

---

# SatNOGS Config

*Release 0.13.2+0.g01de024.dirty*

**SatNOGS**

**Sep 10, 2022**



# CONTENTS

<b>1</b>	<b>Table of Contents</b>	<b>3</b>
1.1	User guide	3
1.1.1	Requirements	3
1.1.2	Installation	3
1.1.2.1	Debian	3
1.1.2.2	SatNOGS Config	3
1.1.3	Usage	3
1.1.3.1	Basic	4
1.1.3.2	Advanced	4
1.1.3.3	Show	4
1.1.3.4	Upgrade	4
1.1.3.5	Update	4
1.1.3.6	Reset	4
1.1.3.7	Apply	4
1.2	Developer guide	4
1.2.1	Installation	4
1.2.2	Configuration	5
1.2.3	Extending	5
1.2.4	Modifying the menu	5
1.2.5	Code Quality Assurance	5
1.2.6	Automation	5
1.2.6.1	Environments	5
1.3	Modules reference	6
1.3.1	satnogsconfig	6
1.3.2	satnogsconfig.config	6
1.3.3	satnogsconfig.helpers	7
1.3.4	satnogsconfig.menu	10
<b>2</b>	<b>Indices and tables</b>	<b>13</b>
	<b>Python Module Index</b>	<b>15</b>
	<b>Index</b>	<b>17</b>



SatNOGS Config is a utility for generating the configuration for a Debian-based SatNOGS client system. It is a menu driven configuration tool built on-top of `pythondialog`. The main purpose of this utility is to create the YAML configuration file for SatNOGS Client Ansible which provisions the system. In addition to that, it interacts with `satnogs-setup`, which is a wrapper to bootstrap Ansible, upgrades the system packages using APT and probes the hardware to show the user a more limited set of configuration options.



## TABLE OF CONTENTS

### 1.1 User guide

#### 1.1.1 Requirements

- Python 3.4+
- libapt 1.8+ Python bindings
- lsb-release 10+

#### 1.1.2 Installation

##### 1.1.2.1 Debian

To install the required dependencies in Debian Bullseye run:

```
$ apt-get install python3-apt lsb-release
```

##### 1.1.2.2 SatNOGS Config

To install SatNOGS Config run:

```
$ pip install satnogs-config
```

This will install a console script called `satnogs-config`.

#### 1.1.3 Usage

To execute the script, run it on the command line:

```
$ satnogs-config
```

The main menu will be shown

### 1.1.3.1 Basic

The **Basic** sub-menu contains the minimum required configuration options for provisioning the SatNOGS client system.

### 1.1.3.2 Advanced

The **Advanced** sub-menu contains a larger set of configuration options. It is intended for advanced users and contains options categorized by component.

### 1.1.3.3 Show

The **Show** option displays the current configuration file.

### 1.1.3.4 Upgrade

The **Upgrade** option calls APT utilities to upgrade the distribution packages.

### 1.1.3.5 Update

The **Update** option works in conjunction with `satnogs-setup` to re-bootstrap Ansible and update the configuration tools.

### 1.1.3.6 Reset

The **Reset** option resets the configuration file generated by the utility, effectively removing all the configuration options set by the user.

### 1.1.3.7 Apply

The **Apply** button runs SatNOGS Client Ansible which provisions Debian to become a SatNOGS client system.

## 1.2 Developer guide

### 1.2.1 Installation

To install the required dependencies in Debian Bullseye run:

```
$ apt-get install python3-apt lsb-release
```

It is recommended to install SatNOGS Config in a virtualenv. The virtualenv needs to have access to system Python bindings. To create the virtualenv, you can use `virtualenvwrapper`. On the first time, create the virtualenv by running:

```
$ mkvirtualenv --system-site-packages -a . satnogs-config
```

To activate the virtualenv after it is created run:



```
$ workon satnogs-config
```

To install SatNOGS Config for development run in the project root directory:

```
$ pip install -e .[dev]
```

## 1.2.2 Configuration

This project uses `python-dotenv`. Configuration of `satnogsconfig/settings.py` can be overridden by setting the respective environment variables or an `.env` file placed on the project root directory.

