



# **Automate appliance installation and configuration (SG100 and SG1000)**

StorageGRID

NetApp  
March 01, 2022

This PDF was generated from <https://docs.netapp.com/us-en/storagegrid-116/sg100-1000/automating-appliance-configuration-using-storagegrid-appliance-installer.html> on March 01, 2022. Always check docs.netapp.com for the latest.

# Table of Contents

- Automate appliance installation and configuration (SG100 and SG1000) . . . . . 1
  - Automate appliance configuration using StorageGRID Appliance Installer . . . . . 1
  - Automate installation and configuration of appliance nodes using configure-sga.py script . . . . . 4
  - Automate configuration of StorageGRID . . . . . 7

# Automate appliance installation and configuration (SG100 and SG1000)

You can automate the installation and configuration of your appliances and configuration of the whole StorageGRID system.

## About this task

Automating installation and configuration can be useful for deploying multiple StorageGRID instances or one large, complex StorageGRID instance.

To automate installation and configuration, use one or more of the following options:

- Create a JSON file that specifies the configuration settings for your appliances. Upload the JSON file using the StorageGRID Appliance Installer.



You can use the same file to configure more than one appliance.

- Use the `StorageGRIDconfigure-sga.py` Python script to automate the configuration of your appliances.
- Use additional Python scripts to configure other components of the whole StorageGRID system (the "grid").



You can use StorageGRID automation Python scripts directly, or you can use them as examples of how to use the StorageGRID Installation REST API in grid deployment and configuration tools you develop yourself. See the instructions for [downloading and extracting the StorageGRID installation files](#).

## Automate appliance configuration using StorageGRID Appliance Installer

You can automate the configuration of an appliance by using a JSON file that contains the configuration information. You upload the file using the StorageGRID Appliance Installer.

### What you'll need

- Your appliance must be on the latest firmware compatible with StorageGRID 11.5 or higher.
- You must be connected to the StorageGRID Appliance Installer on the appliance you are configuring using a [supported web browser](#).

## About this task

You can automate appliance configuration tasks such as configuring the following:

- Grid Network, Admin Network, and Client Network IP addresses
- BMC interface
- Network links
  - Port bond mode
  - Network bond mode

- Link speed

Configuring your appliance using an uploaded JSON file is often more efficient than performing the configuration manually using multiple pages in the StorageGRID Appliance Installer, especially if you have to configure many nodes. You must apply the configuration file for each node one at a time.



Experienced users who want to automate both the installation and configuration of their appliances can [use the configure-sga.py script](#).

## Steps

1. Generate the JSON file using one of the following:

- The [ConfigBuilder application](#).
- The `configure-sga.py` [appliance configuration script](#). You can download the script from the StorageGRID Appliance Installer (**Help > Appliance Configuration Script**).

The node names in the JSON file must follow these requirements:

- Must be a valid hostname containing at least 1 and no more than 32 characters
- Can use letters, numbers, and hyphens
- Cannot start or end with a hyphen
- Cannot contain only numbers



Ensure that the node names (the top-level names) in the JSON file are unique, or you will not be able to configure more than one node using the JSON file.

2. Select **Advanced > Update Appliance Configuration**.

The Update Appliance Configuration page appears.

### Update Appliance Configuration

Use a JSON file to update this appliance's configuration. You can generate the JSON file from the [ConfigBuilder](#) application or from the [appliance configuration script](#).

You might lose your connection if the applied configuration from the JSON file includes "link\_config" and/or "networks" sections. If you are not reconnected within 1 minute, re-enter the URL using one of the other IP addresses assigned to the appliance.

### Upload JSON

JSON  
configuration

Browse

Node name

-- Upload a file ▼

Apply JSON configuration

3. Select the JSON file with the configuration you want to upload.
  - a. Select **Browse**.
  - b. Locate and select the file.
  - c. Select **Open**.

The file is uploaded and validated. When the validation process is complete, the file name is shown next to a green check mark.



