Automate the installation

StorageGRID software

NetApp
January 14, 2026

Table of Contents

Automate the installation

Automate the installation (Linux)

You can automate the installation of the StorageGRID host service and the configuration of grid nodes.

About this task

"Linux" refers to a RHEL, Ubuntu, or Debian deployment. For a list of supported versions, see the NetApp Interoperability Matrix Tool (IMT).

Automating the deployment might be useful in any of the following cases:

- You already use a standard orchestration framework, such as Ansible, Puppet, or Chef, to deploy and configure physical or virtual hosts.
- You intend to deploy multiple StorageGRID instances.
- You are deploying a large, complex StorageGRID instance.

The StorageGRID host service is installed by a package and driven by configuration files. You can create the configuration files using one of these methods:

- Create the configuration files interactively during a manual installation.
- Prepare the configuration files ahead of time (or programmatically) to enable automated installation using standard orchestration frameworks, as describe in this article.

  StorageGRID provides optional Python scripts for automating the configuration of StorageGRID appliances and the entire StorageGRID system (the "grid"). You can use these scripts directly, or you can inspect them to learn how to use the StorageGRID installation REST API in grid deployment and configuration tools you develop yourself.

Automate the installation and configuration of the StorageGRID host service

You can automate the installation of the StorageGRID host service using standard orchestration frameworks such as Ansible, Puppet, Chef, Fabric, or SaltStack.

The StorageGRID host service is packaged in a DEB (Ubuntu or Debian) or an RPM (RHEL) and is driven by configuration files that you can prepare ahead of time (or programmatically) to enable automated installation. If you already use a standard orchestration framework to install and configure your Linux deployment, adding StorageGRID to your playbooks or recipes should be straightforward.

You can automate all of the steps for preparing the hosts and deploying virtual grid nodes.

Example Ansible role and playbook

Example Ansible role and playbook are supplied with the installation archive in the `/extras` folder. The Ansible playbook shows how the `storagegrid` role prepares the hosts and installs StorageGRID onto the target servers. You can customize the role or playbook as necessary.

The example playbook does not include the steps required to create network devices before starting the StorageGRID host service. Add these steps before finalizing and using the playbook.

RHEL

For RHEL, the installation tasks in the provided `storagegrid` role example use the `ansible.builtin.dnf` module to perform the installation from the local RPM files or a remote Yum repository. If the module is unavailable or not supported, you might need to edit the appropriate Ansible tasks in the following files to use the `yum` or `ansible.builtin.yum` module:

- `roles/storagegrid/tasks/rhel_install_from_repo.yml`
- `roles/storagegrid/tasks/rhel_install_from_local.yml`

Ubuntu or Debian

For Ubuntu or Debian, the installation tasks in the provided `storagegrid` role example use the `ansible.builtin.apt` module to perform the installation from the local DEB files or a remote apt repository. If the module is unavailable or not supported, you might need to edit the appropriate Ansible tasks in the following files to use the `ansible.builtin.apt` module:

- `roles/storagegrid/tasks/deb_install_from_repo.yml`
- `roles/storagegrid/tasks/deb_install_from_local.yml`

Automate the configuration of StorageGRID

After deploying the grid nodes, you can automate the configuration of the StorageGRID system.

Before you begin

- You know the location of the following files from the installation archive.

Filename	Description
`configure-storagegrid.py`	Python script used to automate the configuration
`configure-storagegrid.sample.json`	Example configuration file for use with the script
`configure-storagegrid.blank.json`	Blank configuration file for use with the script

- You have created a `configure-storagegrid.json` configuration file. To create this file, you can modify the example configuration file (`configure-storagegrid.sample.json`) or the blank configuration file (`configure-storagegrid.blank.json`).

Store the management password and provisioning passphrase from the passwords section of the modified `configure-storagegrid.json` configuration file in a secure location. These passwords are required for installation, expansion, and maintenance procedures. You should also back up the modified `configure-storagegrid.json` configuration file and store it in a secure location.

About this task

You can use the `configure-storagegrid.py` Python script and the `configure-storagegrid.json` configuration file to automate the configuration of your StorageGRID system.

> You can also configure the system using the Grid Manager or the Installation API.

Steps

1. Log in to the Linux machine you are using to run the Python script.
2. Change to the directory where you extracted the installation archive.