## 1.2.3 Extending

SatNOGS Config functionality can be extended by implementing additional helpers. The helpers are used to enhance menu functionality beyond the core function of generating a YAML file for Ansible.

## 1.2.4 Modifying the menu

The menu itself is also expressed in YAML format. The menu structure is defined in `menu.yml` file and shipped along with the package.

## 1.2.5 Code Quality Assurance

The following code quality assurance tools are used in this project:

- flake8
- isort
- yapf
- pylint

## 1.2.6 Automation

`tox` is used to automate development tasks. To execute the default list of tasks run:

```
$ tox
```

### 1.2.6.1 Environments

The following `tox` environments are available:

- flake8 - Check code for common errors, coding style and complexity
- isort - Check code for correct imports order
- isort-apply - Sort imports
- yapf - Check code for correct formatting
- yapf-apply - Reformat source code

- `pylint` - Execute static code analysis
- `build` - Build source and binary distributions
- `upload` - Upload source and binary distributions to PyPI
- `docs` - Build documentation
- `deps` - Check for broken dependencies

To execute a single environment run:

```
$ tox -e <environment>
```

## 1.3 Modules reference

### 1.3.1 `satnogsconfig`

SatNOGS Config module initialization

`satnogsconfig.main()`

SatNOGS Setup utility

### 1.3.2 `satnogsconfig.config`

Configuration module

**class** `satnogsconfig.config.Config(filename)`

Manage configuration file

**Parameters**

**filename** (*str*) – File path of configuration

**`_load_config()`**

Load and parse YAML configuration

**Returns**

Configuration dictionary

**Return type**

dict or NoneType

**`clear_config()`**

Clear configuration file

**`dump_config(to_file=False)`**

Dump configuration in YAML format

**Parameters**

**to\_file** (*bool, optional*) – Dump to file

**Returns**

YAML configuration

**Return type**

str

**get\_variable**(*variable*)

Get variable value from configuration

**Parameters**

**variable** (*str*) – Variable to get the value

**Returns**

Value of variable

**Return type**

str or bool or NoneType

**set\_variable**(*variable, value*)

Set variable value in configuration

**Parameters**

- **variable** (*str*) – Variable to set the value
- **value** (*str or bool or NoneType*) – Value of variable

### 1.3.3 satnogsconfig.helpers

Helpers module

**class** satnogsconfig.helpers.**Ansible**(*ansible\_dir*)

Call Ansible playbooks

**pull**(*playbooks, url, branch=None, tags=None, extra\_args=None*)

Pull and run Ansible playbook

**Parameters**

- **playbooks** (*list*) – List of playbooks
- **url** (*str*) – Git URL to pull playbooks
- **branch** (*str, optional*) – Git branch to pull playbooks
- **tags** (*list, optional*) – List of tags
- **extra\_args** (*list, optional*) – List of extra arguments to pass to Ansible

**Returns**

Whether Ansible execution succeeded

**Return type**

bool

**run**(*playbooks, tags=None, extra\_args=None*)

Run Ansible playbook

**Parameters**

- **playbooks** (*list*) – List of playbooks
- **tags** (*list, optional*) – List of tags
- **extra\_args** (*list, optional*) – List of extra arguments to pass to Ansible

