SAS® Visual Investigator 10.3 on the Cloud: Deployment Guide
# Contents

**Chapter 1 / Introduction**
- About This Guide ........................................... 1
- How Deployment Works .................................. 1
- What Gets Deployed ...................................... 1
- Contact SAS Technical Support ....................... 2

**Chapter 2 / System Requirements for SAS Visual Investigator**
- Cloud Platform Software Requirements ........... 3
- Virtual Machine Operating System and Software Requirements ................. 4
- Security Requirements .................................. 5
- Client Requirements ................................... 6

**Chapter 3 / Pre-installation Tasks**
- Enable Required Ports ................................ 7

**Chapter 4 / Installing SAS Visual Investigator on SAS Viya**
- Prepare the Installation Tools ....................... 9
- Prepare the Installation Environment .......... 12
- Deploy SAS Visual Investigator .................. 16

**Chapter 5 / Post-Installation Tasks**
- Configure and Onboard Initial Tenant .......... 19

**Chapter 6 / Validating the Deployment**
- Verify RabbitMQ ......................................... 21
- Verify PostgreSQL ...................................... 21
- Validate Elasticsearch ................................. 21
- Validate Consul ....................................... 22

**Chapter 7 / Completing the Deployment**
- Further Documentation ......................... 25

**Chapter 8 / Appendix A: Troubleshooting**
- BOSH .................................................. 27
- Cloud Foundry Run Time ......................... 30
About This Guide

Use this guide to deploy SAS Visual Investigator in a private cloud.

- The contents of this document are subject to continual updates. Make sure that you have the latest version of this document, which is available from the SAS Viya Install Center.
- To use this guide successfully, you should have a working knowledge of BOSH, Cloud Foundry, and vSphere or OpenStack, depending on your environment.
- Unless another situation is specifically cited, the information in this guide pertains to the software that you ordered.

How Deployment Works

SAS Visual Investigator takes advantage of industry-standard software and tools to deploy to the cloud.

- Cloud Foundry is the supported PaaS.
- vSphere and OpenStack are the supported IaaS environments.
- BOSH is used to upload the required stemcells and releases for SAS Visual Investigator. After the stemcells and releases are in place, you will create files that are used in deploying the services and applications. Deploying SAS Visual Investigator consists of deploying stateful services to BOSH, and deploying stateless services and the microservices.
- The SAS Visual Investigator services and applications are packaged as a container. You will access the container from a binary installer file that you download from SAS.

What Gets Deployed

This guide provides information for deploying the following products and supporting components:
SAS Visual Investigator

SAS Cloud Analytic Services (CAS), which is used as the run-time environment and the analytics server for SAS Visual Investigator

SAS/ACCESS to Oracle

SAS/ACCESS to PostgreSQL

SAS/ACCESS to DB2

SAS/ACCESS to Teradata

---

**Contact SAS Technical Support**

Technical support is available to all customers who license SAS software. However, we encourage you 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 call, explore the SAS Support website at [support.sas.com/techsup/](http://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.
System Requirements for SAS Visual Investigator

Cloud Platform Software Requirements

Cloud Foundry Requirements

SAS Visual Investigator requires versions 254 through 276 of Cloud Foundry.

The supported stemcell type is CentOS 3345.7, based on CentOS 7.x.

For additional information about the supported run-time environment and stemcells, see https://support.sas.com/en/documentation/third-party-software-reference/viya/support-for-operating-systems.html#cf.

BOSH Director

BOSH Director is required in order to install SAS Viya components. BOSH Director provides cloud orchestration and management, including VM creation, deployment, and other software life cycle events. SAS Visual Investigator deployment uses BOSH manifest schema v2 only. BOSH CLI 1.3262.26.0 or later is required.

Other Required Software

The following third-party software is included with your SAS software:

- HashiCorp Consul – Enables service discovery and configuration.
- PostgreSQL – Creates the SAS Infrastructure Data Server, which stores user content such as reports, custom groups, audit records, and preferences.
- RabbitMQ – Provides an open-source, standards-based platform for SAS components and applications to send and receive messages.
- Elasticsearch – Provides search capabilities.
- SAS Cache Locator – Used for session caching.

**Supported Data Sources**

The following external data stores are supported by SAS Visual Investigator:

- IBM DB2:
  - DB2 V8.x and later for Linux, UNIX, Windows
  - DB2 V8.1 and later for z/OS
  - DB2 UDB V8.1 for z/OS
  - DB2 i 6.1 and later
  - DB2 for i5/OS V5R4 and later
- Oracle 12c
- PostgreSQL 9.4
- Teradata: 14.00, 14.10, 15.00, 15.10, and 16.00

Each external data source also requires a SAS/ACCESS product. These products might have individual system requirements. For more information, see [https://support.sas.com/en/documentation/third-party-software-reference/viya/support-for-databases.html](https://support.sas.com/en/documentation/third-party-software-reference/viya/support-for-databases.html).

A PostgreSQL database is also used as an internal data store, named SAS Infrastructure Data Server. It is based on PostgreSQL version 9 and is configured specifically to support SAS software by storing user content and preferences.

---

**Virtual Machine Operating System and Software Requirements**

**Requirements for the Deployment Machine**

The VM instance or physical machine that is used to install SAS Visual Investigator has the following requirements:

- Red Hat Enterprise Linux or CentOS 7.2 or later must be installed.
- The systemd-nspawn component must be installed.
- 30 GB of free space must be available.

