
CadVlan Documentation

Release

Author

December 21, 2018

1	About Globo NetworkAPI WebUI	3
1.1	Description	3
1.2	Features	3
1.3	Architecture	4
1.4	Related Projects	4
2	Installing GloboNetworkAPI WebUI	5
2.1	Using pre-configured VM	5
2.2	Installing from scratch	5
2.3	Create a specific User/Group	5
2.4	Download Code	5
2.5	Create a VirtualEnv	6
2.6	Install Dependencies	6
2.7	Install Memcached	6
2.8	HTTP Server Configuration	6
2.9	Install CVS	7
2.10	Test installation	7
2.11	LDAP Server Configuration	7
2.12	Working with Documentation	7
3	GloboNetworkAPI WebUI package	9
3.1	Subpackages	9
3.2	Submodules	18
3.3	CadVlan.forms module	18
3.4	CadVlan.manage module	18
3.5	CadVlan.messages module	18
3.6	CadVlan.permissions module	18
3.7	CadVlan.processExceptionMiddleware module	18
3.8	CadVlan.settings module	18
3.9	CadVlan.templates module	18
3.10	CadVlan.urls module	18
3.11	Module contents	18
4	Indices and tables	19
	Python Module Index	21

Contents:

About Globo NetworkAPI WebUI

Description

Globo NetworkAPI WebUI is a Front End tool to manage and automate networking resources (routers, switches and load balancers) and document logical and physical networking.

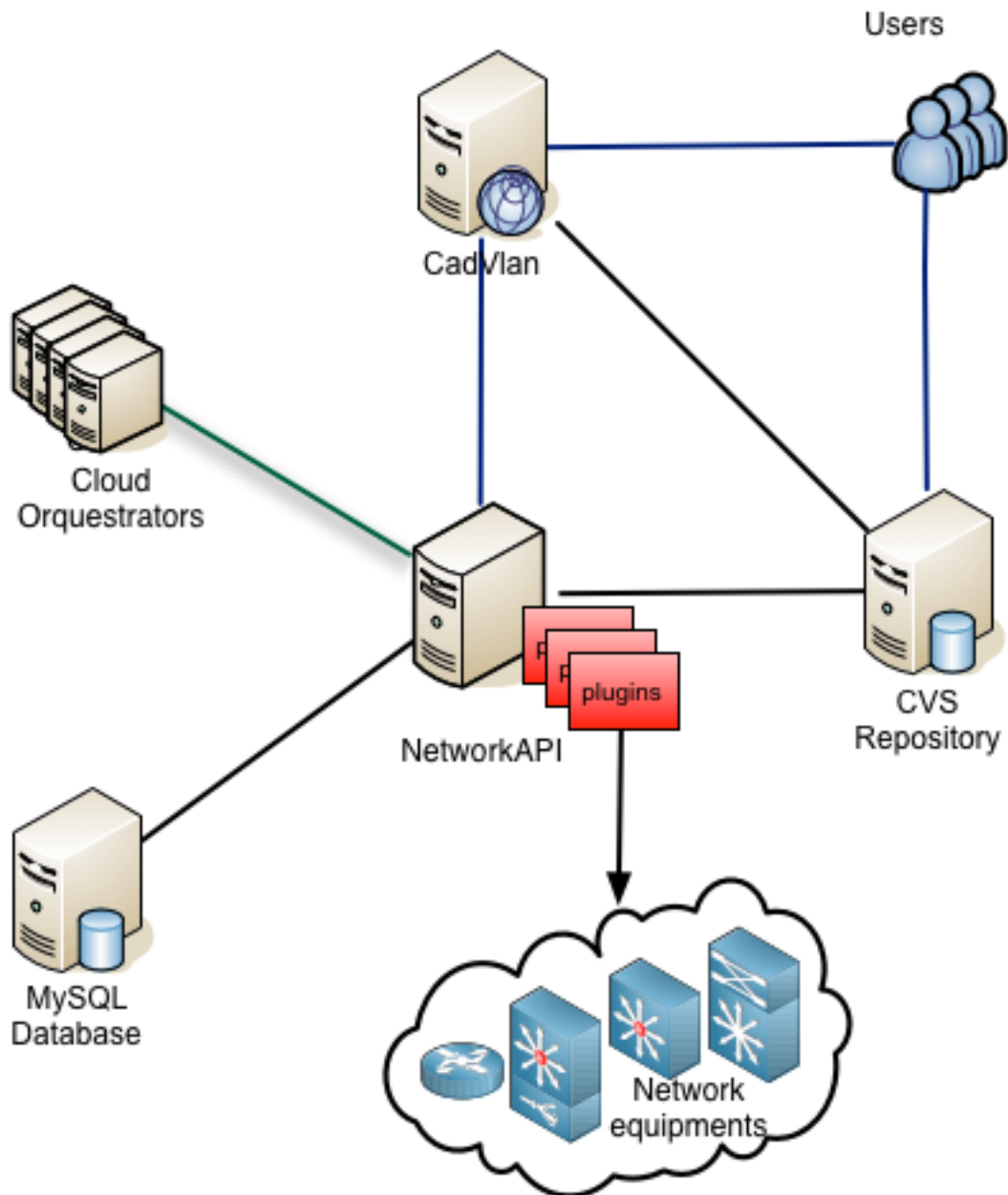
They were created to be vendor agnostic and to support different orchestrators and environments without losing the centralized view of all network resources allocated.