**Returns**

Whether Ansible execution succeeded

**Return type**

bool

**class** satnogsconfig.helpers.GrSatnogs

Get and set gr-satnogs configuration

**property** gr\_satnogs\_version

Get gr-satnogs version

**probe\_soapysdr()**

Probe SoapySDR hardware

**class** satnogsconfig.helpers.SatnogsSetup

Interact with satnogs-setup

**property** gnuradio\_version

Get installed gnuradio version

**Returns**

Version of gr-soapy

**Return type**

str

**property** gr\_satnogs\_version

Get installed gr-satnogs version

**Returns**

Version of gr-satnogs

**Return type**

str

**property** gr\_soapy\_version

Get installed gr-soapy version

**Returns**

Version of gr-soapy

**Return type**

str

**property** is\_applied

Check whether configuration has been applied

**Returns**

Whether configuration has been applied

**Return type**

bool

**static** restart(*boot=False*)

Restart satnogs-setup script

**Parameters**

**boot** (*bool*, *optional*) – Whether to bootstrap or not

**property** satnogs\_client\_ansible\_version

Get installed SatNOGS Client Ansible version

**Returns**

Version of SatNOGS Client Ansible

**Return type**

str

**property satnogs\_client\_version**

Get installed SatNOGS Client version

**Returns**

Version of SatNOGS Client

**Return type**

str

**property satnogs\_flowgraphs\_version**

Get installed satnogs-flowgraphs version

**Returns**

Version of satnogs-flowgraphs

**Return type**

str

**property tags**

Get satnogs-setup tags

**Returns**

Set of tags

**Return type**

set

**upgrade\_system()**

Upgrade system packages

**class** satnogsconfig.helpers.**Support**(*config, satnogs\_setup, ansible*)

Create support information to be used for reporting bugs

**dump**(\*args, \*\*kwargs)

Dump support information

**Returns**

JSON dump of support information

**Return type**

str

**property info**

Support information

**Returns**

Support information dictionary

**Return type**

dict

### 1.3.4 satnogsconfig.menu

Menu module

**class** satnogsconfig.menu.**Menu**(*menu*, *config*, *backtitle=None*)

Show a menu structure based on dialog

**Parameters**

- **menu** (*dict*) – Menu dictionary
- **config** (*dict*) – Configuration dictionary
- **backtitle** (*str*, *optional*) – Default dialog backtitle

**\_apply**(*menu*)

Request setup from configuration management tool

**\_configbox**(*menu*)

Show scrollbox for viewing configuration

**Parameters**

**menu** (*dict*) – Menu dictionary

**\_exit**(*menu*)

Exit the utility

**Parameters**

**menu** (*dict*) – Menu dictionary

**\_get\_common\_options**(*menu*)

Get dialog common options

**Parameters**

**menu** (*dict*) – Menu dictionary

**Returns**

Common option dictionary

**Return type**

dict

**\_msgbox**(*menu*)

Show msgbox

**Parameters**

**menu** (*dict*) – Menu dictionary

**\_reboot**(*menu*)

Reboot system

**Parameters**

**menu** (*dict*) – Menu dictionary

**\_resetyesno**(*menu*)

Reset configuration

**Parameters**

**menu** (*dict*) – Menu dictionary

**\_set\_default\_backtitle**(*backtitle=None*)

Set backtitle of menu

**Parameters**

**backtitle** (*str*) – Menu backtitle

**\_submenu**(*menu*)

Show submenu

**Parameters**

**menu** (*dict*) – Menu dictionary

**\_support**(*\_*)

Show support information

**\_update**(*\_*)

Request tool update

**\_update\_stack**(*menu, response*)

Update stack based on dialog responses

**Parameters**

- **menu** (*dict*) – Menu dictionary
- **response** (*str*) – Dialog response

**\_variablebox**(*menu*)

Show inputbox for setting a variable

**Parameters**

**menu** (*dict*) – Menu dictionary

**\_variableyesno**(*menu*)

Show boolean selection for setting a variable

**Parameters**

**menu** (*dict*) – Menu dictionary

**property backtitle**

Get default backtitle

**Returns**

Background title

**Return type**

str

**show**(*\_*)

Show dialog menu structure

satnogsconfig.menu.**\_clear\_screen**(*\_*)

Clear screen

satnogsconfig.menu.**\_get\_variables**(*menu, name=None, mandatory=False*)

Get all menu variable items

**Parameters**

- **menu** (*dict*) – Menu dictionary
- **name** (*str, optional*) – Name of menu item

- **mandatory** (*bool*, *optional*) – Return only mandatory variables

**Returns**

Menu variables dictionary

**Return type**

dict

satnogsconfig.menu.\_load\_menu(*file*)

Load menu structure from YAML file

**Parameters**

**file** (*file*) – Menu file stream

**Returns**

Menu dictionary

**Return type**

dict

satnogsconfig.menu.\_reboot()

Reboot system



## INDICES AND TABLES

- genindex
- modindex
- search



## PYTHON MODULE INDEX

### S

satnogsconfig, 6  
satnogsconfig.config, 6  
satnogsconfig.helpers, 7  
satnogsconfig.menu, 10