   For example:

   ```
   cd StorageGRID-Webscale-version/platform
   ```

   where `platform` is `debs`, `rpms`, or `vsphere`.

3. Run the Python script and use the configuration file you created.

   For example:

   ```
   ./configure-storagegrid.py ./configure-storagegrid.json --start-install
   ```

Result

A Recovery Package `.zip` file is generated during the configuration process, and it is downloaded to the directory where you are running the installation and configuration process. You must back up the Recovery Package file so that you can recover the StorageGRID system if one or more grid nodes fails. For example, copy it to a secure, backed up network location and to a secure cloud storage location.

> The Recovery Package file must be secured because it contains encryption keys and passwords that can be used to obtain data from the StorageGRID system.

If you specified that random passwords should be generated, open the `Passwords.txt` file and look for the passwords required to access your StorageGRID system.

```
#######################################################################
##### The StorageGRID "Recovery Package" has been downloaded as: #####
#####           ./sgws-recovery-package-994078-rev1.zip           #####
#####   Safeguard this file as it will be needed in case of a     #####
#####                 StorageGRID node recovery.                  #####
#######################################################################
```

Your StorageGRID system is installed and configured when a confirmation message is displayed.

```
StorageGRID has been configured and installed.
```

Automate the installation (VMware)

You can use the VMware OVF Tool to automate the deployment of grid nodes. You can also automate the configuration of StorageGRID.

Automate grid node deployment

Use the VMware OVF Tool to automate the deployment of grid nodes.

Before you begin

- You have access to a Linux/Unix system with Bash 3.2 or later.
- You have VMware vSphere with vCenter
- You have VMware OVF Tool installed and correctly configured.
- You know the username and password to access VMware vSphere using the OVF Tool
- You have the sufficient permissions to deploy VMs from OVF files and power them on, and permissions to create additional volumes to attach to the VMs. See the `ovftool` documentation for details.
- You know the virtual infrastructure (VI) URL for the location in vSphere where you want to deploy the StorageGRID virtual machines. This URL will typically be a vApp, or Resource Pool. For example: `vi://vcenter.example.com/vi/sgws`

> ⓘ You can use the VMware `ovftool` utility to determine this value (see the `ovftool` documentation for details).

> ⓘ If you are deploying to a vApp, the virtual machines will not start automatically the first time, and you must power them on manually.

- You have collected all the required information for the deployment configuration file. See Collect information about your deployment environment for information.
- You have access to the following files from the VMware installation archive for StorageGRID:

Filename	Description
`NetApp-SG-version-SHA.vmdk`	The virtual machine disk file that is used as a template for creating grid node virtual machines. **Note:** This file must be in the same folder as the `.ovf` and `.mf` files.

`vsphere-primary-admin.ovf`
 	
`vsphere-primary-admin.mf`
	The Open Virtualization Format template file (`.ovf`) and manifest file (`.mf`) for deploying the primary Admin Node.

`vsphere-non-primary-admin.ovf`
 	
`vsphere-non-primary-admin.mf`
	The template file (`.ovf`) and manifest file (`.mf`) for deploying non-primary Admin Nodes.

Filename	Description

`vsphere-gateway.ovf`
 	
`vsphere-gateway.mf`
	The template file (`.ovf`) and manifest file (`.mf`) for deploying Gateway Nodes.

`vsphere-storage.ovf`
 	
`vsphere-storage.mf`
	The template file (`.ovf`) and manifest file (`.mf`) for deploying virtual machine-based Storage Nodes.

`deploy-vsphere-ovftool.sh`	The Bash shell script used to automate the deployment of virtual grid nodes.
`deploy-vsphere-ovftool-sample.ini`	The example configuration file for use with the `deploy-vsphere-ovftool.sh` script.

Define the configuration file for your deployment

You specify the information needed to deploy virtual grid nodes for StorageGRID in a configuration file, which is used by the `deploy-vsphere-ovftool.sh` Bash script. You can modify an example configuration file, so that you don't have to create the file from scratch.

Steps

1. Make a copy of the example configuration file (`deploy-vsphere-ovftool.sample.ini`). Save the new file as `deploy-vsphere-ovftool.ini` in the same directory as `deploy-vsphere-ovftool.sh`.
2. Open `deploy-vsphere-ovftool.ini`.
3. Enter all of the information required to deploy VMware virtual grid nodes.