You might lose connection to the appliance if the configuration from the JSON file includes sections for "link\_config", "networks", or both. If you are not reconnected within 1 minute, re-enter the appliance URL using one of the other IP addresses assigned to the appliance.

#### Upload JSON

|   |                                       |                        |
|---|---------------------------------------|------------------------|
| JSON configuration                                      | <input type="button" value="Browse"/> | ✓ appliances.orig.json |
| Node name   | -- Select a node ▼                    |                        |
| <input type="button" value="Apply JSON configuration"/> |                                       |                        |

The **Node name** drop down is populated with the top-level node names defined in the JSON file.



If the file is not valid, the file name is shown in red and an error message is displayed in a yellow banner. The invalid file is not applied to the appliance. You can use ConfigBuilder to ensure you have a valid JSON file.

4. Select a node from the list in the **Node name** drop down.

The **Apply JSON configuration** button is enabled.

#### Upload JSON

|   |                                       |                        |
|---|---------------------------------------|------------------------|
| JSON configuration                                      | <input type="button" value="Browse"/> | ✓ appliances.orig.json |
| Node name   | Lab-80-1000 ▼                         |                        |
| <input type="button" value="Apply JSON configuration"/> |                                       |                        |

5. Select **Apply JSON configuration**.

The configuration is applied to the selected node.

# Automate installation and configuration of appliance nodes using `configure-sga.py` script

You can use the `configure-sga.py` script to automate many of the installation and configuration tasks for StorageGRID appliance nodes, including installing and configuring a primary Admin Node. This script can be useful if you have a large number of appliances to configure. You can also use the script to generate a JSON file that contains appliance configuration information.

## What you'll need

- The appliance has been installed in a rack, connected to your networks, and powered on.
- Network links and IP addresses have been configured for the primary Admin Node using the StorageGRID Appliance Installer.
- If you are installing the primary Admin Node, you know its IP address.
- If you are installing and configuring other nodes, the primary Admin Node has been deployed, and you know its IP address.
- For all nodes other than the primary Admin Node, all Grid Network subnets listed on the IP Configuration page of the StorageGRID Appliance Installer have been defined in the Grid Network Subnet List on the primary Admin Node.
- You have downloaded the `configure-sga.py` file. The file is included in the installation archive, or you can access it by clicking **Help > Appliance Installation Script** in the StorageGRID Appliance Installer.



This procedure is for advanced users with some experience using command-line interfaces. Alternatively, you can [use the StorageGRID Appliance Installer to automate the configuration](#).

## Steps

1. Log in to the Linux machine you are using to run the Python script.
2. For general help with the script syntax and to see a list of the available parameters, enter the following:

```
configure-sga.py --help
```

The `configure-sga.py` script uses five subcommands:

- `advanced` for advanced StorageGRID appliance interactions, including BMC configuration and creating a JSON file containing the current configuration of the appliance
- `configure` for configuring the RAID mode, node name, and networking parameters
- `install` for starting a StorageGRID installation
- `monitor` for monitoring a StorageGRID installation
- `reboot` for rebooting the appliance

If you enter a subcommand (`advanced`, `configure`, `install`, `monitor`, or `reboot`) argument followed by the `--help` option you will get a different help text providing more detail on the options available within that subcommand:

```
configure-sga.py subcommand --help
```

3. To confirm the current configuration of the appliance node, enter the following where *SGA-install-ip* is any one of the IP addresses for the appliance node:

```
configure-sga.py configure SGA-INSTALL-IP
```

The results show current IP information for the appliance, including the IP address of the primary Admin Node and information about the Admin, Grid, and Client Networks.

```
Connecting to +https://10.224.2.30:8443+ (Checking version and
connectivity.)
2021/02/25 16:25:11: Performing GET on /api/versions... Received 200
2021/02/25 16:25:11: Performing GET on /api/v2/system-info... Received
200
2021/02/25 16:25:11: Performing GET on /api/v2/admin-connection...
Received 200
2021/02/25 16:25:11: Performing GET on /api/v2/link-config... Received
200
2021/02/25 16:25:11: Performing GET on /api/v2/networks... Received 200
2021/02/25 16:25:11: Performing GET on /api/v2/system-config... Received
200
```