It was not created to be an inventory database, so it does not have CMDB functionalities.

Features

- LDAP authentication
- Supports cabling documentation (including patch-panels/DIO's)
- Separated Layer 2 and Layer 3 documentation (vlan/network)
- IPv4 and IPv6 support
- Automatic allocation of Vlan's, Networks and IP's
- ACL (access control list) automation (documentation/versioning/applying)
- Load-Balancer support
- Automated deploy of allocated resources on switches, routers and load balancers
- Load balancers management
- Expandable plugins for automating configuration

Architecture



Related Projects

[Globo NetworkAPI.](#)

[Globo NetworkAPI Python Client.](#)

[Globo NetworkAPI Python Java.](#)

Installing GloboNetworkAPI WebUI

Using pre-configured VM

In order to use the pre-configured VM you need to have *vagrant* <<https://www.vagrantup.com/downloads.html>> and *VirtualBox* <<https://www.virtualbox.org/wiki/Downloads>> installed in your machine.

After that, go to the directory you want to install and do:

```
git clone https://github.com/globocom/GloboNetworkAPI-WebUI
cd GloboNetworkAPI-WebUI
vagrant plugin install vagrant-omnibus
vagrant up
```

After this you'll have the WebUI running on <http://10.0.0.3:8080/>

Installing from scratch

Following examples were based on CentOS 7.0.1406 installation.

All root passwords were configured to “default”.

Create a specific User/Group

```
useradd -m -U webui
passwd webui
visudo
    webui          ALL= (ALL)          ALL

sudo mkdir /opt/app/
sudo chmod 777 /opt/app/
```

Download Code

Download GloboNetworkAPI code from [Globocom GitHub](#).

In this example we are downloading code to `/opt/app/`:

```
sudo yum install git
cd /opt/app/
git clone https://github.com/globocom/GloboNetworkAPI-WebUI
```

We are exporting this variable below to better document the install process:

```
export WEBUI_FOLDER=/opt/app/GloboNetworkAPI-WebUI/
echo "export WEBUI_FOLDER=/opt/app/GloboNetworkAPI-WebUI/" >> ~/.bashrc
```

Create a VirtualEnv

```
sudo yum install python-virtualenv
sudo easy_install pip
virtualenv ~/virtualenvs/webui_env
source ~/virtualenvs/webui_env/bin/activate
echo "source ~/virtualenvs/webui_env/bin/activate" >> ~/.bashrc
```