   See Configuration file settings for information.

4. When you have entered and verified all of the necessary information, save and close the file.

Configuration file settings

The `deploy-vsphere-ovftool.ini` configuration file contains the settings that are required to deploy virtual grid nodes.

The configuration file first lists global parameters, and then lists node-specific parameters in sections defined by node name. When the file is used:

- *Global parameters* are applied to all grid nodes.
- *Node-specific parameters* override global parameters.

Global parameters

Global parameters are applied to all grid nodes, unless they are overridden by settings in individual sections. Place the parameters that apply to multiple nodes in the global parameter section, and then override these settings as necessary in the sections for individual nodes.

- **OVFTOOL_ARGUMENTS**: You can specify OVFTOOL_ARGUMENTS as global settings, or you can apply arguments individually to specific nodes. For example:

```
OVFTOOL_ARGUMENTS = --powerOn --noSSLVerify --diskMode=eagerZeroedThick
--datastore='datastore_name'
```

You can use the `--powerOffTarget` and `--overwrite` options to shut down and replace existing virtual machines.

> ⚠ You should deploy nodes to different datastores and specify OVFTOOL_ARGUMENTS for each node, instead of globally.

- **SOURCE**: The path to the StorageGRID virtual machine template (`.vmdk`) file and the `.ovf` and `.mf` files for individual grid nodes. This defaults to the current directory.

```
SOURCE = /downloads/StorageGRID-Webscale-version/vsphere
```

- **TARGET**: The VMware vSphere virtual infrastructure (vi) URL for the location where StorageGRID will be deployed. For example:

```
TARGET = vi://vcenter.example.com/vm/sgws
```

- **GRID_NETWORK_CONFIG**: The method used to acquire IP addresses, either STATIC or DHCP. The default is STATIC. If all or most of the nodes use the same method for acquiring IP addresses, you can specify that method here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
GRID_NETWORK_CONFIG = STATIC
```

- **GRID_NETWORK_TARGET**: The name of an existing VMware network to use for the Grid Network. If all or most of the nodes use the same network name, you can specify it here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
GRID_NETWORK_TARGET = SG Admin Network
```

- **GRID_NETWORK_MASK**: The network mask for the Grid Network. If all or most of the nodes use the same network mask, you can specify it here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
GRID_NETWORK_MASK = 255.255.255.0
```

- **GRID_NETWORK_GATEWAY**: The network gateway for the Grid Network. If all or most of the nodes use the same network gateway, you can specify it here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
GRID_NETWORK_GATEWAY = 10.1.0.1
```

- **GRID_NETWORK_MTU**: Optional. The maximum transmission unit (MTU) on the Grid Network. If specified, the value must be between 1280 and 9216. For example:

```
GRID_NETWORK_MTU = 9000
```

If omitted, 1400 is used.

If you want to use jumbo frames, set the MTU to a value suitable for jumbo frames, such as 9000. Otherwise, keep the default value.

> The MTU value of the network must match the value configured on the virtual switch port in vSphere that the node is connected to. Otherwise, network performance issues or packet loss might occur.

> For the best network performance, all nodes should be configured with similar MTU values on their Grid Network interfaces. The **Grid Network MTU mismatch** alert is triggered if there is a significant difference in MTU settings for the Grid Network on individual nodes. The MTU values don't have to be the same for all network types.

- **ADMIN_NETWORK_CONFIG**: The method used to acquire IP addresses, either DISABLED, STATIC, or DHCP. The default is DISABLED. If all or most of the nodes use the same method for acquiring IP addresses, you can specify that method here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
ADMIN_NETWORK_CONFIG = STATIC
```

- **ADMIN_NETWORK_TARGET**: The name of an existing VMware network to use for the Admin Network. This setting is required unless the Admin Network is disabled. If all or most of the nodes use the same network name, you can specify it here. Unlike the Grid Network, all nodes do not need to be connected to the same Admin Network. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
ADMIN_NETWORK_TARGET = SG Admin Network
```