#### StorageGRID Appliance

```
Name:          LAB-SGA-2-30
Node type:     storage
```

#### StorageGRID primary Admin Node

```
IP:           172.16.1.170
State:        unknown
Message:      Initializing...
Version:      Unknown
```

#### Network Link Configuration

##### Link Status

| Link | State | Speed (Gbps) |
|------|-------|--------------|
| ---- | ----- | -----        |
| 1    | Up    | 10           |
| 2    | Up    | 10           |
| 3    | Up    | 10           |
| 4    | Up    | 10           |
| 5    | Up    | 1            |
| 6    | Down  | N/A          |

##### Link Settings

```
Port bond mode:  FIXED
Link speed:      10GBE
```

```
Grid Network:    ENABLED
Bonding mode:    active-backup
```

```

VLAN:          novlan
MAC Addresses: 00:a0:98:59:8e:8a 00:a0:98:59:8e:82

Admin Network:  ENABLED
Bonding mode:   no-bond
MAC Addresses:  00:80:e5:29:70:f4

Client Network:  ENABLED
Bonding mode:    active-backup
VLAN:           novlan
MAC Addresses:   00:a0:98:59:8e:89 00:a0:98:59:8e:81

```

#### Grid Network

```

CIDR:          172.16.2.30/21 (Static)
MAC:           00:A0:98:59:8E:8A
Gateway:       172.16.0.1
Subnets:      172.17.0.0/21
               172.18.0.0/21
               192.168.0.0/21
MTU:           1500

```

#### Admin Network

```

CIDR:          10.224.2.30/21 (Static)
MAC:           00:80:E5:29:70:F4
Gateway:       10.224.0.1
Subnets:      10.0.0.0/8
               172.19.0.0/16
               172.21.0.0/16
MTU:           1500

```

#### Client Network

```

CIDR:          47.47.2.30/21 (Static)
MAC:           00:A0:98:59:8E:89
Gateway:       47.47.0.1
MTU:           2000

```

```

#####
##### If you are satisfied with this configuration, #####
##### execute the script with the "install" sub-command. #####
#####

```

4. If you need to change any of the values in the current configuration, use the `configure` subcommand to update them. For example, if you want to change the IP address that the appliance uses for connection to the primary Admin Node to `172.16.2.99`, enter the following:  
`configure-sga.py configure --admin-ip 172.16.2.99 SGA-INSTALL-IP`
5. If you want to back up the appliance configuration to a JSON file, use the `advanced` and `backup-file`



subcommands. For example, if you want to back up the configuration of an appliance with IP address *SGA-INSTALL-IP* to a file named *appliance-SG1000.json*, enter the following:

```
configure-sga.py advanced --backup-file appliance-SG1000.json SGA-INSTALL-IP
```

The JSON file containing the configuration information is written to the same directory you executed the script from.



Check that the top-level node name in the generated JSON file matches the appliance name. Do not make any changes to this file unless you are an experienced user and have a thorough understanding of StorageGRID APIs.

6. When you are satisfied with the appliance configuration, use the `install` and `monitor` subcommands to install the appliance:

```
configure-sga.py install --monitor SGA-INSTALL-IP
```

7. If you want to reboot the appliance, enter the following:

```
configure-sga.py reboot SGA-INSTALL-IP
```

## Automate configuration of StorageGRID

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

### What you'll need

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

| Filename                                       | Description                                       |
|--|---|
| <code>configure-storagegrid.py</code>          | Python script used to automate the configuration  |
| <code>configure-storagegrid.sample.json</code> | Sample configuration file for use with the script |
| <code>configure-storagegrid.blank.json</code>  | 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 sample configuration file (`configure-storagegrid.sample.json`) or the blank configuration file (`configure-storagegrid.blank.json`).

### 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
```

### After you finish

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, you need to extract 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.
```

## Copyright Information

Copyright © 2022 NetApp, Inc. All rights reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means-graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system- without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

## Trademark Information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.