Install Dependencies

You will need the following packages in order to install the next python packages via pip:

```
sudo yum install gcc
yum install openldap-devel
```

Install the packages listed on `$WEBUI_FOLDER/requirements.txt` file:

```
pip install -r $WEBUI_FOLDER/requirements.txt
```

Create a `sitecustomize.py` inside your `/path/to/lib/python2.X` folder with the following content:

```
import sys
sys.setdefaultencoding('utf-8')

echo -e "import sys\nsys.setdefaultencoding('utf-8')\n" > ~/virtualenvs/webui_env/lib/python2.7/sitecustomize.py
```

Install Memcached

You can run memcached locally or you can set file variable `CACHES{default{LOCATION}}` to use a remote memcached farm in file `$WEBUI_FOLDER/settings.py`.

In case you need to run locally:

```
sudo yum install memcached
sudo systemctl start memcached
sudo systemctl enable memcached
```

HTTP Server Configuration

For a better performance, install Green Unicorn to run NetworkAPI.

```
pip install gunicorn
```

There is no need to install a nginx or apache to proxy pass the requests, once there is no static files in the API.

Edit `$WEBUI_FOLDER/gunicorn.conf.py` to use your log files location and [user preferentes](#) and run gunicorn:

```
cd $WEBUI_FOLDER/  
gunicorn wsgi:application
```

Install CVS

You will need the JDK software in order to install CVS

```
sudo yum install java-x-openjdk.x86_64
```

Set the `JAVA_HOME` variable with the path to java folder

```
JAVA_HOME=/etc/java-x-openjdk.x86_64
```

Download CVS software

```
sudo yum install cvs
```

Uncomment and set the variable `CVS_JAVA_HOME` in `cvs.sh`

Run cvs

Put the following command in `.bashrc`

```
export CVSROOT=:pserver:<user>@<host>:/<path>
```

Folder: `GloboNetworkAPI-WebUI/Cadvlan/ACLS:`

```
cvs checkout <repo>/ACLS
```

Test installation

Try to access the root location of the API:

```
http://your_location:8080/
```

This should take you the login page.

LDAP Server Configuration

If you want to use LDAP authentication, configure the following variables in `FILE:`

```
!TODO
```

Working with Documentation

If you want to generate documentation, you need the following python modules installed:

```
pip install sphinx==1.2.2
pip install sphinx-rtd-theme==0.1.6
pip install pytest==2.2.4
```

GloboNetworkAPI WebUI package

Subpackages

CadVlan.AccessType package

Submodules

CadVlan.AccessType.forms module

CadVlan.AccessType.views module

Module contents

CadVlan.Acl package

Submodules

CadVlan.Acl.acl module

CadVlan.Acl.forms module

CadVlan.Acl.views module

Module contents

CadVlan.Auth package

Submodules

CadVlan.Auth.AuthSession module

CadVlan.Auth.business module

CadVlan.Auth.forms module

CadVlan.Auth.models module

class CadVlan.Auth.models.**User** (*id, user, nome, email, pwd, permission, ativo, user_ldap*)

Bases: object

get_active()

get_email()

get_id()

get_name()

get_password()

get_permission()

get_user_ldap()

get_username()

has_perm (*permission, write=None, read=None*)

Validates that the user has access permission

Parameters

- **permission** – access permission to be validated
- **write** – permission be write
- **read** – permission be read

has_perm_menu (*write=None, read=None*)

Validates that the user has access permission in top menu

Parameters

- **write** – permission be write
- **read** – permission be read

set_password (*password*)

