

# Step-by-step: How to create a VM package for NFVIS

## What is a VNF package?

VNF package is essentially a set of files bundled for ease of distribution and automation of deployment. VNF package typically contains the disk image(s), bootstrap configuration(s), meta-data that represents the capability, manifest file that confirms integrity.

## Why package?

Creating a VNF package is not mandatory however it has its advantages

Provides a way to scale out deployments.

Support for Day 0 configuration for Cisco and 3rd party VNFs

The packaging utility creates a tar.gz file which contains

The raw QCOW2 file. Note: ISO, vmdk formats would require conversion to qcow2 before package can be created.

Image properties file

Supported and default profiles

Day 0 configs

Image properties file is created by using either the GUI or using the packaging utility provided with every release.

Note: vManage Network Design workflow is recommended for NFVIS device automation and management. VNF package created by NFVIS packaging tool would be imported to vManage Image repository image repository.

There are at least 3 common ways to build a package.

Method 1. Unpack>Edit>Repack.

Method 2. GUI approach, available via NFVIS local portal

Method 3. Python script based tool downloaded from NFVIS local portal. There are validated bootstrap-config file examples for packaging ASAv, ISRv, Thirdparty VM, vedge-cloud, ubuntu, ISRv XE-SDWAN, etc.

## Method 1: Unpack>Edit>Repack

This method is the quickest if minor modifications are required to the files in package. Procedure is described using C8000v VNF package but the method can be used for any NFVIS VNF package.

Starting NFVIS 4.4 release, C8000v is the recommended Cisco routing VNF for use with NFVIS platforms, for autonomous and controller-mode deployments. VNF package is downloaded from [Cisco software downloads site](#) and Untar->Edit->and Retar for use from vManage or from Local NFVIS GUI.

There are requirements to enable NAT configuration in C8000v for NFVIS to utilize the single-IP feature in NFVIS. Also, if vManage uses Enterprise Root Certificate, there is a requirement to add the same in C8000v bootstrap for automated deployment/management.

**sdwan\_cloud\_init.cfg** file has NAT and Root certificate modifications in this example, this would require checksum to be recalculated and updated in **package.mf** .

### [Spoiler](#)

```
[root@kramesh-cent1 vbranch]# tar -xvf C8000v_17.06.01a_8G_serial_vBranch.tar.gz
c8000v-universalk9_8G_serial.17.06.01a.qcow2
ovf-env.xml
iosxe_config.txt
sdwan_cloud_init.cfg
```



