to the SAS Event Stream Manager user interface indicates that the software has been installed correctly. To validate that services have been installed and started successfully, see “Verify SAS Event Stream Manager Status” on page 39.

(Optional) Enable Kerberos Connections

Note: If you have not purchased SAS Event Stream Manager or Kerberos is not enabled in your environment, skip this section.

When Kerberos is configured for the machine where the ESP server is running, additional setup is required. When Kerberos is used for authentication, you must edit a configuration file to enable SAS Event Stream Manager to connect to the ESP server.

1 Contact a system administrator who maintains Kerberos configuration at your organization. Request the location of the keytab file and the user principal name for the machine where the ESP server is running.

2 Using a user account with sudoers privileges, log on to the machine where you have installed SAS Event Stream Manager.

3 Use your preferred text editor to modify the following file, or create it if it does not already exist:

   /opt/sas/viya/config/etc/sysconfig/esm-service.conf

4 Add the following lines to set two environment variables, substituting the values that you obtained from your system administrator. Here is an example:

   export ESM_KEYTAB_LOCATION=/etc/keytab-file-name
   export ESM_USER_PRINCIPAL=user-name/fully-qualified-host-name@KERBEROS-REALM

   For keytab-file-name, substitute the name of the keytab file such as krb5.keytab.

   For user-name, substitute the primary portion of the user principal name, which is typically a user name.

   For fully-qualified-host-name, substitute the fully qualified host name of the machine where the ESP server is running. An example is myhost.machine.domain.com.

   For KERBEROS-REALM, substitute the name of the Kerberos realm of which the user is a member, such as MYREALM.COM.

5 Save your changes to the file.

6 Restart the SAS Event Stream Manager service. Run the appropriate command:

   For Red Hat Enterprise Linux 6.7:

   sudo service sas-viya-esm-service-default stop
   sudo service sas-viya-esm-service-default start

   For Red Hat Enterprise Linux 7.x or SUSE Linux:

   sudo systemctl stop sas-viya-esm-service-default
   sudo systemctl start sas-viya-esm-service-default
Validating the Deployment

Verify SAS Event Stream Manager Status

To verify that a deployment of SAS Event Stream Manager has completed successfully, check that the required SAS services are available. You can check the status of all the SAS Event Stream Manager services by running the following commands on Red Hat Enterprise Linux 6.x:

```
sudo service sas-viya-esm-service-default status
```

Run the following commands on Red Hat Enterprise Linux 7.x or SUSE Linux:

```
sudo systemctl status sas-viya-esm-service-default
```

Here is typical command output from Red Hat Enterprise Linux 6.7 to indicate that the software is running normally:

```
sas-viya-esm-service-default is running
```

The output is different on Linux 7.x or SUSE Linux, but it reports that the service is running.

Troubleshoot Service Status

If the output from the `service sas-viya-all-services status` command indicates that one or more services is not up, you can restart an individual service using its script. The SAS Event Stream Manager service script is installed in the `/etc/init.d` directory.

Here is an example of the command to check the status of the SAS Event Stream Manager service on Red Hat Enterprise Linux 6.x:
sudo service sas-viya-esm-service-default status
To start the service if it is not running:
sudo service sas-viya-esm-service-default start
To check the status of the client service on Red Hat Enterprise Linux 7.x or SUSE Linux:
sudo systemctl status sas-viya-esm-service-default
To start the service if it is not running:
sudo systemctl start sas-viya-esm-service-default

Access Log Files
If you encounter difficulties during the deployment, log files that include information about installation and service status are written to the following directory:

/opt/sas/viya/config/var/log/
If the deployment fails, check the logs in this location first.

Verify SAS Message Broker

1. To verify that SAS Message Broker has been deployed correctly, go to the machine that you assigned to the [rabbitmq] host group.

2. Open a browser and go to the following address:
   - If HTTPS is enabled:
     https://RabbitMQ-IP-address:15672/#/
     
     Note: If you did not add compliant certificates and instead kept the default security settings and certificates, a message appears, stating that Your connection is not private. SAS recommends that you replace the certificates before you give end users access to SAS Viya. For details, see HTTPS Access to SAS Message Broker.