**BOSH Virtual Machine Requirements**

The requirements for virtual machines are typically dependent on the number of users who access the SAS Visual Investigator environment and the amount of data that is imported and indexed. However, with SAS Visual Investigator for Cloud Foundry, requirements for the VMs that are deployed by BOSH are different from the
requirements for resources that are dynamically allocated for applications within the Cloud Foundry Elastic Runtime environment. Although the Elastic Runtime runs on host VMs called *cells* in Cloud Foundry, VMs that are deployed by BOSH are referred to as *component VMs*.

BOSH component VMs are not dynamically allocated. Use the guidelines in the following table to allocate initial VM resources in BOSH:

<table>
<thead>
<tr>
<th>Role</th>
<th>VCPUs</th>
<th>Disk Space</th>
<th>RAM</th>
<th>Storage Volume (Min.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cache</td>
<td>4</td>
<td>40 GB</td>
<td>8 GB</td>
<td>20 GB</td>
</tr>
<tr>
<td>CAS Controller</td>
<td>2</td>
<td>20 GB</td>
<td>2 GB</td>
<td>20 GB</td>
</tr>
<tr>
<td>CAS Workers</td>
<td>8</td>
<td>40 GB</td>
<td>16 GB</td>
<td>20 GB</td>
</tr>
<tr>
<td>Consul</td>
<td>2</td>
<td>20 GB</td>
<td>2 GB</td>
<td>20 GB</td>
</tr>
<tr>
<td>Elasticsearch Master</td>
<td>2</td>
<td>20 GB</td>
<td>2 GB</td>
<td>20 GB</td>
</tr>
<tr>
<td>Elasticsearch Client</td>
<td>4</td>
<td>40 GB</td>
<td>4 GB</td>
<td>20 GB</td>
</tr>
<tr>
<td>Elasticsearch Data</td>
<td>16</td>
<td>40 GB</td>
<td>32 GB</td>
<td>20 GB</td>
</tr>
<tr>
<td>PgPool</td>
<td>8</td>
<td>40 GB</td>
<td>8 GB</td>
<td>20 GB</td>
</tr>
<tr>
<td>PostgreSQL</td>
<td>16</td>
<td>40 GB</td>
<td>32 GB</td>
<td>20 GB</td>
</tr>
<tr>
<td>RabbitMQ</td>
<td>4</td>
<td>40 GB</td>
<td>4 GB</td>
<td>20 GB</td>
</tr>
</tbody>
</table>

*Note:* The size of the storage volume shown is the minimum amount. For CAS workers, Elasticsearch data, and the PostgreSQL database (SAS Infrastructure Data Server), you will likely adjust the size of the volume to accommodate the amount of data that is generated by end users. You can determine this amount based on the number of users and their expected usage levels.

If you decide to use a single VM for CAS, and use only the controller node without workers, follow the guidelines for CAS workers that are provided in the table.

## Security Requirements

### User Accounts

The user account that is used to perform the deployment process must have sudoers privileges and a home directory.

SAS Viya requires a predefined user account that enables the administrator to log on after the deployment has completed. This account is called *viadmin* in this document. Before the deployment process is started, the viadmin user account must exist as a valid LDAP user.

The administrator creates groups for users as well as a group for administrators by adding users and groups through the Administration interface.
Authentication

SAS Visual Investigator supports LDAP for user authentication. Microsoft Active Directory and OpenLDAP are supported LDAP implementations. In addition, the CAS server uses OAuth tokens for all clients in your deployment. The deployment process configures LDAP settings automatically.

Before SAS Visual Investigator is deployed, ensure that these requirements have been met:

☐ Make sure that SAS Visual Investigator users are able to authenticate to your LDAP provider.

☐ Make sure that SAS Visual Investigator has Read access to your LDAP server.

Note: To bind to the LDAP server, SAS Visual Investigator requires a userDN and password. LDAP anonymous binding is not supported.

☐ (Optional) If you are deploying multi-tenancy, verify that tenants are set up in LDAP. Use the documentation that is appropriate for your LDAP implementation to set them up.

After the deployment has completed, the designated administrator logs on to the administration application as the viadmin user. Valid LDAP user accounts can then be added from the Administration interface.

Client Requirements

Web Browser Requirements

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

For information about supported web browsers to access SAS Visual Investigator user interfaces, see https://support.sas.com/en/documentation/third-party-software-reference/viya/support-for-web-browsers.html.
Enable Required Ports

The following ports are used by SAS Visual Investigator and 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>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTPD</td>
<td>80 (internal)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>443 (external)</td>
<td></td>
</tr>
<tr>
<td>default Erlang Port Mapper Daemon (epmd) port</td>
<td>4369</td>
<td></td>
</tr>
<tr>
<td>SAS Infrastructure Data Server</td>
<td>5430–5439</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>CAS Server Starting Port</td>
<td>5577</td>
<td>Used by clients to make binary connections to CAS.</td>
</tr>
<tr>
<td>CAS Communicator Port</td>
<td>5580</td>
<td></td>
</tr>
<tr>
<td>default SAS Messaging Broker AMQP client access port</td>
<td>5672</td>
<td></td>
</tr>
<tr>
<td>SAS Configuration Server</td>
<td>8300–8309, 8500 and 8501</td>
<td>SAS uses HashiCorp Consul as its configuration server. All Consul ports should be open to TCP and UDP traffic.</td>
</tr>
<tr>
<td>Object Spawner</td>
<td>8591</td>
<td></td>
</tr>
<tr>
<td>CAS Server Monitor</td>
<td>8777</td>
<td>Used by clients to make REST HTTP calls to CAS, as with the Python REST interface.</td>
</tr>
<tr>
<td>Process</td>
<td>Required Port</td>
<td>Notes</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>---------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>Elasticsearch</td>
<td>9200</td>
<td></td>
</tr>
<tr>
<td>Default PgPool port</td>
<td>9432</td>
<td></td>
</tr>
<tr>
<td>SAS Cache Locator</td>
<td>10335</td>
<td></td>
</tr>
<tr>
<td>default SAS Messaging Broker management</td>
<td>15672</td>
<td>web console port</td>
</tr>
<tr>
<td>SAS/CONNECT Spawner</td>
<td>17551</td>
<td></td>
</tr>
<tr>
<td>SAS Cloud Analytic Services Server</td>
<td>19990-19999</td>
<td></td>
</tr>
<tr>
<td>default SAS Messaging Broker clustering port</td>
<td>25672</td>
<td></td>
</tr>
<tr>
<td>SAS Cache Server</td>
<td>40404</td>
<td></td>
</tr>
</tbody>
</table>
Installing SAS Visual Investigator on SAS Viya