CadVlan.Auth.views module

Module contents

CadVlan.BlockRules package

Submodules

CadVlan.BlockRules.forms module

CadVlan.BlockRules.views module

Module contents

CadVlan.Environment package

Submodules

CadVlan.Environment.forms module

CadVlan.Environment.views module

Module contents

CadVlan.EnvironmentVip package

Submodules

CadVlan.EnvironmentVip.business module

CadVlan.EnvironmentVip.business.**get_ambiente_by_id**(*ambientes*, *id_ambiente*)

CadVlan.EnvironmentVip.form module

CadVlan.EnvironmentVip.views module

Module contents

CadVlan.EquipAccess package

Submodules

CadVlan.EquipAccess.forms module

CadVlan.EquipAccess.views module

Module contents

CadVlan.EquipGroup package

Submodules

CadVlan.EquipGroup.forms module

CadVlan.EquipGroup.views module

Module contents

CadVlan.EquipInterface package

Submodules

CadVlan.EquipInterface.business module

CadVlan.EquipInterface.forms module

CadVlan.EquipInterface.views module

Module contents

CadVlan.EquipScript package

Submodules

CadVlan.EquipScript.forms module

CadVlan.EquipScript.views module

Module contents

CadVlan.Equipment package

Submodules

CadVlan.Equipment.business module

CadVlan.Equipment.forms module

CadVlan.Equipment.views module

Module contents

CadVlan.EquipmentType package

Submodules

CadVlan.EquipmentType.forms module

CadVlan.EquipmentType.views module

Module contents

CadVlan.EventLog package

Submodules

CadVlan.EventLog.forms module

CadVlan.EventLog.views module

Module contents

CadVlan.Filter package

Submodules

CadVlan.Filter.form module

CadVlan.Filter.views module

Module contents

CadVlan.GroupEquip package

Submodules

CadVlan.GroupEquip.business module

`CadVlan.GroupEquip.business.to_complete_list` (*user_list*, *group_list*)

CadVlan.GroupEquip.forms module

CadVlan.GroupEquip.views module

Module contents

CadVlan.GroupUser package

Submodules

CadVlan.GroupUser.forms module

CadVlan.GroupUser.models module

CadVlan.GroupUser.views module

Module contents

CadVlan.HealthcheckExpect package

Submodules

CadVlan.HealthcheckExpect.forms module

CadVlan.HealthcheckExpect.views module

Module contents

CadVlan.Ldap package

Submodules

CadVlan.Ldap.form module

CadVlan.Ldap.model module

CadVlan.Ldap.views module

Module contents

CadVlan.Net package

Submodules

CadVlan.Net.business module

`CadVlan.Net.business.is_valid_ipv4(ip)`
Validates IPv4 addresses.

`CadVlan.Net.business.is_valid_ipv6(ip)`
Validates IPv6 addresses.

CadVlan.Net.forms module

CadVlan.Net.views module

Module contents

CadVlan.NetworkType package

Submodules

CadVlan.NetworkType.forms module

CadVlan.NetworkType.views module

Module contents

CadVlan.OptionVip package

Submodules

CadVlan.OptionVip.forms module

CadVlan.OptionVip.views module

Module contents

CadVlan.Script package

Submodules

CadVlan.Script.forms module

CadVlan.Script.views module

Module contents

CadVlan.ScriptType package

Submodules

CadVlan.ScriptType.forms module

CadVlan.ScriptType.views module

Module contents

CadVlan.User package

Submodules

CadVlan.User.forms module

CadVlan.User.views module

Module contents

CadVlan.UserGroup package

Submodules

CadVlan.UserGroup.forms module

CadVlan.UserGroup.views module

Module contents

CadVlan.Util package

Subpackages

CadVlan.Util.converters package

Submodules

CadVlan.Util.converters.util module

CadVlan.Util.converters.util.**replace_id_to_name** (*main_list*, *type_list*, *str_main_id*,
str_type_id, *str_prop*)

Return list replacing type id to type name

CadVlan.Util.converters.util.**split_to_array** (*string*, *sep*=';')