   - If HTTP is enabled:
     http://RabbitMQ-IP-address:15672/#/

     If the RabbitMQ logon window appears, then SAS Message Broker is functioning as expected.
Verify SAS Infrastructure Data Server

Use these steps to verify that SAS Infrastructure Data Server has been deployed correctly.

1. On the machine that you assigned to the [pgpoolc] host group, to check status:
   - On Red Hat Enterprise Linux 6.x and Linux 7.x:
     ```
sudo service sas-viya-sasdatasvrc-postgres status
     ```
   - For SUSE Linux:
     ```
sudo /etc/init.d/sas-viya-sasdatasvrc-postgres status
     ```

2. If SAS Infrastructure Data Server is running appropriately, you should receive a response like this:

   PGPoll is running with PID=11445
   Checking Postgresql nodes status...

<table>
<thead>
<tr>
<th>node_id</th>
<th>hostname</th>
<th>port</th>
<th>status</th>
<th>lb_weight</th>
<th>role</th>
<th>select_cnt</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>machine1</td>
<td>5452</td>
<td>up</td>
<td>0.250000</td>
<td>primary</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>machine2</td>
<td>5452</td>
<td>up</td>
<td>0.250000</td>
<td>standby</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>machine3</td>
<td>5452</td>
<td>up</td>
<td>0.250000</td>
<td>standby</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>machine4</td>
<td>5452</td>
<td>up</td>
<td>0.250000</td>
<td>standby</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>load_balance_node</th>
<th>replication_delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>true</td>
<td>0</td>
</tr>
<tr>
<td>false</td>
<td>0</td>
</tr>
<tr>
<td>false</td>
<td>0</td>
</tr>
<tr>
<td>false</td>
<td>0</td>
</tr>
</tbody>
</table>
   (4 rows)

   A status of **up** for a node indicates the node is running.
After you install, configure, and verify the deployment, you are ready to begin using SAS Event Stream Manager to manage SAS Event Stream Processing applications and analyze streaming event data in real time.

The next step is to read the SAS Event Stream Manager user documentation, which explains how to manage SAS Event Stream Processing deployments. You can find all the SAS Event Stream Processing documentation on the SAS Event Stream Processing product page.

The SAS Event Stream Processing product page also offers multiple tutorials and examples to help you learn to use SAS Event Stream Manager. See: https://support.sas.com/en/software/event-stream-processing-support.html#tutorials.
Managing Your Software

Support for Upgrades

Upgrades of the SAS Event Stream Manager software from version 5.2 or 6.1 to 6.2 are supported.

SAS Event Stream Manager 6.2 has not been tested with back-level versions of SAS Event Stream Processing. SAS recommends that you upgrade both products at the same time.

What Is an Update?

An update provides modifications for features that are not working as intended or adds minor software enhancements and compatibility. Software updates are released to address security issues when they occur, to address minor bugs discovered in the software, and to improve the operation of hardware or peripherals. These incremental updates improve the operation of your software and are small
enough that they do not require a new order. Updated software is intended to be compatible with existing configuration, content, and data.

To perform an update, you will run the same tools that were run during the initial deployment. You might determine that your software needs updating or you might be notified by SAS that updates are available.

What Is an Upgrade?

An upgrade adds significant feature changes or improvements to SAS Event Stream Manager. To perform an upgrade, you will run the same tools that were run during the initial deployment. You will need a new order to upgrade your deployed software, and you must get an updated version of the Orchestration CLI to create a new playbook.

An upgrade might require changes to the deployed software's configuration.

You might determine that your software needs upgrading or you might be notified by SAS that upgrades are available. SAS recommends creating a backup of the deployed software environment before performing an upgrade.

Updating Your SAS Event Stream Manager Software

Overview

An update replaces some or all of your deployed software with the latest versions of that software. You perform the update with the same command that was used to install SAS Event Stream Manager, and use the same software order and the same playbook. If you mirrored your software, you must also update the mirror.

 Updating your software requires an outage period because some services are stopped and restarted automatically during the update process.

Note: The update process preserves any user-modified configuration values in the vars.yml file, but changes made to other files in the deployment might be lost. Therefore, SAS recommends that you make changes to vars.yml when possible in order to avoid any loss of customizations that you made to other files.