Prepare the Installation Tools
- Download the Binary Installer File
- Unzip the Downloaded File
- Extract the Binary File
- Start the Container
- Sign On to the Container
- Access Product Help

Prepare the Installation Environment
- Set the Site Name
- Set the Configuration File
- Edit the Configuration File
- Save the Configuration File
- Authenticate to Cloud Foundry and BOSH
- Upload the BOSH Stemcells
- Upload the BOSH Releases
- Create the Files for Deployment
- Prepare to Examine the Files
- Examine the Files Related to BOSH
- Examine the Files Related to Cloud Foundry
- Using Logs

Deploy SAS Visual Investigator
- Overview
- Deploy the Stateful Services to BOSH
- Deploy the Stateless Applications and Microservices

Prepare the Installation Tools

Download the Binary Installer File
Download the binary installer file to a physical machine or a virtual machine (VM) that has access to the IaaS environment. The VM is used as the secure administration host (jump server) for the installation.

Download the binary installer file sas.bin to the /home/installer-ID directory on the jump server. For example, if the binary installer file has been placed on a web server, you can use the `wget` command to download the binary installer file. Here is an example:

```
wget http://0.0.0.0/visualinvestigator__10_3__xxx_lax.zip
```
Unzip the Downloaded File

Unzip the downloaded file using the `unzip` utility:

```
unzip visualinvestigator_10_3__xxx__lax.zip
```

Extract the Binary File

After the downloaded file has been unzipped, extract the binary file:

```
bash visualinvestigator_10_3__xxx__lax.bin
```

Here is an excerpt of the results:

```
Validating archive...
Extracting to /home/test/sas...
```

The installer creates a top-level directory named `sas`, which contains two subdirectories: `bin` and `image`.

Here are typical results:

```
--2016-06-20 14:53:04--  http://10.120.16.120/visualinvestigator_10_3__xxx__lax.zip
Connecting to 0.0.0.0:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5508425807 (5.1G) [application/octet-stream]
Saving to: 'visualinvestigator_10_3__xxx__lax.zip'
100%[=================================================>] 5,508,425,807 74.4MB/s
in 1m 43s

2016-06-20 14:54:47 (50.8 MB/s) - 'visualinvestigator_10_3__xxx__lax.zip' saved [5508425807/5508425807]
```

`bin` consists of a single Bash script named `start`.

`image` is a systemd-nspawn namespace container that is invoked by the `start` Bash script and contains functionality that is used in deployment activities for SAS Visual Investigator. The container can be shut down and restarted whenever these functions are needed.

In addition, the entire container, with any changes that are made, can be repackaged, archived, or moved to another server or a VM as needed.

Start the Container

To start the container, run the `start.sh` script from the user’s home directory:

```
./sas/bin/start
```

Here are typical results:

```
Starting SAS Visual Investigator Deployment Container...
Spawning container <instance>.<hostname>.unx.sas.com on /home/test/sas/image.
Press ^C three times within is to kill container.
systemd 219 running in system mode. (+PAM +AUDIT +SELINUX +IMA -APPARMOR +SMACK +SYSSVINIT +UTMP
+LIBCRYPTSETUP +GCRYPT +GNUTLS +ACL +XZ -LZ4 -SECCOMP +BLKID +ELFUTILS +KMOD +IDN)
Detected virtualization systemd-nspawn.
Detected architecture x86-64.
```
Welcome to CentOS Linux 7 (Core)!

Because the container attempts to alter the state of the host operating system, the following error message
might be displayed. You can safely ignore the message:

[FAILED] Failed to listen on RPCbind Server Activation Socket.
[FAILED] Failed to start LSB: Bring up/down networking.
See 'systemctl status network.service' for details.

Sign On to the Container

1  After the container is started, a login prompt is displayed:

    CentOS Linux 7 (Core)
    Kernel 3.10.0-693.2.2.el7.x86_64 on an x86_64

    np140801 login: sas

2  At the prompt, enter the user ID sas and then enter this password in the specified casing XyZZy.

After you sign on successfully, the screen appears as follows:

    Last login: Wed Jun 15 12:41:34 on console

    Site Unset 12:23:56 !4 [~]

The prompt now includes the string, Site Unset, followed by a timestamp. The Z shell (zsh) is set
automatically. It is the standard shell for a SAS Visual Investigator deployment.

Note: You should change the password immediately.

Note: Do not change shells. Any other shell might produce unexpected results.

The window of your X terminal emulator displays the title Container. The title is a reminder that your work
environment is within the container and that you are no longer working within the host file system.

Access Product Help

To access Visual Investigator help, run the following command:

    sas help -g

To show product information, run the following command:

    sas show info

    SAS Visual Investigator : Info

        Product
        Name              : SAS Visual Investigator
        Version:          : 10.3.0.1562.0
        Build Date:       : 2017-10-20.092451

Note: The information displayed by the sas show info command will change after loading the data from
the JSON configuration file.