DWVuZnYtc2R3YW4tQowxIDAeBgkqhkiG9woBCQEWEWtyYW1lc2hAY2lzY28uY29t  
MB4XDTIwMDcwMzIzMjgwN1oXDTMwMDcwMTIzMjgwN1owgZYxCzAJBgNVBAYTA1VT  
MQswCQYDVQQIDAJDQTEQMA4GA1UEBwwHU2FuSm9zZTEWMBQGA1UECgwNZW5mdi1z  
ZHdhbi1DTDEWMBQGA1UECwwNZW5mdi1zZHdhbi1DTDEWMBQGA1UEAwwNZW5mdi1z  
ZHdhbi1DTDEgMB4GCSqGSIb3DQEJARYRa3JhbWVzaEBjaXNjby5jb2owggLiMAoG  
CSqGSIb3DQEBAQUAA4ICDwAwggIKAoICAQDj6KNLvCJO9DmBHE4TCZ5GS7wcCLjC  
gfm2KANSpGy6LgP1zosdxzIfV7zNn3xpOse4/oYsGxuNiwzy3mVxQB8w+BQPxklf  
E+Rh+UUOW8d3lHd3oMASoegzQB2kbD7qwHZPxRAwEROZRQ3dKEEWLZCcFBrkYa+  
esgsQtHq4kebeza1Tb1AdtmnJHnRONxu8aWocH4dBggqEHw+q267NOAnwHAD5B2p  
7VR+Ew56raKIEVAX5y2iDcY7Ozn5SvWmThdSVg+i3QK2IO/vkXbbP4HolyUgxUag  
u6GCYU4rZ6mrwDCXWouBL2Zk78z8d6CWdxjhIYZXh1Ivi9y1LM7C+OXuivyCd4f  
PpwxwwivVrtdo4IPV8v+xQQUYFmq6HTl+OsHo8zJ+9LFRDUY9FDNS+vtAaDVsifp  
pbAKwSRAV7kMSJKoARsHrtyHu8t4O1Oz65IS8Ru5GCOZ5ilIqomDOsz/tqyngCX  
LlIoiPbt6WmRBELMyLGghYKkU1ypmYrjNXvTnhHhblLUdvt1H4+g5M5GrBqysV92  
yu+RcEI9kSO7Lzdz3XLX5X/uI79zRqnleRPxP+XrG22fGHmwajhb6fvJ4c8SsN4  
BT3+NtnNUreZFXEJF1TSoAXQixZt/Q3LalIIJoFioNxCoIRA9pVbFswxUfvVGk+E  
tAJtZmsokKBrXQIDAQABo2owazAdBgNVHQ4EFgQUvA2FZE8IAKQyWvqRmXgQgQBm  
y+UwHwYDVR0jBBgwFoAUvA2FZE8IAKQyWvqRmXgQgQBmy+UwDAYDVR0TBAAUwAwEB  
/zAbBgNVHREEFDAShWssE6DDhwSsE6B+hwTAqAGWMAoGCSqGSIb3DQEBCwUAA4IC  
AQDA5QvV17rA58g6fZZHYa8tU6jZywaVXZc2aZD/Jkccq4hLYQyKB5od6C3BFXhHQ  
URr4ufD5cs9fgSPUSraBy1wnDRom5NNKmtY57GQHs6JlEt/bgQoOl/gXZWNNZX2N  
LH+uJNoVBnQlMcuxnuCB4WU4A26hCzqqIzajaA9wFhcuMBYnX+ZBkzvWAlf31nHq  
/6i7NcsffyW5A8AzzHG7E2smIIF9PpOu7o7xozuGNg1kmPC79+kAor2bNnbqHl9W  
ZTasAXa5TDx1xmGN0+6Oa437Wp6JWdIod1PofjefdkGgpFGvnHKHRcH2iLLFUJ+z  
tkoXpLQWhTVPyOdgYdD1Oz/CBvkrgr+6Eakjblk5oqbJYdhikoP1Au3BCkr4QbC1  
HW89k4uMQ7y3/gBpEWxt/7EIOePDLflKx1T7hbwd250+73/lqowGCobCzdeUsE4P  
E//aEK8YBUxLI2aCtgNx4Hc2k2EQ3FZWBvuoRPGfrfQJuTkZ5PjXacrAsPD4Tk+J  
qoSrp8NMK2L1qxPHu/XccTzP2LMmR8i2CHDHSgc4S11zDeZDZjQW4WJL4bXtNw5E  
GKoLdk3qiYFafJUN1jL/dSfJByJI09NmS9+gUlRdKxTztSvl6MSq4sRxP5D6ai09  
xEiG4NULuokZbaazIRTc8+Jzpsydho/WVqIhLEa6g+Qkqw==  
-----END CERTIFICATE-----