You will need to know the location of the directory on each machine where you stored deployment and maintenance files.

If you are using a PDF version of this guide, go to the Deployment Guides web page at https://support.sas.com/en/documentation/install-center/viya/deployment-guides.html and verify that you have the latest version of the deployment documentation before you start the update process. The release date of each document is located in the bottom right corner of the front page.
User Requirements for Performing the Update

To perform the update process, you must have administrator privileges for the machine. In addition, your account must have superuser (sudo) access. To verify sudo user privileges, run the following command: `sudo -v` or `sudo -l`.

List the Packages That Are Available for Update

Deployments without a Mirror Repository

To list the packages that are available for the update process, run the following command:

- **On Red Hat Enterprise Linux:**
  ```
  sudo yum check-update "sas-**"
  ```
- **On SUSE Linux:**
  ```
  sudo zyppr list-updates | grep "sas-"
  ```

Deployments with a Mirror Repository

**Important:** The method that you use to list packages for deployments with a mirror repository depends on whether you have internet access.

With Internet Access

To list packages in a mirror repository in a deployment with internet access:

1. List the packages that are available for the update process by running the following command on the machine where the mirror repository is located:
   ```
   mirrormgr mirror diff --deployment-data path-to-deployment-zip-file-from-SOE --path path-to-mirror-destination --latest
   ```
2. Before performing the update, you must synchronize the mirror repository with SAS. To synchronize, run the following command on the machine where the mirror repository is located:
   ```
   mirrormgr mirror --deployment-data path-to-deployment-zip-file-from-SOE --path path-to-mirror-destination --latest
   ```

Without Internet Access

To list packages in a mirror repository in a deployment without internet access:

1. To list the packages that are available for the update process, run the following command on the machine where the connected mirror repository is located:
   ```
   mirrormgr mirror diff --deployment-data path-to-deployment-zip-file-from-SOE
   ```
Before performing an update, you must synchronize the mirror repository with SAS. To synchronize, run the following command on the machine where the connected mirror repository is located:

```
mirrormgr mirror --deployment-data path-to-deployment-zip-file-from-SOE
--path path-to-mirror-destination --latest
```

Move the files from the machine where the connected mirror repository is located to the machine where the unconnected mirror repository is located.

### Update with Yum

You can only use yum to update your software if your deployment is on Red Hat Enterprise Linux or an equivalent distribution. To update a SAS deployment using yum, repeat these steps for each machine in the deployment:

1. (Optional) Record the existing list of installed software before you begin.
   
   For example, you can use the following command to create a text file that lists all the SAS RPM packages:
   
   ```
sudo rpm -qg SAS > /sas/install/sas_rpms.txt
   ```
   
   Create a file that lists the SAS yum groups that are installed on a machine. For example, you can use the following command to create a text file that lists all the SAS yum groups:
   
   ```
sudo yum grouplist "SAS*" > /sas/install/sas_yumgroups.txt
   ```
   
   **Note:** If you receive a message such as the following, it can be ignored.

   Repository repositoryname is listed more than once in the configuration.

2. Stop all the SAS services on the machine:
   
   ```
sudo service sas-viya-all-services stop
   ```

3. To update all SAS software on the machine:
   
   ```
sudo yum update $(rpm -qg SAS)
   ```
   
   You must run this command to update any external software applications on which the SAS yum groups depend.

4. At the prompt Is this ok, review the available updates and then enter y.

5. Restart the services that are installed on the machine.
   
   To restart all the SAS services on the machine:
   
   ```
sudo service sas-viya-all-services start
   ```

6. (Optional) After the update process has completed, record the new list of installed software.
   
   On each machine in your deployment, create a file that lists the names and versions of all the RPM packages that are installed. For example, you can use the following command to create a text file that lists all the SAS RPM packages:
   
   ```
sudo rpm -qg SAS > /sas/install/new_sas_rpms.txt
   ```
On each machine in your deployment, create a file that lists the SAS yum groups that are installed on a machine. For example, you can run the following command to create a text file that lists all the SAS yum groups:

```
sudo yum grouplist "SAS*" > /sas/install/new_sas_yumgroups.txt
```

You can see the differences between the previous and current deployments by comparing the lists of installed software before the update and after the update.