For more details about your configuration, run the following command:

    sas conf use configuration-filename
Prepare the Installation Environment

Set the Site Name

The site name uniquely identifies your deployment. The container can manage multiple deployments at different sites. This enables you to manage development, test, and production environments from a single deployment container. You can also manage a paired set of sites that will act as a warm backup for disaster recovery.

To set the site name, run the command:

```
sas site set site-name
```

The name can contain only lowercase alphabetic characters, numbers, and hyphens. Nonalphanumeric characters, including a space, are not allowed. SAS recommends a site name that is three to ten characters long.

The site name will also be used as the datacenter name in Consul.

Note: Once the site name has been set, it will appear at the far left side of the prompt, replacing the initial value, Site Unset.

Set the Configuration File

1. You must choose a template file as a basis for the configuration file that you will set, edit, and save. To list the available template files, run the command:

```
sas conf list
```

Here are the results:

```
SITE CONFIGURATIONS:
SYSTEM CONFIGURATIONS:
openstack
vsphere
```

SEE ALSO:
```
sas conf set <configurationName>  Change the active configuration
```

To use a configuration, use the command:

```
sas conf set <configurationName>
```

Now using configuration:

2. Select the template file (vSphere or OpenStack) that corresponds with the underlying IaaS for your Cloud Foundry installation and run the following command:

```
sas conf set configuration
```

where xxxx is either vsphere or openstack.

3. Save the configuration file with a new configuration filename that is meaningful in your environment.

```
sas conf save filename
```

Here is the output:

```
Saved configuration file: filename
```

4. Set the new configuration file as the active configuration file for the framework to use.

```
sas conf set filename
```
Here is the output:

Now using configuration: filename

Edit the Configuration File

Once the template file is ready to use, edit the configuration file using the vim editor, using the following command:

```bash
sas conf edit
```

Note: SAS has configured the vim utility to check JSON syntax. If the file contains invalid JSON syntax, you will be notified when you exit the utility.

Save the Configuration File

To save the changes that you have made to a permanent location, run the following command:

```bash
sas conf save filename
```

Here is the output:

Saved configuration file: test

This command does not change the active configuration file. Instead, it saves a copy of it to a saved configuration directory. You can see your configuration name now by running the command:

```bash
sas conf list
```

Here is the output:

```
SITE CONFIGURATIONS:
test (active configuration)

SYSTEM CONFIGURATIONS:
opencrypt
vSphere

SEE ALSO:
sas conf set configuration-name Change the active configuration
```

Authenticate to Cloud Foundry and BOSH

Now that you have entered the information about your Cloud Foundry installation and the servers to be created, you need to connect to Cloud Foundry. This step is required even if the jump server is signed on to Cloud Foundry and BOSH. This step is also required whenever you change your configuration to a new configuration that has different authentication details.

To authenticate, run the command:

```bash
sas cf auth
```

Here is typical output:

```
Authentication for user: sas
Current target is https://0.0.0.0:25555 (your-bosh)

API endpoint: https://api.sas.sas.com (API version: 2.48.0)
User: test
Org: test
Space: test
```
If authentication is successful, then you know that the data that you entered for the Cloud Foundry and BOSH environments is correct. If authentication is not successful, refer to the data that you entered for the director and run-time sections of the configuration file. Correct them for your installation and try again.

**CAUTION!** If you do not have a working connection to both BOSH and Cloud Foundry, do not continue to the next section.

### Upload the BOSH Stemcells

Once connectivity is established to the BOSH environment, you must upload the required stemcells for the creation of virtual machines (VMs). This is done in the IaaS using the BOSH Cloud Provider Interface (CPI).

1. Check to see whether the required stemcells are already installed. Run the following command:

   ```bash
   bosh stemcells
   ```

   If the output contains the following information, the correct stemcell is already installed. You can skip the remainder of this section.

   ```
   Name="bosh-openstack-kvm-centos-7-go_agent"
   Version="3445.7"
   ```

2. If the `bosh-openstack-kvm-centos-7-go_agent` stemcell is not installed, run the following command:

   ```bash
   sas bosh upload stemcells
   ```

   Two stemcells are delivered: one each for OpenStack and vSphere.

3. To ensure that the correct stemcells are loaded, even if there are existing stemcells, run the following command:

   ```bash
   sas bosh upload stemcells
   ```

   If the correct stemcells are already installed, you receive a warning message stating that the correct version already exists. You can safely ignore this message and continue with the upload.

### Upload the BOSH Releases

Now you need to upload the included BOSH releases to the BOSH blob storage so that they are available to BOSH when it is time to deploy the services.

To upload the BOSH releases, run the following command:

```bash
sas deploy releases
```

The following release packages are delivered, one for each of the stateful services.

- Cloud Analytic Server (CAS)
- Elasticsearch
- PostgreSQL
- Consul
- RabbitMQ
- PGPool
- SAS Cache Locator

### Create the Files for Deployment

Once the stemcells and releases are in place, you must create files that are used in deploying SAS Visual Investigator services and apps.
This step should be performed each time there is a change in the active configuration file. If these files have already been generated, then remove them from the container first by running the following commands from the home directory of the installer ID.

```
    cd ~/site-name
    rm -rf *
```

The configuration file provides the values to be substituted into the template files that are part of the framework.

To create the file system and files, run the following command:

```
sas deploy files
```

As the command runs, it copies the JAR files that will be deployed in Cloud Foundry. Then it performs the token substitution on the various files that are required for deploying the solution.

When file creation is complete, token substitution is checked to determine whether there are any unresolved tokens. If tokenization is successful, the message **All tokens successfully resolved** is displayed. If tokenization is incomplete and an error message is shown, the configuration file is incomplete and needs to be edited again.