-----2587222130433519110==  
Content-Type: text/cloud-boothook; charset="us-ascii"  
MIME-Version: 1.0  
Content-Transfer-Encoding: 7bit  
Content-Disposition: attachment;  
filename="config-default.txt"  
#cloud-boothook  
system  
host-name {HOSTNAME}  
system-ip {SYSTEM\_IP}  
overlay-id 1  
site-id {SITE\_ID}  
port-offset 0  
control-session-pps 300  
admin-tech-on-failure  
sp-organization-name "{ORGNAME}"  
organization-name "{ORGNAME}"  
port-hop  
track-transport  
track-default-gateway  
console-baud-rate 115200

```
vbond {VBOND} port 12346
logging
disk
enable
!
!
!
bfd app-route multiplier 6
bfd app-route poll-interval 600000
sslproxy
no enable
rsa-key-modulus 2048
certificate-lifetime 730
eckey-type P256
ca-tp-label PROXY-SIGNING-CA
settings expired-certificate drop
settings untrusted-certificate drop
settings unknown-status drop
settings unsupported-protocol-versions drop
settings unsupported-cipher-suites drop
settings failure-mode close
settings minimum-tls-ver TLSv1
!
no tcp-proxy enable
!
sdwan
interface GigabitEthernet2
tunnel-interface
encapsulation ipsec weight 1
no border
color default
no last-resort-circuit
no low-bandwidth-link
no vbond-as-stun-server
vmanage-connection-preference 5
port-hop
carrier default
nat-refresh-interval 5
hello-interval 1000
hello-tolerance 12
no allow-service all
no allow-service bgp
allow-service dhcp
allow-service dns
allow-service icmp
allow-service sshd
no allow-service netconf
no allow-service ntp
no allow-service ospf
no allow-service stun
allow-service https
no allow-service snmp
```

```
exit
exit
appqoe
no tcpopt enable
!
omp
no shutdown
send-path-limit 4
ecmp-limit 4
graceful-restart
no as-dot-notation
timers
holdtime 60
advertisement-interval 1
graceful-restart-timer 43200
eor-timer 300
exit
address-family ipv4
advertise connected
advertise static
!
address-family ipv6
advertise connected
advertise static
!
!
!
security
ipsec
rekey 86400
replay-window 512
authentication-type sha1-hmac ah-sha1-hmac
!
!
no service pad
service tcp-keepalives-in
service tcp-keepalives-out
no service tcp-small-servers
no service udp-small-servers
username admin privilege 15 secret 0 admin
vrf definition Mgmt-intf
description Transport VPN
rd 1:512
address-family ipv4
route-target export 1:512
route-target import 1:512
exit-address-family
!
address-family ipv6
exit-address-family
!
!
```

```
vrf definition 511
address-family ipv4
exit-address-family
!
address-family ipv6
exit-address-family
!
!
vrf definition {DATA_VPN_NUMBER}
!
address-family ipv4
exit-address-family
!
address-family ipv6
exit-address-family
!
vrf definition {MANAGEMENT_VPN_NUMBER}
!
address-family ipv4
exit-address-family
!
address-family ipv6
exit-address-family
!
hostname {HOSTNAME}
username {SSH_USERNAME} privilege 15 secret o {SSH_PASSWORD}
enable password {ENABLE_PASSWORD}
!
ip name-server {DNS_IP}
!
ip arp proxy disable
no ip finger
no ip rcmd rcp-enable
no ip rcmd rsh-enable
no ip dhcp use class
ip multicast route-limit 2147483647
ip bootp server
no ip source-route
no ip http server
no ip http secure-server
no ip http ctc authentication
no ip igmp ssm-map query dns
interface GigabitEthernet1
vrf forwarding 511
description MGMT
no shutdown
arp timeout 1200
ip address ${NICID_o_IP_ADDRESS} ${NICID_o_NETMASK}
ip redirects
ip mtu 1500
mtu 1500
negotiation auto
```

```
exit
interface GigabitEthernet2
description Transport
no shutdown
arp timeout 1200
ip address {VPNo_WAN_IP_ADDRESS} {VPNo_WAN_NETMASK}
ip nat outside
ip redirects
ip mtu 1500
mtu 1500
negotiation auto
exit
interface GigabitEthernet3
vrf forwarding {MANAGEMENT_VPN_NUMBER}
ip address {MGMT_IP_ADDRESS} {MGMT_NETMASK}
no shutdown
exit
!
interface GigabitEthernet4
vrf forwarding {DATA_VPN_NUMBER}
ip address {LAN_IP_ADDRESS} {LAN_NETMASK}
no shutdown
exit
!
interface Tunnel2
no shutdown
ip unnumbered GigabitEthernet2
no ip redirects
ipv6 unnumbered GigabitEthernet2
no ipv6 redirects
tunnel source GigabitEthernet2
tunnel mode sdwan
exit
clock timezone UTC 0 0
logging persistent size 104857600 filesize 10485760
logging buffered 512000
no logging rate-limit
logging persistent
aaa authentication login default local
aaa authorization exec default local
aaa session-id common
no crypto ikev2 diagnose error
no crypto isakmp diagnose error
snmp-server ifindex persist
line con 0
login authentication default
speed 115200
stopbits 1
!
line vty 0 4
transport input ssh
!
```

```

line vty 5 80
transport input ssh
!
lldp run
nat64 translation timeout tcp 60
nat64 translation timeout udp 1
!
!
ip route 0.0.0.0 0.0.0.0 {VPNo_WAN_GATEWAY}
!
ip nat inside source list nat-dia-vpn-hop-access-list interface GigabitEthernet2 overload
ip nat route vrf 511 0.0.0.0 0.0.0.0 global
!