Note: If you receive a message such as the following, it can be ignored.

```
Repository repositoryname is listed more than once in the configuration
```

---

### Update with Zypper

You can only use zypper to update your software if your deployment is on SUSE Linux or an equivalent distribution. To update your deployment using zypper, repeat these steps for each machine in the deployment:

1. (Optional) Record the existing list of installed software before you begin.
   
   On each machine in your deployment, create a file that lists the names and versions of all the RPM packages of the SAS software that are installed. For example, you can use the following command to create a text file that lists all the SAS RPM packages:
   
   ```
   sudo rpm -qg SAS > /sas/install/sas_rpms.txt
   ```
   
   Note: If you receive a message such as the following, it can be ignored.
   
   ``Repository repositoryname is listed more than once in the configuration``

2. Stop all the SAS services on the machine:
   
   ```
   sudo /etc/init.d/sas-viya-all-services stop
   ```

3. Stop the metering server:
   
   ```
   dfesp_xml_client -url "http://host-name:http-port/SASESP/exit"
   ```
   
   Replace `host-name` with the host name of the machine where the Metering Server is running.
   
   Replace `http-port` with the port number for the Metering Server. By default, it uses port 31001.

4. To update all SAS software on the machine:
   
   ```
   sudo zypper update "sas-*"
   ```

5. At the prompt `Continue? [y/n]`, review the available updates and then enter `y`.

6. To restart all the SAS services on the machine:
   
   ```
   sudo /etc/init.d/sas-viya-all-services start
   ```

7. (Optional) After the update process has completed, record the new list of installed software.
   
   On each machine in your deployment, create a file that lists the names and versions of all the RPM packages of the SAS software that are installed. For example, you can use the following command to create a text file that lists all the SAS RPM packages:
Update with Ansible

To update your deployment using Ansible:

1. (Optional) Record the existing list of installed software before you begin.

   On each machine in your deployment, create a file that lists the names and versions of all the RPM packages that are installed. Create this file in the directory on each machine where you stored deployment and maintenance files. For example, you can run the following command to create a text file that lists all the SAS RPM packages:

   ```
   sudo rpm -qg SAS > /sas/install/sas_rpms.txt
   ```

   On each machine in your deployment, create a file that lists the SAS host groups that are installed on a machine. Create this file in the directory on each machine where you stored deployment and maintenance files. For example, you can run the following command to create a text file that lists all the SAS host groups:

   ```
   on Red Hat Enterprise Linux:
   sudo yum grouplist "SAS*" > /sas/install/sas_hostgroups.txt
   ```

   To list the SAS packages on SUSE Linux, run the following command:

   ```
   sudo rpm -qa | grep "sas-
   ```

   Note: If you receive a message such as the following, it can be ignored.

   Repository repositoryname is listed more than once in the configuration

2. Review the *_deployment.* files in the existing deployment for any user-modified changes. If there are any user-modified changes to the *_deployment.* files, back up the file and update the vars.yml file with the changes before you perform the update. If you have questions, contact SAS Technical Support.

   Note: SAS recommends that you add your customizations to the vars.yml file rather than to a *_deployment.* file in order to preserve your customizations. Otherwise, your customizations would be lost during the update process.

3. Stop the metering server:

   ```
   dfesp_xml_client -url "http://host-name:http-port/SASESP/exit"
   ```

   Replace host-name with the host name of the machine where the metering Server is running.

   Replace http-port with the port number for the metering server. By default, it uses port 31001.

4. To initiate the update, run the following command:

   ```
   ansible-playbook update-only.yml
   ```
5 (Optional) After the update process has completed, record the new list of installed software.

Create a file that lists the names and versions of all the RPM packages that are installed. Create this file in the directory on each machine where you stored deployment and maintenance files. For example, you can use the following command to create a text file that lists all the SAS RPM packages:

```
sudo rpm -qg SAS > /sas/install/new_sas_rpms.txt
```

You can see the differences between the previous and current deployments by comparing the lists of installed software before the update and after the update.