- **ADMIN_NETWORK_MASK**: The network mask for the Admin Network. This setting is required if you are using static IP addressing. If all or most of the nodes use the same network mask, you can specify it here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
ADMIN_NETWORK_MASK = 255.255.255.0
```

- **ADMIN_NETWORK_GATEWAY**: The network gateway for the Admin Network. This setting is required if you are using static IP addressing and you specify external subnets in the ADMIN_NETWORK_ESL

setting. (That is, it is not required if ADMIN_NETWORK_ESL is empty.) If all or most of the nodes use the same network gateway, you can specify it here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
ADMIN_NETWORK_GATEWAY = 10.3.0.1
```

- **ADMIN_NETWORK_ESL**: The external subnet list (routes) for the Admin Network, specified as a comma-separated list of CIDR route destinations. If all or most of the nodes use the same external subnet list, you can specify it here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
ADMIN_NETWORK_ESL = 172.16.0.0/21,172.17.0.0/21
```

- **ADMIN_NETWORK_MTU**: Optional. The maximum transmission unit (MTU) on the Admin Network. Don't specify if ADMIN_NETWORK_CONFIG = DHCP. If specified, the value must be between 1280 and 9216. If omitted, 1400 is used. If you want to use jumbo frames, set the MTU to a value suitable for jumbo frames, such as 9000. Otherwise, keep the default value. If all or most of the nodes use the same MTU for the Admin Network, you can specify it here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
ADMIN_NETWORK_MTU = 8192
```

- **CLIENT_NETWORK_CONFIG**: The method used to acquire IP addresses, either DISABLED, STATIC, or DHCP. The default is DISABLED. If all or most of the nodes use the same method for acquiring IP addresses, you can specify that method here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
CLIENT_NETWORK_CONFIG = STATIC
```

- **CLIENT_NETWORK_TARGET**: The name of an existing VMware network to use for the Client Network. This setting is required unless the Client Network is disabled. If all or most of the nodes use the same network name, you can specify it here. Unlike the Grid Network, all nodes do not need to be connected to the same Client Network. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
CLIENT_NETWORK_TARGET = SG Client Network
```

- **CLIENT_NETWORK_MASK**: The network mask for the Client Network. This setting is required if you are using static IP addressing. If all or most of the nodes use the same network mask, you can specify it here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
CLIENT_NETWORK_MASK = 255.255.255.0
```

- **CLIENT_NETWORK_GATEWAY**: The network gateway for the Client Network. This setting is required if you are using static IP addressing. If all or most of the nodes use the same network gateway, you can specify it here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
CLIENT_NETWORK_GATEWAY = 10.4.0.1
```

- **CLIENT_NETWORK_MTU**: Optional. The maximum transmission unit (MTU) on the Client Network. Don't specify if CLIENT_NETWORK_CONFIG = DHCP. If specified, the value must be between 1280 and 9216. If omitted, 1400 is used. If you want to use jumbo frames, set the MTU to a value suitable for jumbo frames, such as 9000. Otherwise, keep the default value. If all or most of the nodes use the same MTU for the Client Network, you can specify it here. You can then override the global setting by specifying different settings for one or more individual nodes. For example:

```
CLIENT_NETWORK_MTU = 8192
```

- **PORT_REMAP**: Remaps any port used by a node for internal grid node communications or external communications. Remapping ports is necessary if enterprise networking policies restrict one or more ports used by StorageGRID. For the list of ports used by StorageGRID, see internal grid node communications and external communications in Networking guidelines.

  Don't remap the ports you are planning to use to configure load balancer endpoints.

  If only PORT_REMAP is set, the mapping that you specify is used for both inbound and outbound communications. If PORT_REMAP_INBOUND is also specified, PORT_REMAP applies only to outbound communications.

  The format used is: *`network type/protocol/default port used by grid node/new port`*, where network type is grid, admin, or client, and protocol is tcp or udp.

  For example:

```
PORT_REMAP = client/tcp/18082/443
```

  If used alone, this example setting symmetrically maps both inbound and outbound communications for the grid node from port 18082 to port 443. If used in conjunction with PORT_REMAP_INBOUND, this example setting maps outbound communications from port 18082 to port 443.

  You can also remap multiple ports using a comma-separated list.

  For example:

```
PORT_REMAP = client/tcp/18082/443, client/tcp/18083/80
```

- **PORT_REMAP_INBOUND**: Remaps inbound communications for the specified port. If you specify PORT_REMAP_INBOUND but don't specify a value for PORT_REMAP, outbound communications for the port are unchanged.