```

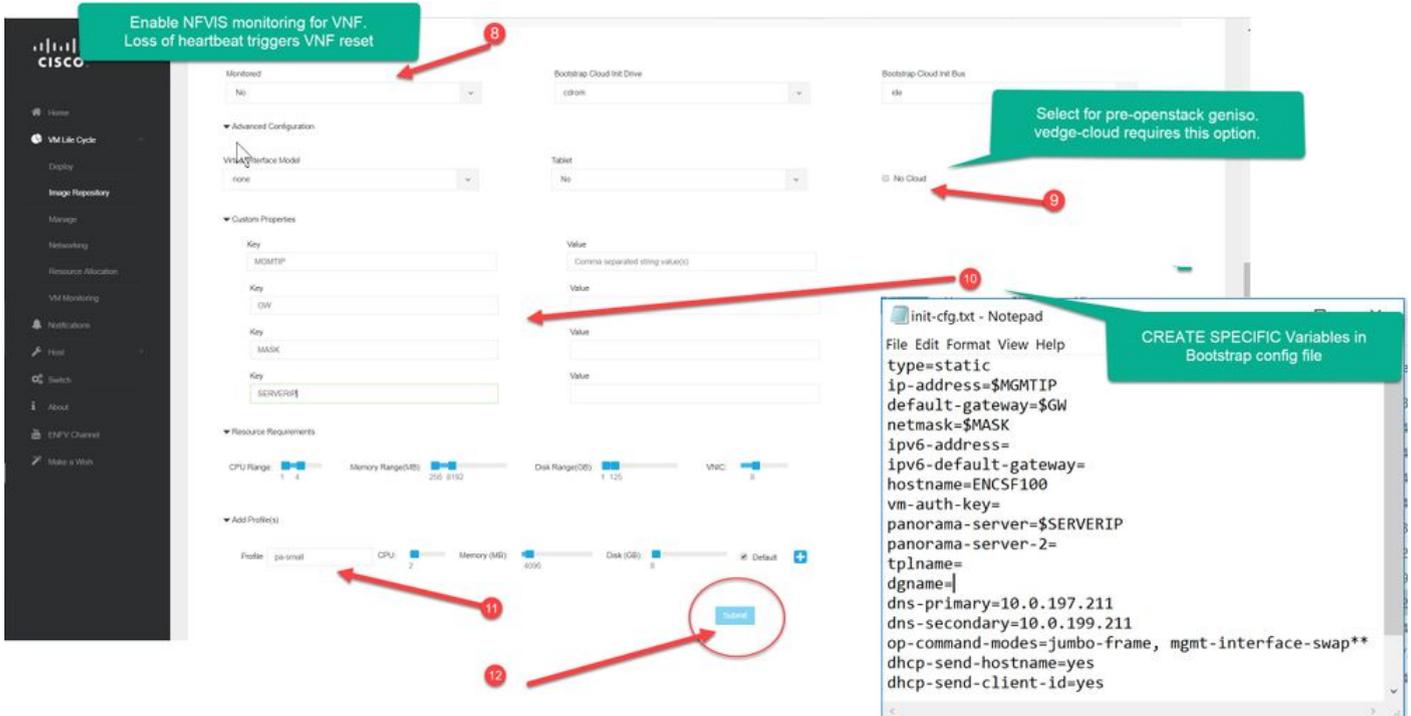
If there are new variable added to the bootstrap file or other environmental parameter changes, image\_properties.xml file would also require update and the corresponding checksum update in package.mf.

## Method 2: NFVIS Local Portal based Image Packaging

[Spoiler](#)

The screenshot displays the NFVIS Image Packaging interface. The left sidebar shows the navigation menu with 'Image Repository' highlighted. The main content area is divided into sections for 'Image Registration', 'Browse Datastore', 'USB Upload', and 'Image Packaging'. The 'Image Packaging' section is circled in red. Red arrows point to various fields: 1. Package Name (PaloFW), 2. VM Version (7.1.4), 3. Image Packaging, 4. Dedicate Cores(Optimize) (Yes), 5. Serial Console (Enable), 6. Choose drivers supported by VNF, and 7. Upload Bootstrap Files. A green callout box states: 'For Network Function, YES Recommended'. Another green callout box states: 'Most VMs boot with virtio, some require IDE to boot'. A third green callout box states: 'FILENAME and the MOUNT Directory are VM specific. refer VM documentation. Note: Some VMs require multiple bootstrap config files.' The interface also shows a table for 'Upload Progress' and 'Raw Disk File Bus' (virtio) and 'Thick Disk Provisioning' (No).

Image Packaging for NFVIS



### Image Packaging for NFVIS (Contd.)

Next step is to register the package that was built for use in the local system.