---

### Upgrading Your Software

#### Overview

An upgrade adds significant feature changes or improvements to your deployed software. To perform an upgrade, you will run the same tools that were run during the initial deployment. You will need a new software order to upgrade your deployed software. An upgrade might require changes to the deployed software’s configuration. SAS recommends that you create a backup of the deployed software environment before performing an upgrade.

Upgrading SAS Event Stream Manager requires an outage period because some services are stopped and restarted automatically during the upgrade process.

You will need to know the location of the directory where you stored deployment and maintenance files.

---

### Prepare to Upgrade SAS Event Stream Manager

To prepare to upgrade your deployment:

1 (Optional) Record a list of the existing installed software before you begin.

Create a file that lists the names and versions of all the RPM packages that are installed. Create this file in the directory on each machine where you stored deployment and maintenance files. For example, you can run the following command to create a text file that lists all the SAS RPM packages:

```
sudo rpm -qg SAS > /sas/install/sas_rpms.txt
```

Create a file that lists the SAS yum groups that are installed on a machine. Create this file in the directory on each machine where you stored deployment and maintenance files. For example, you can run the following command to create a text file that lists all the SAS yum groups:

```
sudo yum grouplist "SAS*" > /sas/install/sas_yumgroups.txt
```

Note: If you receive a message such as the following, it can be ignored.
2 When performing an upgrade, you receive a new Software Order Email (SOE) from SAS. Use your SOE to download the SAS Orchestration CLI.

3 Using the SAS Orchestration CLI that you downloaded, create a new playbook using the instructions on the SAS Orchestration Command Line Interface (CLI) download site.

4 You must extract the new playbook to a location that is different from that of your original playbook. For example, if you extracted your original playbook to `/sas/install/`, you might extract the new playbook to `/sas/upgrade/` instead. You must extract the new playbook to a location that is different from the one that you used for your deployment for these reasons:
   
   - To preserve the original vars.yml file and the inventory file.
   - To ensure that the playbook directory correctly reflects what is delivered. If a new playbook is mistakenly extracted over an existing playbook, files that were removed in the newer playbook would still be available and could negatively affect the process for researching and resolving deployment issues.

   To extract the new playbook, use a command that is similar to the following:

   ```bash
tar xf SAS_Viya_playbook.tgz -C /sas/upgrade/
   ```

5 Merge the vars.yml file and the inventory file from the previous deployment into the new playbook. If the previous inventory file contains any spaces that are used to indent machine names, do not include the extra spaces.

   a Compare the two vars.yml files, and compare the two inventory files to check for additions or changes in the newer set of files.

   ```bash
diff /sas/install/sas_viya_playbook/vars.yml /sas/upgrade/sas_viya_playbook/vars.yml

diff /sas/install/sas_viya_playbook/inventory-file /sas/upgrade/sas_viya_playbook/inventory.ini
   ```

   b If the new files contain new content, merge your customized edits from the two original files into the two new files. If a key/value pair in the original file is not included in the new file, you do not need to add the key/value pair to the new file. If you have any questions, contact SAS Technical Support.

---

**Stop a Clustered RabbitMQ Configuration**

If RabbitMQ is deployed in a clustered configuration, additional steps are required in the upgrade process.

1 On your Ansible controller host, locate the `[rabbitmq]` host group in your playbook inventory file.

2 If only one RabbitMQ target is defined, skip the rest of this section and upgrade your SAS Viya deployment.

3 If more than one RabbitMQ target is defined, log on to the last `rabbitMQ` target.

4 Stop the RabbitMQ server using the appropriate command:
If the RabbitMQ target is a 6.x Linux system, run the following command:
```
sudo service sas-viya-rabbitmq-server-default stop
```

If the RabbitMQ target is a 7.x Linux system, run the following command:
```
sudo systemctl stop sas-viya-rabbitmq-server-default
```

5 For the other RabbitMQ targets, log on to each RabbitMQ target and run a command to stop RabbitMQ, in the reverse order in which they are listed in the inventory.ini file.