Don’t remap the ports you are planning to use to configure load balancer endpoints.

The format used is: *network type/protocol/_default port used by grid node/new port*, where network type is grid, admin, or client, and protocol is tcp or udp.

For example:

```
PORT_REMAP_INBOUND = client/tcp/443/18082
```

This example takes traffic that is sent to port 443 to pass an internal firewall and directs it to port 18082, where the grid node is listening for S3 requests.

You can also remap multiple inbound ports using a comma-separated list.

For example:

```
PORT_REMAP_INBOUND = grid/tcp/3022/22, admin/tcp/3022/22
```

- **TEMPORARY_PASSWORD_TYPE**: The type of temporary installation password to be used when accessing the VM console or the StorageGRID Installation API, or using SSH, before the node joins the grid.

  If all or most of the nodes use the same type of temporary installation password, specify the type in the global parameter section. Then, optionally use a different setting for an individual node. For example, if you select **Use Custom Password** globally, you can use **CUSTOM_TEMPORARY_PASSWORD=<password>** to set the password for each node.

  TEMPORARY_PASSWORD_TYPE can be one of the following:

  - **Use node name**: The node name is used as the temporary installation password and provides access to VM console, the StorageGRID Installation API, and SSH.
  - **Disable password**: No temporary installation password will be used. If you need to access the VM to debug installation issues, see Troubleshoot installation issues.
  - **Use custom password**: The value provided with **CUSTOM_TEMPORARY_PASSWORD=<password>** is used as the temporary installation password and provides access to VM console, the StorageGRID Installation API, and SSH.

    Optionally, you can omit the **TEMPORARY_PASSWORD_TYPE** parameter and only specify **CUSTOM_TEMPORARY_PASSWORD=<password>**.

- **CUSTOM_TEMPORARY_PASSWORD=<password>** Optional. The temporary password to use during installation when accessing VM console, the StorageGRID Installation API, and SSH. Ignored if **TEMPORARY_PASSWORD_TYPE** is set to **Use node name** or **Disable password**.

Node-specific parameters

Each node is in its own section of the configuration file. Each node requires the following settings:

- The section head defines the node name that will be displayed in the Grid Manager. You can override that

value by specifying the optional NODE_NAME parameter for the node.

- **NODE_TYPE**: VM_Admin_Node, VM_Storage_Node, or VM_API_Gateway_Node
- **STORAGE_TYPE**: combined, data, or metadata. This optional parameter for storage nodes defaults to combined (data and metadata) if it is not specified. For more information, see Types of Storage Nodes.
- **GRID_NETWORK_IP**: The IP address for the node on the Grid Network.
- **ADMIN_NETWORK_IP**: The IP address for the node on the Admin Network. Required only if the node is attached to the Admin Network and ADMIN_NETWORK_CONFIG is set to STATIC.
- **CLIENT_NETWORK_IP**: The IP address for the node on the Client Network. Required only if the node is attached to the Client Network and CLIENT_NETWORK_CONFIG for this node is set to STATIC.
- **ADMIN_IP**: The IP address for the primary Admin node on the Grid Network. Use the value that you specify as the GRID_NETWORK_IP for the primary Admin Node. If you omit this parameter, the node attempts to discover the primary Admin Node IP using mDNS. For more information, see How grid nodes discover the primary Admin Node.

  The ADMIN_IP parameter is ignored for the primary Admin Node.

- Any parameters that were not set globally. For example, if a node is attached to the Admin Network and you did not specify ADMIN_NETWORK parameters globally, you must specify them for the node.

Primary Admin Node

The following additional settings are required for the primary Admin Node:

- **NODE_TYPE**: VM_Admin_Node
- **ADMIN_ROLE**: Primary

This example entry is for a primary Admin Node that is on all three networks:

```
[DC1-ADM1]
  ADMIN_ROLE = Primary
  NODE_TYPE = VM_Admin_Node
  TEMPORARY_PASSWORD_TYPE = Use custom password
  CUSTOM_TEMPORARY_PASSWORD = Passw0rd

  GRID_NETWORK_IP = 10.1.0.2
  ADMIN_NETWORK_IP = 10.3.0.2
  CLIENT_NETWORK_IP = 10.4.0.2
```

The following additional setting is optional for the primary Admin Node:

- **DISK**: By default, Admin Nodes are assigned two additional 200 GB hard disks for audit and database use. You can increase these settings using the DISK parameter. For example:

  ```
  DISK = INSTANCES=2, CAPACITY=300
  ```

  For Admin nodes, INSTANCES must always equal 2.