**Note:** Be sure to remove the files that were generated before running the command `sas deploy files` again.

---

## Prepare to Examine the Files

Examination of the BOSH and Cloud Foundry manifest files is required in order to verify that the information in the files is correct. You must have knowledge about the Cloud Foundry environment and the vim utility.

**Note:** Do not change the information in the manifest files manually because the data comes from the configuration file. Any changes should be made to the configuration file. The contents of the `site-name` directory should then be deleted, and the command to deploy files should be rerun to create a new set of manifest files.

---

## Examine the Files Related to BOSH

The manifest files that deploy stateful services into the BOSH managed environment are located in the `/home/sas/site-name/services` directory in the container.

To examine the manifest files:

1. Change to the `/home/sas/site-name/services` directory.
2. Run the following command to display the vim editor: `vi */man*.yml`. This displays the vim editor in the window, with the first file in the vim buffer. Use the `vim:n` command to edit the next file match. Use the `:rew` command to rewind to the beginning if you want to review all files again.

**Note:** It is acceptable to change the configurable entries in the jobs section of the manifest.yml file, which is in the Postgres directory under the `services` directory. They are intended to be managed outside the framework.

When examining the manifest files, check the following items:

- `director_uuid` – make sure this matches what was entered in the configuration file.
- Verify that the IP address is the correct one entered for each server. Look under `jobs` and then `static_ips`.

Once you have examined the BOSH manifest files, examine the script files in the services directory. To examine the script files:

1. Change to the `/home/sas/site-name/services` directory.
2. Run the following command to display the vim editor:

```
    vi *//*.sh
```
The most important script in this group is the **post_deploy_consul.sh** script. In this file, check for empty values ("") and determine whether an empty value is reasonable. Some expected empty values include the `archive.storage.local.destination` property and in the **Folder** loop at the bottom of the file. A property with an empty value might indicate that a JSON property in the configuration file was not specified. If you discover any entries like this, return to the configuration file and make sure that all required entries are completed. Then delete the contents of the home directory and re-create the files.

**Note:** The following service_tag parameters are blank for all environments: `elasticsearch/deploy.sh service_tags`, `postgres/deploy.sh service_tags`, and `rabbitmq/deploy.sh service_tags`. This is standard and can be ignored.

**Note:** If you are using OpenStack, the `static_ips` addresses in the `cas-worker.yml` and `manifest-data.yml` files are blank. This is standard for OpenStack and can be ignored.

---

**Examine the Files Related to Cloud Foundry**

Manifest files that deploy stateless applications and microservices into the Cloud Foundry run-time environment are located in the `/home/sas/apps` directory in the container.

To examine the manifest files:

1. Change to the `/home/sas/site-name/apps` directory.
2. Run the following command to display the vim editor:
   ```bash
   vi */man*.yml
   ```

   In these files, check for empty values ("") and determine whether an empty value is reasonable. This might indicate that a JSON property in the configuration file was not specified. If you discover any entries like this, return to the configuration file and make sure that all required entries are completed. Then delete the directories under the `/home/sas` directory and re-create the files.

---

**Using Logs**

Log data for applications and microservices is provided through the Cloud Foundry Loggregator system. Using a Cloud Foundry firehose and nozzle to collect log data and to redirect it to a logging server is the standard method of providing access to application logs. The rsyslog logging service has been added to the services deployed in Cloud Foundry BOSH. This information can be routed to any rsyslog server over TCP. This method is compatible with any of the various log store and display applications.

---

**Deploy SAS Visual Investigator**

**Overview**

Installing SAS Visual Investigator consists of the following tasks:

- Install the stateful services to BOSH.
- Install the stateless applications and microservices.

**Deploy the Stateful Services to BOSH**

To deploy the stateful services to BOSH, run the following command:

```bash
sas deploy services
```
When deployments are complete, test the applications to ensure that the installation is working as expected. For more information, see "Validating the Deployment" on page 21.

**CAUTION!** Do not proceed to the next task if you see any errors during the installation of the stateful services. For error recovery, see "Appendix A: Troubleshooting" on page 27.

## Deploy the Stateless Applications and Microservices

1. To install the stateless applications and microservices, run the following command:

   ```bash
   sas deploy apps
   ```

   This command is similar to the command that installs the stateful services. The `sas deploy apps` command runs the `deploy.sh` script in the `/home/sas/apps` directory. This script executes each of the `deploy.sh` scripts in the subdirectories under the `apps` directory. These scripts deploy application JAR files in the Cloud Foundry run-time environment.

2. To display the status of the full installation, run the following command:

   ```bash
   sas show status
   ```

   **Note:** The `sas show status` command implicitly performs a status check of the VMs and it also runs the `cf apps` and `cf routes` commands. To ensure that all applications are fully started, run the command:

   ```bash
   sas show instances.
   ```

   This will return the actual state of the applications within the Cloud Foundry runtime.

3. Run the following command to display information about the installed applications:

   ```bash
   sas show info
   ```

   The results include information about the product, the configuration, the URLs that enable you to connect to the application interfaces, and the environment. All the information is surfaced from the configuration file.

4. To ensure that the stateful services and applications are up and in a satisfactory condition, run the command,

   ```bash
   sas show details
   ```

   This will check the applications’ health endpoint for each application and also show the health status of the stateful services as reported to Consul.

   Here is an excerpt of typical output:

   **Service Status**
   
   - pass cas-acme-default (Success)
   - pass cas-shared-default (Success)
   - pass consul -111-222-7-86 (Agent alive and reachable)
   - pass consul -111-222-7-87 (Agent alive and reachable)
   ... 