Returns an array of strings separated by regex

Module contents

CadVlan.Util.extends package

Submodules

CadVlan.Util.extends.formsets module

Module contents

CadVlan.Util.forms package

Submodules

CadVlan.Util.forms.customRenderer module

CadVlan.Util.forms.decorators module

CadVlan.Util.forms.validators module

Module contents

CadVlan.Util.templatetags package

Submodules

CadVlan.Util.templatetags.util module

Module contents

Submodules

CadVlan.Util.Decorators module

CadVlan.Util.Enum module

```
class CadVlan.Util.Enum.Enum
    Bases: set
```

CadVlan.Util.Json module

CadVlan.Util.cvs module

CadVlan.Util.file module

```
class CadVlan.Util.file.File
```

```
    classmethod create (file_name)
        Creating File
```

Parameters **file_name** – File name

Raises `FileError` Failed to creating file

classmethod `read` (*file_name*)

Reading File

Parameters `file_name` – File name

Raises `FileError` Failed to reading file

classmethod `remove` (*file_name*)

Removing File

Parameters `file_name` – File name

Raises `FileError` Failed to removing file

classmethod `write` (*file_name*, *content*)

Writing File

Parameters

- `file_name` – File name
- `content` – File content

Raises `FileError` Failed to writing file

exception `CadVlan.Util.file.FileError` (*error*)

Bases: `exceptions.Exception`

CadVlan.Util.models module

CadVlan.Util.shortcuts module

CadVlan.Util.utility module

Module contents

CadVlan.VipRequest package

Submodules

CadVlan.VipRequest.encryption module

CadVlan.VipRequest.forms module

CadVlan.VipRequest.views module

Module contents

CadVlan.Vlan package

Submodules

CadVlan.Vlan.business module

CadVlan.Vlan.forms module

CadVlan.Vlan.views module

Module contents

Submodules

CadVlan.forms module

CadVlan.manage module

CadVlan.messages module

CadVlan.permissions module

```
CadVlan.permissions.ACL_APPLY = 'aplicar_acl'
```

Pool Permissions

CadVlan.processExceptionMiddleware module

```
class CadVlan.processExceptionMiddleware.LoggingMiddleware
    Bases: object

    process_exception(request, exception)
        HIDE PASSWORD VALUES
```

CadVlan.settings module

CadVlan.templates module

CadVlan.urls module

Module contents

Indices and tables

- *genindex*
- *modindex*
- *search*

C

CadVlan, 18
CadVlan.AccessType, 9
CadVlan.Acl, 9
CadVlan.Auth, 10
CadVlan.Auth.models, 10
CadVlan.BlockRules, 10
CadVlan.Environment, 11
CadVlan.EnvironmentVip, 11
CadVlan.EnvironmentVip.business, 11
CadVlan.EquipAccess, 11
CadVlan.EquipGroup, 11
CadVlan.EquipInterface, 12
CadVlan.Equipment, 12
CadVlan.EquipmentType, 12
CadVlan.EquipScript, 12
CadVlan.EventLog, 12
CadVlan.Filter, 13
CadVlan.GroupEquip, 13
CadVlan.GroupEquip.business, 13
CadVlan.GroupUser, 13
CadVlan.HealthcheckExpect, 13
CadVlan.Ldap, 14
CadVlan.messages, 18
CadVlan.Net, 14
CadVlan.Net.business, 14
CadVlan.NetworkType, 14
CadVlan.OptionVip, 14
CadVlan.permissions, 18
CadVlan.processExceptionMiddleware, 18
CadVlan.Script, 14
CadVlan.ScriptType, 15
CadVlan.settings, 18
CadVlan.templates, 18
CadVlan.User, 15
CadVlan.UserGroup, 15
CadVlan.Util, 17
CadVlan.Util.converters, 15
CadVlan.Util.converters.util, 15
CadVlan.Util.Enum, 16
CadVlan.Util.extends, 16
CadVlan.Util.file, 16
CadVlan.Util.forms, 16
CadVlan.Util.templatetags, 16
CadVlan.VipRequest, 17
CadVlan.Vlan, 18