Storage Node

The following additional setting is required for Storage Nodes:

- **NODE_TYPE**: VM_Storage_Node

  This example entry is for a Storage Node that is on the Grid and Admin Networks, but not on the Client Network. This node uses the ADMIN_IP setting to specify the primary Admin Node's IP address on the Grid Network.

```
[DC1-S1]
  NODE_TYPE = VM_Storage_Node

  GRID_NETWORK_IP = 10.1.0.3
  ADMIN_NETWORK_IP = 10.3.0.3

  ADMIN_IP = 10.1.0.2
```

  This second example entry is for a Storage Node on a Client Network where the customer's enterprise networking policy states that an S3 client application is only permitted to access the Storage Node using either port 80 or 443. The example configuration file uses PORT_REMAP to enable the Storage Node to send and receive S3 messages on port 443.

```
[DC2-S1]
  NODE_TYPE = VM_Storage_Node

  GRID_NETWORK_IP = 10.1.1.3
  CLIENT_NETWORK_IP = 10.4.1.3
  PORT_REMAP = client/tcp/18082/443

  ADMIN_IP = 10.1.0.2
```

  The last example creates a symmetric remapping for ssh traffic from port 22 to port 3022, but explicitly sets the values for both inbound and outbound traffic.

```
[DC1-S3]
  NODE_TYPE = VM_Storage_Node

  GRID_NETWORK_IP = 10.1.1.3

  PORT_REMAP = grid/tcp/22/3022
  PORT_REMAP_INBOUND = grid/tcp/3022/22

  ADMIN_IP = 10.1.0.2
```

The following additional settings are optional for Storage Nodes:

- **DISK**: By default, Storage Nodes are assigned three 4 TB disks for RangeDB use. You can increase these settings with the DISK parameter. For example:

```
DISK = INSTANCES=16, CAPACITY=4096
```

- **STORAGE_TYPE**: By default, all new Storage Nodes are configured to store both object data and metadata, known as a *combined* Storage Node. You can change the Storage Node type to store only data or metadata with the STORAGE_TYPE parameter. For example:

```
STORAGE_TYPE = data
```

Gateway Node

The following additional setting is required for Gateway Nodes:

- **NODE_TYPE**: VM_API_Gateway

This example entry is for an example Gateway Node on all three networks. In this example, no Client Network parameters were specified in the global section of the configuration file, so they must be specified for the node:

```
[DC1-G1]
  NODE_TYPE = VM_API_Gateway

  GRID_NETWORK_IP = 10.1.0.5
  ADMIN_NETWORK_IP = 10.3.0.5

  CLIENT_NETWORK_CONFIG = STATIC
  CLIENT_NETWORK_TARGET = SG Client Network
  CLIENT_NETWORK_MASK = 255.255.255.0
  CLIENT_NETWORK_GATEWAY = 10.4.0.1
  CLIENT_NETWORK_IP = 10.4.0.5

  ADMIN_IP = 10.1.0.2
```

Non-primary Admin Node

The following additional settings are required for non-primary Admin Nodes:

- **NODE_TYPE**: VM_Admin_Node
- **ADMIN_ROLE**: Non-Primary

This example entry is for a non-primary Admin Node that is not on the Client Network:

```
[DC2-ADM1]
  ADMIN_ROLE = Non-Primary
  NODE_TYPE = VM_Admin_Node

  GRID_NETWORK_TARGET = SG Grid Network
  GRID_NETWORK_IP = 10.1.0.6
  ADMIN_NETWORK_IP = 10.3.0.6

  ADMIN_IP = 10.1.0.2
```

The following additional setting is optional for non-primary Admin Nodes:

- **DISK**: By default, Admin Nodes are assigned two additional 200 GB hard disks for audit and database use. You can increase these settings using the DISK parameter. For example:

  ```
  DISK = INSTANCES=2, CAPACITY=300
  ```

  For Admin nodes, INSTANCES must always equal 2.