   **App Status**
   
   - pass audit (UP)
   - pass authorization (UP)
   - pass datahub (UP)
   - pass entityResolution (UP)
   ...
Post-Installation Tasks

Configure and Onboard Initial Tenant

1. Locate the `tenant.properties` template in the `/home/sas/site-name` directory of the container.
2. Make a copy of the template, naming the new file with the actual name of the initial tenant that will be created.
   
   For example, if you are creating a tenant named `acme`, you could copy the template file as the new tenant file.

   ```
cp tenant.properties acme.properties
   ```
3. Use the text editor to modify the newly named template file. For example, if the name of the tenant file is `acme.properties` then you could open the file to edit it using the following command.

   ```
   vi acme.properties
   ```
4. Enter the tenant properties. Here is an example showing the edited template.

   ```
   #!/bin/bash

   # This is a template file for tenant on-boarding.
   # Copy this file to a new filename and change the properties below.

   # ===== TENANT NAME
   #
   # Description:
   # This is your tenant name.
   # Tenant names must begin with a lowercase letter and
   # may be followed by additional lowercase letters or digits
   # for a maximum of 16 characters.
   # Enter your Tenant name here:
   svi_TENANT_ID="acme"

   # ===== CAS CONTROLLER IP ADDRESS
   #
   # Description:
   # This is the CAS Controller IP address for your tenant.
   # This is NOT the same IP address as that of the base CAS Controller
   # that is used for licensing.
   # Enter the IP address of your CAS Controller here:
   svi_SERVICE_CAS_CONTROLLER_IP="10.10.10.36"
   ```
# ===== CAS WORKER IP ADDRESSES
#
# Description:
#     These are the CAS Worker IP addresses for your tenant.
#     These are NOT the same IP addresses as that of the base CAS workers.
#     An empty array is valid.
#
# Enter the IP addresses of your CAS Workers here:
svi_SERVICE_CAS_WORKER_IP=( "10.10.0.37" )

# ===== TENANT USERNAME
#
# Description:
#     This is the first user added for the new tenant.
#     The tenant first user is a username that can be authenticated
#     using the configured LDAP type.
#     This username may be different for each deployed tenant.
#
# Enter the tenant username:
svi_TENANT_USERNAME="demo"

5 Save the changes.
6 To verify that all Consul services and applications have a passing status, run the following command:

   sas show details

7 Make the initial tenant by running the following command:

   sas make tenant tenant-name.properties

   For example, if your tenant name is acme, then the command would be as follows:

   sas make tenant acme.properties

8 Initialize the first tenant by running the following command:

   sas deploy tenant -i

9 Deploy the initial tenant by running the following command:

   sas deploy tenant tenant-name

10 Check the status of the initial tenant by running the following commands:

   sas show tenant status tenant-name

11 (Optional) Scale the deployment by running the following command:

   cf scale app-name -i n

   where you specify n as the number of instances that should be running when the command completes.
Validating the Deployment

Verify RabbitMQ

To verify that RabbitMQ has been deployed correctly, open a browser and go to the following address:

http://RabbitMQ-IP-address:15672/#/

If the RabbitMQ logon window appears, then RabbitMQ is functioning as expected.

Verify PostgreSQL

Note: This section is applicable only if your order contains PostgreSQL. If it does not, skip this section.

1 Run the following command:

   /opt/sas/viya/home/bin/sas-bootstrap-config kv read "config/application/postgres/password"

2 Note the output of the command. It is the password for the dbmsowner.

3 Connect to the database:

   /opt/sas/viya/home/bin/psql -h IP-address-for-PostgreSQL-database -U dbmsowner postgres

4 When prompted, enter the password that you noted in step 2:

   Password for user dbmsowner:

5 If PostgreSQL is deployed appropriately, you should receive a response like this:

   psql (9.4.9)
   Type "help" for help
   postgres=#

6 To exit the prompt, type \q and press Enter.

Validate Elasticsearch

To determine the health of the deployed Elasticsearch, use the following command:
curl -XGET 'http://IP-address-for-Elasticsearch-master-node:9200/_cluster/health?pretty=true'

Typical output follows:

```json
{
  "cluster_name" : "testcluster",
  "status" : "green",
  "timed_out" : false,
  "number_of_nodes" : 2,
  "number_of_data_nodes" : 3,
  ...
}
```

If the value of status is **green**, the cluster is fully functional. For additional information about Elasticsearch cluster health, refer to https://www.elastic.co/guide/en/elasticsearch/reference/current/cluster-health.html.

---

**Validate Consul**

To validate the installation of Consul:

1. Open a web browser and enter the URL for Consul.
2. To determine the correct URL for Consul, run the following command:

   ```
   sas show info
   ```

   The following output is displayed:

   **SAS Visual Investigator: Info**

   **Product:**
   - Name: SAS Visual Investigator
   - Version: 10.2.2.517.0
   - Build Date: 2017-08-01.0851

   **Configuration:**
   - JSON Configuration: test
   - Org: test
   - Space: test
   - Host: test

   **URLs:**
   - Consul: http://10.10.10.01:8500
   - RabbitMQ: http://10.10.10.02:15672

   **IP Addresses:**
   - CAS Controller: 10.10.10.04
   - CAS Worker: 0
   - Consul: 10.10.10.01
   - Elastic Search
     - Master Nodes: 10.10.10.05
     - Client Nodes: 10.10.10.06
     - Data Nodes: 2
   - Cache Locator: 10.10.10.07
<table>
<thead>
<tr>
<th>Service</th>
<th>IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cache Server</td>
<td>10.10.10.08</td>
</tr>
<tr>
<td>RabbitMQ</td>
<td>10.10.10.02 10.10.10.03</td>
</tr>
<tr>
<td>PostgreSQL</td>
<td>10.10.10.09:9432</td>
</tr>
</tbody>
</table>