## Symbols

\_apply() (*satnogsconfig.menu.Menu method*), 10  
 \_clear\_screen() (*in module satnogsconfig.menu*), 11  
 \_configbox() (*satnogsconfig.menu.Menu method*), 10  
 \_exit() (*satnogsconfig.menu.Menu method*), 10  
 \_get\_common\_options() (*satnogsconfig.menu.Menu method*), 10  
 \_get\_variables() (*in module satnogsconfig.menu*), 11  
 \_load\_config() (*satnogsconfig.config.Config method*), 6  
 \_load\_menu() (*in module satnogsconfig.menu*), 12  
 \_msgbox() (*satnogsconfig.menu.Menu method*), 10  
 \_reboot() (*in module satnogsconfig.menu*), 12  
 \_reboot() (*satnogsconfig.menu.Menu method*), 10  
 \_resetyesno() (*satnogsconfig.menu.Menu method*), 10  
 \_set\_default\_backtitle() (*satnogsconfig.menu.Menu method*), 10  
 \_submenu() (*satnogsconfig.menu.Menu method*), 11  
 \_support() (*satnogsconfig.menu.Menu method*), 11  
 \_update() (*satnogsconfig.menu.Menu method*), 11  
 \_update\_stack() (*satnogsconfig.menu.Menu method*), 11  
 \_variablebox() (*satnogsconfig.menu.Menu method*), 11  
 \_variableyesno() (*satnogsconfig.menu.Menu method*), 11

## A

Ansible (*class in satnogsconfig.helpers*), 7

## B

backtitle (*satnogsconfig.menu.Menu property*), 11

## C

clear\_config() (*satnogsconfig.config.Config method*), 6

Config (*class in satnogsconfig.config*), 6

## D

dump() (*satnogsconfig.helpers.Support method*), 9

dump\_config() (*satnogsconfig.config.Config method*), 6

## G

get\_variable() (*satnogsconfig.config.Config method*), 6

gnuradio\_version (*satnogsconfig.helpers.SatnogsSetup property*), 8

gr\_satnogs\_version (*satnogsconfig.helpers.GrSatnogs property*), 8

gr\_satnogs\_version (*satnogsconfig.helpers.SatnogsSetup property*), 8

gr\_soapy\_version (*satnogsconfig.helpers.SatnogsSetup property*), 8

GrSatnogs (*class in satnogsconfig.helpers*), 8

## I

info (*satnogsconfig.helpers.Support property*), 9

is\_applied (*satnogsconfig.helpers.SatnogsSetup property*), 8

## M

main() (*in module satnogsconfig*), 6

Menu (*class in satnogsconfig.menu*), 10

module

satnogsconfig, 6

satnogsconfig.config, 6

satnogsconfig.helpers, 7

satnogsconfig.menu, 10

## P

probe\_soapysdr() (*satnogsconfig.helpers.GrSatnogs method*), 8

pull() (*satnogsconfig.helpers.Ansible method*), 7

## R

restart() (*satnogsconfig.helpers.SatnogsSetup static method*), 8

run() (*satnogsconfig.helpers.Ansible method*), 7

## S

satnogs\_client\_ansible\_version (*satnogsconfig.helpers.SatnogsSetup property*), 8

satnogs\_client\_version (*satnogsconfig.helpers.SatnogsSetup* property), 9  
satnogs\_flowgraphs\_version (*satnogsconfig.helpers.SatnogsSetup* property), 9  
satnogsconfig  
  module, 6  
satnogsconfig.config  
  module, 6  
satnogsconfig.helpers  
  module, 7  
satnogsconfig.menu  
  module, 10  
SatnogsSetup (*class in satnogsconfig.helpers*), 8  
set\_variable() (*satnogsconfig.config.Config* method),  
  7  
show() (*satnogsconfig.menu.Menu* method), 11  
Support (*class in satnogsconfig.helpers*), 9

## T

tags (*satnogsconfig.helpers.SatnogsSetup* property), 9

## U

upgrade\_system() (*satnogsconfig.helpers.SatnogsSetup* method), 9