Run the Bash script

You can use the `deploy-vsphere-ovftool.sh` Bash script and the deploy-vsphere-ovftool.ini configuration file you modified to automate the deployment of StorageGRID nodes in VMware vSphere.

Before you begin

You have created a deploy-vsphere-ovftool.ini configuration file for your environment.

You can use the help available with the Bash script by entering the help commands (`-h/--help`). For example:

```
./deploy-vsphere-ovftool.sh -h
```

or

```
./deploy-vsphere-ovftool.sh --help
```

Steps

1. Log in to the Linux machine you are using to run the Bash script.
2. Change to the directory where you extracted the installation archive.

   For example:

```
cd StorageGRID-Webscale-version/vsphere
```

3. To deploy all grid nodes, run the Bash script with the appropriate options for your environment.

   For example:

```
./deploy-vsphere-ovftool.sh --username=user --password=pwd ./deploy-
vsphere-ovftool.ini
```

4. If a grid node failed to deploy because of an error, resolve the error and rerun the Bash script for only that node.

   For example:

```
./deploy-vsphere-ovftool.sh --username=user --password=pwd --single
-node="DC1-S3" ./deploy-vsphere-ovftool.ini
```

The deployment is complete when the status for each node is "Passed."

```
Deployment Summary
+------------------------------+----------+----------------------+
| node                         | attempts | status               |
+------------------------------+----------+----------------------+
| DC1-ADM1                     |        1 | Passed               |
| DC1-G1                       |        1 | Passed               |
| DC1-S1                       |        1 | Passed               |
| DC1-S2                       |        1 | Passed               |
| DC1-S3                       |        1 | Passed               |
+------------------------------+----------+----------------------+
```

Automate the configuration of StorageGRID

After deploying the grid nodes, you can automate the configuration of the StorageGRID system.

Before you begin

- You know the location of the following files from the installation archive.

Filename	Description
`configure-storagegrid.py`	Python script used to automate the configuration
`configure-storagegrid.sample.json`	Example configuration file for use with the script

Filename	Description
`configure-storagegrid.blank.json`	Blank configuration file for use with the script

- You have created a `configure-storagegrid.json` configuration file. To create this file, you can modify the example configuration file (`configure-storagegrid.sample.json`) or the blank configuration file (`configure-storagegrid.blank.json`).

> Store the management password and provisioning passphrase from the passwords section of the modified `configure-storagegrid.json` configuration file in a secure location. These passwords are required for installation, expansion, and maintenance procedures. You should also back up the modified `configure-storagegrid.json` configuration file and store it in a secure location.

About this task

You can use the `configure-storagegrid.py` Python script and the `configure-storagegrid.json` grid configuration file to automate the configuration of your StorageGRID system.

> You can also configure the system using the Grid Manager or the Installation API.

Steps

1. Log in to the Linux machine you are using to run the Python script.
2. Change to the directory where you extracted the installation archive.

   For example:

   ```
   cd StorageGRID-Webscale-version/platform
   ```

   where `platform` is debs, rpms, or vsphere.

3. Run the Python script and use the configuration file you created.

   For example:

   ```
   ./configure-storagegrid.py ./configure-storagegrid.json --start-install
   ```

Result

A Recovery Package `.zip` file is generated during the configuration process, and it is downloaded to the directory where you are running the installation and configuration process. You must back up the Recovery Package file so that you can recover the StorageGRID system if one or more grid nodes fails. For example, copy it to a secure, backed up network location and to a secure cloud storage location.

> The Recovery Package file must be secured because it contains encryption keys and passwords that can be used to obtain data from the StorageGRID system.

If you specified that random passwords should be generated, open the `Passwords.txt` file and look for the

passwords required to access your StorageGRID system.

```
#####################################################################
##### The StorageGRID "Recovery Package" has been downloaded as: #####
#####           ./sgws-recovery-package-994078-rev1.zip           #####
#####   Safeguard this file as it will be needed in case of a     #####
#####                 StorageGRID node recovery.                  #####
#####################################################################
```

Your StorageGRID system is installed and configured when a confirmation message is displayed.

```
StorageGRID has been configured and installed.
```

Related information

- Navigate to the Grid Manager
- Installation REST API

Copyright information

Trademark information