From the preceding output, select the URL for Consul. In this example, the URL is `http://10.10.10.01:8500`. 
Completing the Deployment

Further Documentation

You can access *SAS Visual Investigator 10.3: Administrator’s Guide* from within the SAS Visual Investigator application or from the [SAS Visual Investigator documentation](#) page. To access the secure SAS Visual Investigator 10.3 documentation, you must have an access key. The documentation page explains how to contact SAS Technical Support to request the access key.
Appendix A: Troubleshooting

**BOSH**

**Virtual Machine Is Not Created**

The most common problem with a BOSH deployment of a stateful service is the failure to create the Virtual Machine (VM). Generally, the solution is to remove the failed service and then to redeploy again. Follow the procedure:

1. Change to the directory of the failed service.
2. Run the remove.sh script in the directory to remove any traces of the deployment from the BOSH director.
3. Run the deploy.sh script in the directory to redeploy the service.

During the attempted deployment, the RabbitMQ deployment fails and a message is displayed. Here is an example:

```
Error 400007: 'rabbitmq/0 (8e792c02-47c5-495c-9c06-c7b7542bd775)' is not running after update.
Review logs for failed jobs: runrabbit
```

To determine the state of the deployment, run the following command:

```
bosh vms host-rabbitmq-deployment-ha
```

where *host* is the value of the label key in the configuration file. For this example, the test host is used.

After running the command, the output might look like this:

```
bosh vms test-rabbitmq-ha
RSA 1024 bit CA certificates are loaded due to old openssl compatibility
Acting as user 'admin' on deployment 'test-rabbitmq-ha' on 'test'

Director task 10206

Task 10206 done
```
No VMs

To clean up residual information in the Bosh director's database, change to the following directory:

    cd services rabbitmq_ha

To ensure that you are in the correct directory, run the command:

    pwd

WARNINGS:

- The remove.sh script uses options that prevent interaction. Therefore, the command continues processing.
- There is a remove.sh script in each of the directories. Make sure that you are in the correct directory or you might remove everything instead of just the deployment that failed. Each of the remove.sh scripts removes the service or the call for each of the remove.sh scripts in any subdirectories that are in the directory in which it resides. For example, if you are in the services directory, the remove.sh script removes all of the services. If you are in the /home/sas directory, the remove.sh script removes everything. If you are in the /home/services/rabbitmq directory, the remove.sh script removes only the RabbitMQ deployment.

In the /home/services/rabbitmq directory, run the command:

    ./remove.sh

Here is typical output:

    RSA 1024 bit CA certificates are loaded due to old openssl compatibility
    Acting as user 'admin' on deployment 'test-rabbitmq-deployment' on 'test'

    You are going to delete deployment 'test-rabbitmq-deployment'.

    THIS IS A VERY DESTRUCTIVE OPERATION AND IT CANNOT BE UNDONE!

    Director task 40322
    Started deleting instances > rabbitmq/0 (43aab57f-b71c-419b-95c3-9b151cfc99a5). Done (00:00:51)

    Started deleting properties
    Started deleting properties > Destroying deployment. Done (00:00:00)

    Task 40322 done

    Started        2016-01-01 00:49:34 UTC
    Finished       2016-01-01 00:50:25 UTC
    Duration       00:00:51

    Deleted deployment 'test-rabbitmq-deployment'

The warning message THIS IS A VERY DESTRUCTIVE OPERATION AND IT CANNOT BE UNDONE! is present for all BOSH delete deployment commands. There are flags in the remove.sh script that prevent interaction but the message is always displayed.

You must check these additional items in the manifest.yml file and the configuration JSON file:

- The watch time must be 30000-1800000 or higher.
- Canary watch time must be 30000-1800000 or higher.
- Increase the number of workers.

If only one VM failed in a set of deployments like ElasticSearch, it might not be desirable to remove the VMs and then redeploy them. In this situation, use the following command:
In this case, you are prompted to continue with the deployment.

```
bosh delete deployment test-elasticsearch-deployment-client
```

RSA 1024 bit CA certificates are loaded due to old openssl compatibility
Acting as user 'admin' on deployment 'test-elasticsearch-deployment-client' on 'test'

You are going to delete deployment 'test-elasticsearch-deployment-client'.

**THIS IS A VERY DESTRUCTIVE OPERATION AND IT CANNOT BE UNDONE!**

Are you sure? (type 'yes' to continue):

Be sure to enter yes. BOSH continues with the deletion of the deployment.

If you enter y or any other character, the following message is displayed:

```
Canceled deleting deployment
```

**BOSH Deployment Fails to Remove a Canceled Deployment**

A deployment was started and then canceled using Ctrl+C in the terminal session in which the deployment was running. When canceled using the Ctrl+C, you can issue the command bosh tasks and then use the bosh cancel task command to cancel unwanted tasks. Further cleanup can be done by running a remove.sh script. An attempt to remove the canceled deployment using the remove.sh script might result in the following message:

```
./remove.sh
RSA 1024 bit CA certificates are loaded due to old openssl compatibility
Deployment set to '/home/sas/services/consul/manifest.yml'
RSA 1024 bit CA certificates are loaded due to old openssl compatibility
Acting as user 'admin' on deployment 'test-consul' on 'test'

You are going to delete deployment 'test-consul-deployment'.

**THIS IS A VERY DESTRUCTIVE OPERATION AND IT CANNOT BE UNDONE!**
```

Director task 49064
Error 100: Redis lock lock:deployment:test-consul-deployment is acquired by another thread

Task 49064 error

For a more detailed error report, run: bosh task 49064 --debug

Error 100 is caused by a lock in the director's Redis data store that prevents the removal of the deployment. To remove the deployment, wait approximately 10 minutes for the lock to clear and then run the remove.sh script again.
Cloud Foundry Run Time

UI Displays with No Tabs and No Access to the Administration UI

In this case, the initial problem is that the UI comes up but there are no tabs and no access to the Administrative UI.

