

# Installation of pynsxv

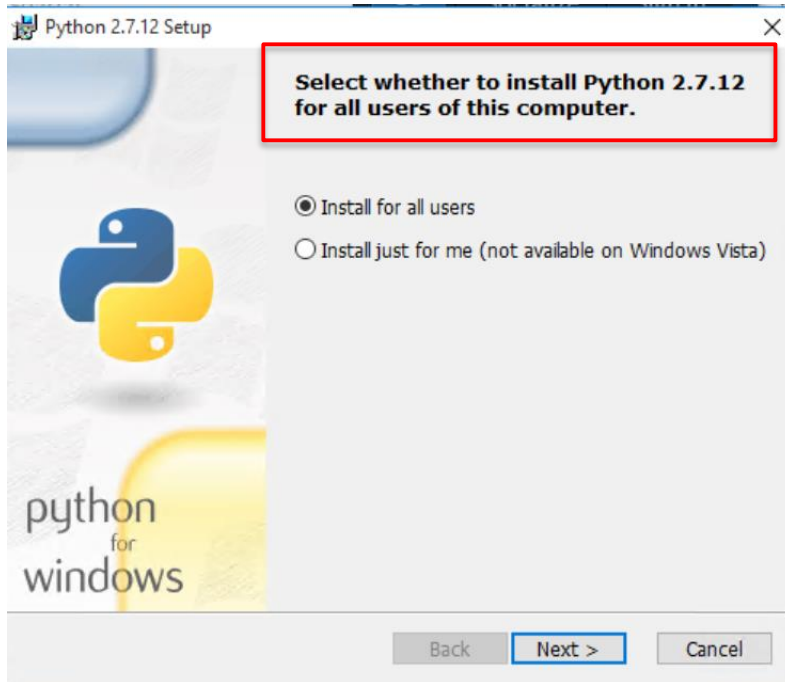
1. **Installation of Python + NSX toolkit (pynsxv)**  
For Windows  
For Linux
2. Validation of pynsxv installation

# Installation of Python + NSX toolkit (pynsxt) – Windows

## 1. Installation of python

- Install python 2.x (<https://www.python.org/downloads/>)

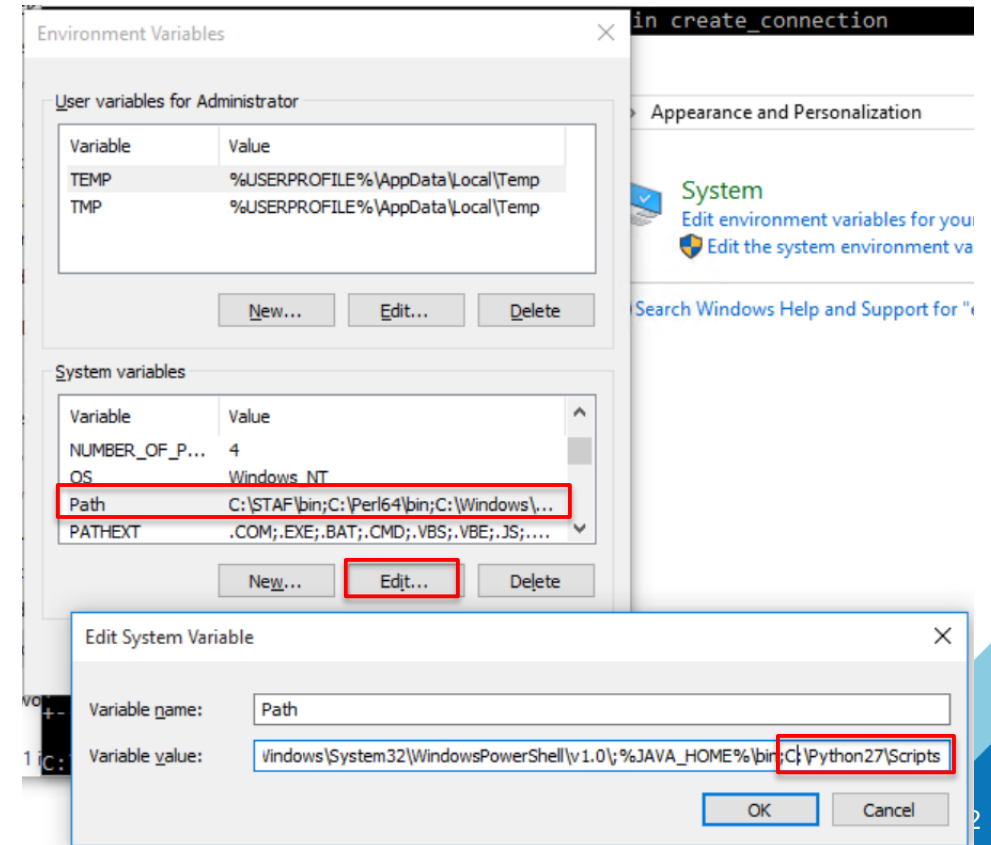
*Note: Attention do NOT install python 3.x*



- Add python installation folder in the Windows path

*So you can run pynsxt from any folder*

Under "Control Panel – Environment Variables", edit the System Variable "Path", and add the "Python27\Scripts" folder:



# Installation of Python + NSX toolkit (pynsxv) – Windows

## 2. Installation of pynsxv

– pip install pynsxv

```
C:\Windows\system32>pip install pynsxv
Collecting pynsxv
  Downloading PyNSXv-0.3.tar.gz (40kB)
    100% |#####| 40kB 655kB/s
Collecting nsxramlclient>=2.0 (from pynsxv)
  Downloading nsxramlclient-2.0.0.tar.gz
Collecting pyvmomi (from pynsxv)
  Downloading pyvmomi-6.0.0.2016.6.tar.gz (218kB)
    100% |#####| 225kB 1.1MB/s
Collecting tabulate (from pynsxv)
```

# Installation of Python + NSX toolkit (pynsxt) – Windows

3. Create nsx.ini file (information for python on NSX + vCenter)

```
nsx.ini - Notepad
File Edit Format View Help
# variables for pynsxt

[nsxv]
nsx_manager = 192.168.10.5
nsx_username = admin
nsx_password = vmware

[vcenter]
vcenter = 192.168.10.4
vcenter_user = administrator@vsphere.local
vcenter_passwd = VMware1!

[defaults]
transport_zone = TZ1
datacenter_name = Lab1
edge_datastore = NFS_Lab1
edge_cluster = Cluster-MgtEdge
|
```

NSX-V Manager information

vCenter information

Other information

4. That's it 😊

# Installation of Python + NSX toolkit (pynsxv) – Linux / Ubuntu

## 1. Installation of python + pip

```
dimi@ubuntu-python:~$ apt-get install libssl-dev libxml2-dev libxslt-dev python-dev zlib1g-dev python-pip -y
```

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'libxslt1-dev' instead of 'libxslt-dev'
```

## 2. Installation of pynsxv

```
dimi@ubuntu-python:~$ sudo pip install pynsxv
```

```
Collecting pynsxv
Collecting nsxramlclient>=2.0 (from pynsxv)
Collecting tabulate (from pynsxv)
Collecting pyvmomi (from pynsxv)
```

# Installation of Python + NSX toolkit (pynsxv) – Linux / Ubuntu

## 3. Create nsx.ini file (information for python on NSX + vCenter)

```
dimi@ubuntu-python:~$ cat nsx.ini
# variables for pynsxv

[nsxv]
nsx_manager = 192.168.10.5
nsx_username = admin
nsx_password = vmware

[vcenter]
vcenter = 192.168.10.4
vcenter_user = administrator@vsphere.local
vcenter_passwd = VMware1!

[defaults]
transport_zone = TZ1
datacenter_name = Lab1
edge_datastore = NFS_Lab1
edge_cluster = Cluster-MgtEdge
```

← NSX-V Manager information

← vCenter information

← Other information

## 4. That's it 😊



# Installation of pynsxv

1. Installation of Python + NSX toolkit (pynsxv)
  - For Windows
  - For Linux
2. **Validation of pynsxv installation**

# Validation of pynsxv installation

## 1. Launch pynsxv help

– pynsxv -h

```
C:\pynsxv>pynsxv.exe -h
usage: pynsxv-script.py [-h] [-i INI] [-v] [-d]
                        {lswitch,dlr,esg,dfw,usage} ...

PyNSXv Command Line Client for NSX for vSphere

positional arguments:
  {lswitch,dlr,esg,dfw,usage}
    lswitch      Functions for logical switches
    dlr          Functions for distributed logical routers
    esg          Functions for edge services gateways
    dfw          Functions for distributed firewall
    usage        Functions to retrieve NSX-v usage statistics

optional arguments:
  -h, --help            show this help message and exit
  -i INI, --ini INI     nsx configuration file
  -v, --verbose         increase output verbosity
  -d, --debug           print low level debug of http transactions
```

Windows

```
dimi@ubuntu-python:~$ pynsxv -h
usage: pynsxv [-h] [-i INI] [-v] [-d] {lswitch,dlr,esg,dfw,usage} ...

PyNSXv Command Line Client for NSX for vSphere

positional arguments:
  {lswitch,dlr,esg,dfw,usage}
    lswitch      Functions for logical switches
    dlr          Functions for distributed logical routers
    esg          Functions for edge services gateways
    dfw          Functions for distributed firewall
    usage        Functions to retrieve NSX-v usage statistics

optional arguments:
  -h, --help            show this help message and exit
  -i INI, --ini INI     nsx configuration file
  -v, --verbose         increase output verbosity
  -d, --debug           print low level debug of http transactions
```

Linux



# Validation of pynsxv installation

## 2. List NSX-v Logical Switches

– `pynsxv lswitch list`

```
C:\pynsxv>pynsxv lswitch list
```

LS name	LS ID
Transit_LS-01	virtualwire-2
Web_LS-01	virtualwire-3
App_LS-01	virtualwire-4
DB_LS-01	virtualwire-5
TSWeb	virtualwire-51
Transit_LS-02	virtualwire-74
Web-01	virtualwire-89
App-01	virtualwire-90
DB-01	virtualwire-91
Transit-01	virtualwire-92

Windows

```
dimi@ubuntu-python:~$ pynsxv lswitch list
```

LS name	LS ID
Transit_LS-01	virtualwire-2
Web_LS-01	virtualwire-3
App_LS-01	virtualwire-4
DB_LS-01	virtualwire-5
TSWeb	virtualwire-51
Transit_LS-02	virtualwire-74
Web-01	virtualwire-89
App-01	virtualwire-90
DB-01	virtualwire-91
Transit-01	virtualwire-92

Linux