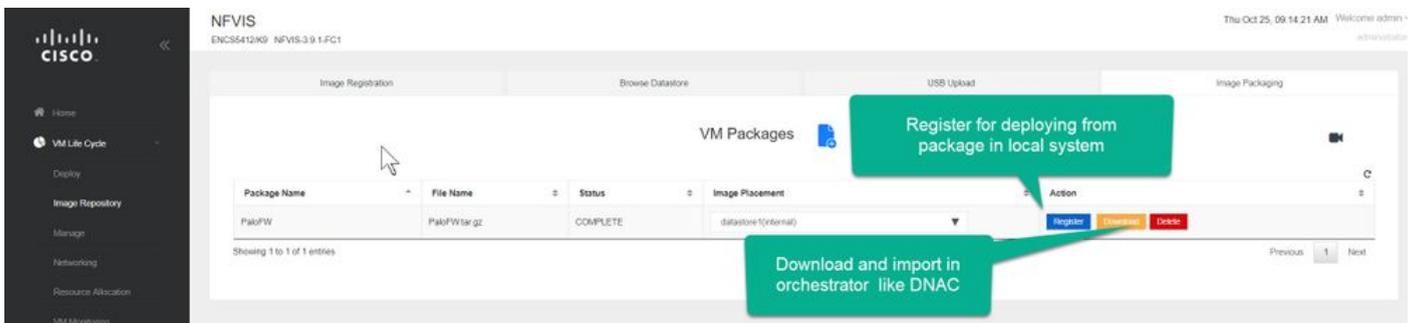
Alternately, the package can be downloaded for use via an orchestrator like DNA Center.

**Note :** For file sizes larger than 1GB, SCP based copy would be a faster and reliable method. By default SCP is disabled in NFVIS system. SCP can be enabled from specific source IP using the following CLI.

```
nfviz(config)# system settings ip-receive-acl 0.0.0.0/0 service scp priority 2 action accept
```

```
nfviz(config-ip-receive-acl-0.0.0.0/0)# commit
```

### Spoiler



### Image Packaging for NFVIS Complete

## Method 3: Download and Execute python script based tool

Untar the downloaded file and the following files are part of the package. Python 2.7+ environment is required for executing the script.

```
tar -xvf nfvizmpackagingtool.tar
```

```
nfvpt.py
```

```
image_properties_template.xml
```

```
tool-usage-examples.txt
```

### Example 1: Usage for ASA v

NOTE: --bootstrap1 dayo-config (The bootstrap filename **\*\*must\*\*** be dayo-config as ASA v only looks for this filename

```
nfvpt.py -o asav961-201 -i asav961-201.qcow2 -n ASA v -t firewall -r 961-201 --monitored true --bootstrap dayo-config:filename1
```

```
--min_vcpu 1 --max_vcpu 4 --min_mem 1024 --max_mem 8192 --min_disk 8 --max_disk 16 --vnic_max 8 --optimize true
--profile ASAv5,"ASAv5 profile",1,1024,8192 --profile ASAv10,"ASAv10 profile",1,4096,8192 --profile ASAv30,"ASAv30
profile",4,8192,16384 --default_profile ASAv5
```

### Example 2: Usage for ISRv

NOTE: --bootstrap1 ovf-env.xml (The bootstrap filename **must** be ovf-env.xml as ISRv only looks for this filename  
nfvpt.py -o isrv.16.03.01 -i isrv-universalk9.16.03.01.qcow2 -n ISRv.16.03.01 -t ROUTER -r 16.03.01 --monitored true  
--privileged true --bootstrap ovf-env.xml:file1,ios-xe.txt:file2 --min\_vcpu 2 --max\_vcpu 8 --min\_mem 4096 --max\_mem 8192  
--min\_disk 8 --max\_disk 8 --vnic\_max 8 --optimize true --profile ISRv-small,"ISRv small profile",2,4096,8192 --profile ISRv-  
medium,"ISRv medium profile",4,4096,8192 --default\_profile ISRv-small --sriov\_list igb,igbvf,i40evf --custom tech\_package,ax

### Example 3: Usage for 3rd Party VM with config drive (ISO) mounted at specific path on the VM:

```
nfvpt.py -o test.1.0 -i test-1.0.qcow2 -n TEST -t OTHER -r 1.0 --monitored true --privileged true --bootstrap /:bootstrap.xml,
/license/lic.txt:license.txt --min_vcpu 2 --max_vcpu 8 --min_mem 4096 --max_mem 8192 --min_disk 8 --max_disk 8
--vnic_max 8 --optimize true --profile small,"small profile",2,4096,8192 --profile medium,"medium profile",4,4096,8192
--default_profile small
```

in this case test.1.0.pkg : bootstrap.xml get mounted as bootstrap.xml at root and license.txt get mounted as /license/lic.txt