A

ACL_APPLY (in module CadVlan.permissions), 18

C

CadVlan (module), 18
CadVlan.AccessType (module), 9
CadVlan.Acl (module), 9
CadVlan.Auth (module), 10
CadVlan.Auth.models (module), 10
CadVlan.BlockRules (module), 10
CadVlan.Environment (module), 11
CadVlan.EnvironmentVip (module), 11
CadVlan.EnvironmentVip.business (module), 11
CadVlan.EquipAccess (module), 11
CadVlan.EquipGroup (module), 11
CadVlan.EquipInterface (module), 12
CadVlan.Equipment (module), 12
CadVlan.EquipmentType (module), 12
CadVlan.EquipScript (module), 12
CadVlan.EventLog (module), 12
CadVlan.Filter (module), 13
CadVlan.GroupEquip (module), 13
CadVlan.GroupEquip.business (module), 13
CadVlan.GroupUser (module), 13
CadVlan.HealthcheckExpect (module), 13
CadVlan.Ldap (module), 14
CadVlan.messages (module), 18
CadVlan.Net (module), 14
CadVlan.Net.business (module), 14
CadVlan.NetworkType (module), 14
CadVlan.OptionVip (module), 14
CadVlan.permissions (module), 18
CadVlan.processExceptionMiddleware (module), 18
CadVlan.Script (module), 14
CadVlan.ScriptType (module), 15
CadVlan.settings (module), 18
CadVlan.templates (module), 18
CadVlan.User (module), 15
CadVlan.UserGroup (module), 15
CadVlan.Util (module), 17

CadVlan.Util.converters (module), 15
CadVlan.Util.converters.util (module), 15
CadVlan.Util.Enum (module), 16
CadVlan.Util.extends (module), 16
CadVlan.Util.file (module), 16
CadVlan.Util.forms (module), 16
CadVlan.Util.templatetags (module), 16
CadVlan.VipRequest (module), 17
CadVlan.Vlan (module), 18
create() (CadVlan.Util.file.File class method), 16

E

Enum (class in CadVlan.Util.Enum), 16

F

File (class in CadVlan.Util.file), 16
FileError, 17

G

get_active() (CadVlan.Auth.models.User method), 10
get_ambiente_by_id() (in module CadVlan.EnvironmentVip.business), 11
get_email() (CadVlan.Auth.models.User method), 10
get_id() (CadVlan.Auth.models.User method), 10
get_name() (CadVlan.Auth.models.User method), 10
get_password() (CadVlan.Auth.models.User method), 10
get_permission() (CadVlan.Auth.models.User method), 10
get_user_ldap() (CadVlan.Auth.models.User method), 10
get_username() (CadVlan.Auth.models.User method), 10

H

has_perm() (CadVlan.Auth.models.User method), 10
has_perm_menu() (CadVlan.Auth.models.User method), 10

I

is_valid_ipv4() (in module CadVlan.Net.business), 14
is_valid_ipv6() (in module CadVlan.Net.business), 14

L

LoggingMiddleware (class in CadVlan.processExceptionMiddleware), [18](#)

P

process_exception() (CadVlan.processExceptionMiddleware.LoggingMiddleware method), [18](#)

R

read() (CadVlan.Util.file.File class method), [17](#)

remove() (CadVlan.Util.file.File class method), [17](#)

replace_id_to_name() (in module CadVlan.Util.converters.util), [15](#)

S

set_password() (CadVlan.Auth.models.User method), [10](#)

split_to_array() (in module CadVlan.Util.converters.util), [15](#)

T

to_complete_list() (in module CadVlan.GroupEquip.business), [13](#)

U

User (class in CadVlan.Auth.models), [10](#)

W

write() (CadVlan.Util.file.File class method), [17](#)