To check the status of all of the components in Consul, run the following command;

```
sas show details
  pass cas-controller (Success)
pass cas-worker@192.168.5.141 (Success)
pass cas-worker@192.168.5.142 (Success)
pass consul_container (Agent alive and reachable)
pass elasticsearch (TCP connect 10.10.10.05:9200: Success. ElasticSearch cluster green.)
pass postgres (Success)
pass rabbitmq (Success)
```

App Status

```
pass audit (UP)
pass authorization (UP)
pass casManagement (UP)
pass datahub (UP)
pass entityResolution (UP)
pass feature (UP)
pass files (UP)
pass identities (UP)
pass networkAnalytics (UP)
pass SASLogon (UP)
pass SASVisualInvestigator (UP)
pass svi-ai (UP)
warn svi-alert (DOWN)
pass svi-core (UP)
pass svi-sand (UP)
pass svi-transport (UP)
```

This displays svi-alert as a warn and the status would be (DOWN).

Most problems in the run-time deployments are linked to failures in the BOSH services. If only one service is failing (such as the svi-alert service), check the log for that service by running the command:

```
cf logs <host>-svi-alert --recent
```

Where host is the value of the label key in the configuration file. For our example, the test host is used.

```
cf logs test-svi-alert --recent
```

This displays the log data that is in the Loggregator buffer. If a store and display log service is set up, then you can use that to see the log data. When reviewing the log, you might see error messages that look like this:

```
Datahub error: stored object not created
```

This points to a problem in the svi-datahub application. Examining the svi-datahub log, the following is displayed:
The key is the `Access is denied` message and the pointer to the feature service svi-feature. Examining the svi-feature log, you see the following error:

```
cf logs test-svi-feature --recent
2016-09-04T22:49:13.12-0400 [APP/0] OUT debug=0x88bfc1e5:TKCASA_GEN_LICENSE_NOT_LOADED.
```

This main problem is that the CAS server cannot find the license file, which is preventing both CAS and SAS Visual Investigator from being operational.

To fix this problem, correct the `nfs_mounts` section of the JSON configuration file. The incorrect mount point for the license file had been entered so that you need to reenter the mount point.

Here was the original entry:

```
"cas_controller": [
  {
    # Description:
    #   The mount that specifies the location of the SAS license file.
    #   Notes: One mandatory mount point for the SAS license file is required.
    #   The entry in the `fs_file` value MUST be "/opt/cas/license".
    #   This is required in order for CAS to locate the SAS license file.
    #   The file must be renamed to setin90.sas.
    "fs_spec": "test.some.server.com:/files/license",
    "fs_file": "/opt/cas/cloud"
  }
],
```

The new entry would look like this:

```
"cas_controller": [
  {
    # Description:
    #   The mount that specifies the location of the SAS license file.
    #   Notes: One mandatory mount point for the SAS license file is required.
    #   The entry in the `fs_file` value MUST be "/opt/cas/license".
    #   This is required in order for CAS to locate the SAS license file.
    #   The file must be renamed to setin90.sas.
    "fs_spec": "test.some.server.com:/files/license",
    "fs_file": "/opt/cas/license"
  },
]
```

Once the network file system mount is corrected, check the application status and verify that all of the applications are up and running before continuing the deployment process.

**Tenant State Has Not Changed to Onboarded**

The following message might be displayed during onboarding.

```
Tenant your-tenant-name's state is ...onboarding..............
```
To onboard the client:

1. Run the following command to restart datahub:
   
   ```
   cf restart xx-svi-datahub
   ```

2. (Optional) Ensure that the tenant has been initialized:
   
   ```
   cd ~/tenants/your-tenant-name/scripts/bin/tenant ./add-tenant-admin.sh -p cf -t your-tenant-name
   ```

3. Load the users:
   
   **Note**: The users are not LDAP users.
   
   ```
   cd ~/tenants/your-tenant-name/scripts/bin/tenant ./add-users-cf.sh -p cf -t your-tenant-name
   ```

If all processes are complete without errors, you can load the user management spreadsheet.

**svi-feature and svi-network-analytics Fail to Start**

Check to ensure that the license is read.

1. Change to the directory for the CAS license controller:
   
   ```
   cd ~/services/cas
   ```

2. Run the following commands:
   
   ```
   bosh deployment manifest-controller.yml
   bosh ssh
   ```

3. Change to the `/tmp` directory.
   
   ```
   cd /tmp
   ```

4. Run the `cat` command on the `init.info` file:
   
   ```
   cat init.info
   ```

   To locate an error with the NFS mount for the CAS license, search for the phrase "Mounting NFS...".

5. On the BOSH virtual machine, change to the `/opt/cas/license` directory.
   
   ```
   cd /opt/cas/license
   ```

   If the license file is present, check the expiration date to determine whether it has expired.

**Destroy Tenant Command Fails on Drop of the Database**

Run the following SQL code using the pgadmin tool, or submit the following SQL using the psql script.

```
REVOKE CONNECT ON DATABASE tenant_name FROM PUBLIC, dbmsowner;
SELECT
  pg_terminate_backend(pid)
FROM
  pg_stat_activity
WHERE
  -- don't kill my own connection!
  pid <> pg_backend_pid()
-- don't kill the connections to other databases
AND datname = 'tenant_name';
```