### Example 4: Usage for vedge-cloud

```
nfvpt.py -o vedge18.3.1 -i viptela-edge-genericx86-64.qcow2 -n vedge.18.03.01 -t ROUTER -r 18.03.01 --monitored false
--privileged true --bootstrap /openstack/latest/user_data:cloudinit.cfg,/openstack/latest/meta_data.json:meta_data,
/openstack/latest/vendor_data.json:vendor_data --min_vcpu 2 --max_vcpu 8 --min_mem 4096 --max_mem 8192 --min_disk
8 --max_disk 8 --vnic_max 8 --optimize true --nocloud true --profile vEdge-small,"vEdge small profile",2,4096,8192 --profile
vEdge-Standard,"vEdge Standard profile",4,4096,8192 --default_profile vEdge-Standard --custom key:ORGNAME,val:""
--custom key:OTP,val:"" --custom key:UUID,val:"" --custom key:SYSTEM_IP,val:"" --custom key:VBOND,val:"" --custom
key:SITE_ID,val:""
```

There are 3 files to be passed as bootstrap config files, please follow the information below and create the 3 files and make them available in the same directory as the nfvpt.py

**File 1: cloudinit.cfg** Action : save the text below in file, name it cloudinit.cfg

#### [Spoiler](#)

```
Content-Type: multipart/mixed; boundary="====8815267485200512281=="
MIME-Version: 1.0
-----8815267485200512281==
Content-Type: text/cloud-config; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="cloud-config"

#cloud-config
vinitparam:
- otp : ${OTP}
- vbond : ${VBOND}
- uuid : ${UUID}
- org : ${ORGNAME}
-----8815267485200512281==
Content-Type: text/cloud-boothook; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment;
```

filename="config-6978c2a4-8f5e-4489-8500-80e8048f60ad.txt"

#cloud-boothook

system

personality vedge

device-model vedge-cloud

chassis-number \${UUID}

ztp-status success

config-template-name "vEdge-cloud-template"

pseudo-confirm-commit 300

!

system

personality vedge

device-model vedge-cloud

host-name vedgecloud1

system-ip \${SYSTEM\_IP}

domain-id 1

site-id \${SITE\_ID}

no route-consistency-check

organization-name "\${ORGNAME}"

vbond \${VBOND} port 12346

aaa

auth-order local radius tacacs

usergroup basic

task system read write

task interface read write

!

usergroup netadmin

!

usergroup operator

task system read

task interface read

task policy read

task routing read

task security read

!

user admin

password

\$6\$siwKBQ==\$wT2lUa9BSreDPI6gB8sl4E6PAJoVXgMbgv/whJ8F1C6sWdRazdxorYYTLrL6syiG6qnLABTnrE96HJiKF6QRq1

!

!

logging

disk

enable

!

!

!

omp

no shutdown

graceful-restart

advertise connected

advertise static

```
!
security
ipsec
authentication-type sha1-hmac ah-sha1-hmac
```

```
!
!
vpn 0
name "Transport VPN"
interface geo/0
ip dhcp-client
tunnel-interface
encapsulation ipsec
no allow-service bgp
allow-service dhcp
allow-service dns
allow-service icmp
no allow-service sshd
no allow-service netconf
no allow-service ntp
no allow-service ospf
no allow-service stun
```

```
!
no shutdown
```

```
!
!
```

```
vpn 1
interface geo/1
no shutdown
```

```
!
!
```

```
vpn 512
name "Transport VPN"
interface eth0
ip dhcp-client
no shutdown
```

```
!
!
!
!
```

-----8815267485200512281----

**File2: meta\_data** Action : save the text below in file, name it meta\_data

```
{"uuid":"myinstance.mydomain.com"}
```

**File3: vendor\_data** Action : save the text below in file, name it vendor\_data

```
{"test":"sample"}
```

### Example 5: Usage for Ubuntu Linux

```
nfvpt.py -o Ubuntu16.04 -i ubuntu.qcow2 -n ubuntu.16.04 -t OTHER -r 16.04 --monitored false --privileged true --bootstrap
user-data:user-data, meta-data:meta_data --min_vcpu 1 --max_vcpu 4 --min_mem 2048 --max_mem 4096 --min_disk 8
--max_disk 20 --vnic_max 4 --optimize true --nocloud true --profile ubuntu-small,"ubuntu small profile",1,2048,8192 --profile
```

ubuntu-medium,"ubuntu medium profile",2,4096,20480 --default\_profile ubuntu-small --custom IP\_ADDRESS, --custom NETMASK, --custom UUID, --custom GATEWAY,

There are 2 files to be passed as bootstrap config files, please follow the information below and create the 2 files and make them available in the same directory as the nfvpt.py