6 Edit the file `/sas/upgrade/sas_viya_playbook/internal/config-start.yml` to set the line `include: rabbitmq.yml` immediately after the line `include: pgpoolc.yml`. Here is an example:
```
include: pgpoolc.yml
include: rabbitmq.yml
```

7 Save your changes to `/sas/upgrade/sas_viya_playbook/internal/config-start.yml` and close the text editor.

---

Upgrade SAS Event Stream Manager

To upgrade your deployment:

1 Log on to the PostgreSQL machine in your deployment.

2 Run the following command:
```
sudo cat -n /opt/sas/viya/config/etc/sasdatasvrc/postgres/pgpool0/pool.cdf
```
All entries in the command’s output should display `healthy`.

3 Run the following command:
```
sudo /etc/init.d/sas-viya-sasdatasvrc-postgres status
```

4 Open vars.yml and locate the `INVOCATION_VARIABLES` section.

5 Compare the `NODE_TYPE` of each node in the PostgreSQL cluster to the output of the `sudo /etc/init.d/sas-viya-sasdatasvrc-postgres status` command.

   - `P` - Primary
   - `S` - Secondary

If `NODE_TYPE` for each node in vars.yml does not match the output of the `sudo /etc/init.d/sas-viya-sasdatasvrc-postgres status` command, you must edit vars.yml.

6 Compare the hostnames in the output of the `sudo /etc/init.d/sas-viya-sasdatasvrc-postgres status` command with the hostname assignments in inventory.ini. If the hostnames do not match, you must edit inventory.ini.

7 Compare the deploy target assignments for each node in inventory.ini to the deploy target assignments for each node in the `INVOCATION_VARIABLES` section of vars.yml. If the deploy target assignments do not match, edit vars.yml to match inventory.ini.
8 Install your software using the steps in the installation chapter.

9 After the software has been installed, complete the following tasks:
   a “Complete SAS Event Stream Manager Setup” on page 35.
   b Validate the Deployment on page 39.

10 The script will interactively prompt you to log on. You must log on using a profile that is a member of the SASAdministrators group. After you log on, the script will run against all CAS servers in the environment and set the new ACLs, and then exit.

11 (Optional) After the upgrade process has completed, record the new list of installed software. For example, you can run the following command to create a text file that lists all the SAS RPM packages:

   ```bash
   sudo rpm -qg SAS > /sas/install/new_sas_rpms.txt
   ```

   You can see the differences between the previous and current deployments by comparing the lists of installed software before the upgrade and after the upgrade.

   Note: If you receive a message such as the following, it can be ignored.

   Repository repositoryname is listed more than once in the configuration
Uninstalling SAS Event Stream Manager

What deploy-cleanup Does

When you use the deploy-cleanup command described in the following sections, it performs these actions:

1. Stop all SAS services.
2. Remove all SAS RPMs.
3. Delete any remaining SAS .pid files.
4. Delete the entitlement_certificate.pem and SAS_CA_Certificate.pem files.

The deploy-cleanup command renames the /opt/sas/viya directory to /opt/sas/viya_epoch. Also, the /opt/sas/spre directory is renamed as /opt/sas/spre_epoch.

The uninstallation does not remove the customized script that you received with your SOE, and it does not remove any users that have been set up.

Uninstall the Software

To uninstall your software from a single-machine deployment, run the following command:

```bash
ansible-playbook -i host_local deploy-cleanup.yml
```

If the environment requires one or more passwords, the command must include additional parameters as specified here:
<table>
<thead>
<tr>
<th>Password Requirements</th>
<th>Additional Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Password for sudo only</td>
<td>--ask-become-pass</td>
</tr>
<tr>
<td>Password for SSH only (applies only if the Ansible controller is on a different machine than your SAS software)</td>
<td>--ask-pass</td>
</tr>
<tr>
<td>Password for both sudo and SSH (applies only if the Ansible controller is on a different machine than your SAS software)</td>
<td>--ask-become-pass --ask-pass</td>
</tr>
</tbody>
</table>

When the appropriate command is executed, Ansible performs a group uninstallation, which removes your SAS software, including both certificates. It also renames the `/opt/sas/viya` directory to `/opt/sas/viya_<epoch>`, where `<epoch>` specifies the UNIX epoch (the number of seconds that have elapsed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970).