**File 1 : user-data    Action : save the text below in file, name it user-data**

[Spoiler](#)

```
#!/bin/bash
passwd root << EOF
cisco123
cisco123

Y
EOF
echo "Cloud-init running user-data off config drive (/dev/sro)"
echo Setting up interfaces and addresses
ifconfig ens3 down
ifconfig ens3 $NICID_o_IP_ADDRESS netmask $NICID_o_NETMASK
ifconfig ens4 down
ifconfig ens4 $IP_ADDRESS netmask $NETMASK
route add default gw $GATEWAY ens3
netstat -rn
adduser lab sudo
ifconfig ens3 up
ifconfig ens4 up
sed -i 's/PermitRootLogin no/PermitRootLogin yes/' /etc/ssh/sshd_config
sed -i 's/PasswordAuthentication no/PasswordAuthentication yes/' /etc/ssh/sshd_config
service ssh restart
```

**File 2 : meta-data    Action : save the text below in file, name it meta-data**

[Spoiler](#)

```
instance-id: ubuntu-bionic
hostname: ubuntu-bionic
local-hostname: ubuntu-bionic
```

**Example 6 : Usage for ISRV XE-SDWAN package**

```
./nfvpt.py -o isrv-sdwan-16.12.01e -i isrv-ucmk9.16.12.01e-vga.qcow2 -n isrv-sdwan.16.12.01e -t ROUTER -r 16.12.01e
--monitored true --privileged true --optimize false --bootstrap ovf-env.xml:isrv_ovf_env.xml --bootstrap
ciscosdwan_cloud_init.cfg:cloudinit.cfg --min_vcpu 1 --max_vcpu 8 --min_mem 4096 --max_mem 8192 --min_disk 8
--max_disk 8 --vnic_max 8 --optimize true --nocloud true --profile ISRV-mini,"ISRV-mini",1,4096,8192 --profile ISRV-
small,"ISRV-small",2,4096,8192 --profile ISRV-medium,"ISRV-medium",4,4096,8192 --default_profile ISRV-small --custom
key:UUID,val:"" --custom key:OTP,val:"" --custom key:SYSTEM_IP,val:"" --custom key:ORG_NAME,val:"" --custom
key:VBOND_IP,val:""
```

There are 2 files to be passed as bootstrap config files, please follow the information below and create the 2 files and make them available in the same directory as the nfvpt.py

**File 1 : ovf-env.xml    Action : save the text below in file, name it isrv\_ovf\_env.xml**

[Spoiler](#)

```
<?xml version="1.0" encoding="UTF-8"?>
<Environment
xmlns:oe="http://schemas.dmtf.org/ovf/environment/1">
```

```

<PropertySection>
<Property oe:key="com.cisco.csr1000v.config-version.1" oe:value="1.0"/>
<Property oe:key="com.cisco.csr1000v.enable-ssh-server.1" oe:value="True"/>
<Property oe:key="com.cisco.csr1000v.login-username.1" oe:value="cisco"/>
<Property oe:key="com.cisco.csr1000v.login-password.1" oe:value="cisco!srv123!"/>
<Property oe:key="com.cisco.csr1000v.mgmt-interface.1" oe:value="GigabitEthernet1"/>
!!!GigabitEthernet1-nicid(o)-int-mgmt-interface-don't change ip address or don't shutdown
<Property oe:key="com.cisco.csr1000v.mgmt-ipv4-addr.1" oe:value="{NICID_o_IP_ADDRESS}/24"/>
<Property oe:key="com.cisco.csr1000v.mgmt-ipv4-network.1" oe:value=""/>
<Property oe:key="com.cisco.csr1000v.license.1" oe:value="{TECH_PACKAGE}"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0001" oe:value="vrf definition Mgmt-intf"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0002" oe:value="address-family ipv4"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0003" oe:value="exit-address-family"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0004" oe:value="address-family ipv6"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0005" oe:value="exit-address-family"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0006" oe:value="exit"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0007" oe:value="interface GigabitEthernet1"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0008" oe:value="vrf forwarding Mgmt-intf"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0009" oe:value="ip address {NICID_o_IP_ADDRESS}
{NICID_o_NETMASK}"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0010" oe:value="no shut"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0011" oe:value="exit"/>
<Property oe:key="com.cisco.csr1000v.ios-config-0012" oe:value="ip route vrf Mgmt-intf 0.0.0.0 0.0.0.0
{NICID_o_GATEWAY}"/>
</PropertySection>
</Environment>

```

**File 2 : ciscosdwan\_cloud\_init.cfg** **Action :** Following is a sample bootstrap config, save the file with filename **cloudinit.cfg**

[Spoiler](#)

```

Content-Type: multipart/mixed; boundary="=====6177259887390062818=="
MIME-Version: 1.0
-----6177259887390062818==
Content-Type: text/cloud-config; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="cloud-config.txt"

#cloud-config
vinitparam:
- uuid : {UUID}
- vbond : {VBOND_IP}
- otp : {OTP}
- org : {ORG_NAME}
- rcc : false
-----6177259887390062818==
Content-Type: text/cloud-boothook; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="cloud-boothook.txt"

#cloud-boothook
viptela-system:system

```

```
personality vedge
device-model vedge-ISRv
host-name cedge10
location Montpellier
gps-location latitude 43.61
gps-location longitude 3.87
system-ip ${SYSTEM_IP}
overlay-id 1
site-id 6
control-session-pps 300
admin-tech-on-failure
sp-organization-name ${ORG_NAME}
organization-name ${ORG_NAME}
console-baud-rate 115200
vbond ${VBOND_IP} port 12346
logging
disk
enable
!
!
!
bfd app-route multiplier 6
bfd app-route poll-interval 600000
omp
no shutdown
graceful-restart
!
security
ipsec
rekey 86400
replay-window 512
authentication-type sha1-hmac ah-sha1-hmac
!
!
no service pad
no service tcp-small-servers
no service udp-small-servers
hostname cedge9
username admin privilege 15 secret 9 $9$2lQL4/EF2FAD4.$ZCo.pWMPXpJMBqlMPRD4AA3NFpB5vFjMBhs9O9O7jYUw
username jmb privilege 15 secret 9 $9$3V6I3VUJ2FUH3U$ckY5VO4xsstH8Q1Uxjo2roFl6t8eSYLqDnpx.JbDF7Vs
vrf definition 10
rd 1:10
address-family ipv4
exit-address-family
!
address-family ipv6
exit-address-family
!
!
vrf definition Mgmt-intf
description Transport VPN
rd 1:512
```

```
address-family ipv4
exit-address-family
!
address-family ipv6
exit-address-family
!
!
no ip finger
no ip rcmd rcp-enable
no ip rcmd rsh-enable
no ip dhcp use class
ip route 0.0.0.0 0.0.0.0 10.60.23.254 1
no ip igmp ssm-map query dns
no ip rsvp signalling rate-limit
no ipv6 mld ssm-map query dns
interface GigabitEthernet1
description MGMT Interface
no shutdown
arp timeout 1200
vrf forwarding Mgmt-intf
ip address ${NICID_o_IP_ADDRESS} ${NICID_o_NETMASK}
ip redirects
ip dhcp client default-router distance 1
ip mtu 1500
mtu 1500
negotiation auto
exit
interface GigabitEthernet2
description INET Transport
no shutdown
arp timeout 1200
ip address dhcp client-id GigabitEthernet2
ip redirects
ip dhcp client default-router distance 1
ip mtu 1500
mtu 1500
negotiation auto
exit
interface Loopback0
no shutdown
arp timeout 1200
vrf forwarding 10
ip address 10.10.10.90 255.255.255.255
ip mtu 1500
exit
interface Tunnel2
no shutdown
ip unnumbered GigabitEthernet2
no ip redirects
ipv6 unnumbered GigabitEthernet2
no ipv6 redirects
tunnel source GigabitEthernet2
```

```
tunnel mode sdwan
exit
clock timezone UTC 0 0
logging persistent size 104857600 filesize 10485760
logging buffered 512000
no logging rate-limit
logging persistent
aaa authentication login default local group radius group tacacs+
aaa authorization exec default local group radius group tacacs+
aaa session-id common
no crypto ikev2 diagnose error
no router rip
line con 0
login authentication default
speed 115200
stopbits 1
!
sdwan
interface GigabitEthernet2
tunnel-interface
encapsulation ipsec weight 1
color default
no last-resort-circuit
vmanage-connection-preference 5
no allow-service all
no allow-service bgp
allow-service dhcp
allow-service dns
allow-service icmp
allow-service sshd
no allow-service netconf
no allow-service ntp
no allow-service ospf
no allow-service stun
exit
exit
interface Loopback0
exit
omp
no shutdown
send-path-limit 4
ecmp-limit 4
graceful-restart
timers
holdtime 60
advertisement-interval 1
graceful-restart-timer 43200
eor-timer 300
exit
address-family ipv4 vrf 10
advertise connected
advertise static
```

```
!  
address-family ipv4  
advertise connected  
advertise static  
!  
!  
!  
policy  
app-visibility  
flow-visibility  
no implicit-acl-logging  
log-frequency 1000  
!  
!  
!
```

```
-----6177259887390062818-----
```