**How to install ocserv ssl vpn server on CentOS 7 linux**

Ocserv (openconnect server) is a great ssl vpn server that simulate cisco anyconnect vpn server on ASAs. Today we are going to show you how to install ocserv ssl vpn server on CentOS 7 linux.

1- Install epel repository

Ocserv is available in epel repository. so at first, we install epel:

# yum install epel-release

then update OS:

# yum update

note: if you faced error such the following, open /etc/yum.repos.d/epel.repo and change all “https” to “http”:

Cannot retrieve metalink for repository: epel/x86\_64. Please verify its path and try again

2- Install ocserv

now it’s time to install ocserv. so, run:

# yum install ocserv gnutls-utils

create cert directory:

# mkdir /etc/ocserv/cert

then change directory to cert:

# cd /etc/ocserv/cert/

create template file:

# touch ca.tmpl

then put the following content in it:

cn = "VPN CA"

organization = "Big Corp"

serial = 1

expiration\_days = -1

ca

signing\_key

cert\_signing\_key

crl\_signing\_key

now we should generate required certificates. so run the following command one by one:

# certtool --generate-privkey --outfile ca-key.pem

# certtool --generate-self-signed --load-privkey ca-key.pem --template ca.tmpl --outfile ca-cert.pem

then create server.tmpl file and put the following content in it:

# vim /etc/ocserv/server.tmpl

cn = "My server"

dns\_name = "www.example.com"

organization = "MyCompany"

expiration\_days = -1

signing\_key

encryption\_key

tls\_www\_server

then again issue the following command one by one:

# certtool --generate-privkey --outfile server-key.pem

# certtool --generate-certificate --load-privkey server-key.pem --load-ca-certificate ca-cert.pem --load-ca-privkey ca-key.pem --template /etc/ocserv/server.tmpl --outfile server-cert.pem

now create ssl directory in /etc/ocserv:

# mkdir /etc/ocserv/ssl/

then copy server certificates into it:

cp ca-cert.pem server-key.pem server-cert.pem /etc/ocserv/ssl/

3- Configure ocserv

now it’s time to change some setting of ocserv in its config file. for authentication there are the following solution:

pam, password file, certificate, radius

Here we want to authenticate with password file. so open /etc/ocserv/ocserv.conf and find the following line:

auth = "pam"

comment it and uncomment the following:

auth = "plain[passwd=/etc/ocserv/sample.passwd,otp=./sample.otp]"

then change it:

auth = "plain[passwd=/etc/ocserv/passwd]"

also change the following line:

server-cert = /etc/pki/ocserv/public/server.crt

server-key = /etc/pki/ocserv/private/server.key

ca-cert = /etc/pki/ocserv/cacerts/ca.crt

to this:

server-cert = /etc/ocserv/ssl/server-cert.pem

server-key = /etc/ocserv/ssl/server-key.pem

ca-cert = /etc/ocserv/ssl/ca-cert.pem

uncomment this line:

#ipv4-network = 192.168.1.0/24

put these lines at the end of file:

dns = 8.8.8.8

dns = 1.1.1.1

Then, set the maximal number of clients. Default is 128. Set to zero for unlimited.

max-clients = 128

Set the number of devices a user is able to login from at the same time. Default is 2. Set to zero for unlimited.

max-same-clients = 2

## 4- Configure firewall and routing

Here we use csf. so port 443 needs to be opened in csf. for more information on how to open port in csf please refere to [How to install CSF on CentOS 7 Linux.](https://www.tuxtips.net/how-to-install-csf-on-centos-7-linux)  
then we should nat our private vpn range to server public ip. because here we use csf, we need to create csfpre.sh file and put nat rules in it.

so run the following commnad:

# vim /etc/csf/csfpre.sh

then put these lines in it. change “eno16777736” and “192.168.26.128” with your server’s public interface and ip address:

#!/bin/bash

iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT

iptables -A FORWARD -s 192.168.114.0/24 -j ACCEPT

iptables -A FORWARD -j REJECT

iptables -t nat -A POSTROUTING -s 192.168.114.0/24 -o ens192 -j MASQUERADE

and reload csf:

# csf -r

furthermore, ip forwarding needs to be enabled. so open /etc/sysctl.conf:

# vim /etc/sysctl.conf

then put this line in it:

net.ipv4.ip\_forward = 1

after that run the following:

# sysctl -p

## 5- Create vpn username and password.

as mentioned earlier, here we use authentication via password file. so we create password file:

# touch /etc/ocserv/passwd

then create your desired username and password. here we create “test” user with group named “default”:

# ocpasswd -c /etc/ocserv/passwd -g default test

## 6- Start Ocserv

Finally, start ocserv service:

# systemctl start ocserv

# systemctl enable ocserv

also make sure it is running:

# systemctl status ocserv

## 7- Install connection

for pc refere to [Graphical OpenConnect client](https://openconnect.github.io/openconnect-gui/)  
for android refer to [OpenConnect](https://play.google.com/store/apps/details?id=app.openconnect" \t "_blank)  
for iOS refer to [Cisco AnyConnect](https://apps.apple.com/us/app/cisco-anyconnect/id1135064690)

# ocpasswd command

this command is used to make user and chage state of that.

**-c** file, **--passwd**=file

Password file.

**-g** string, **--groupname**=string

User's group name.

**-d**, **--delete**

Delete user.

Removes the specified user from the password file

**-l**, **--lock**

Lock user.

Prevents the specified user from logging in

**-u**, **--unlock**

Unlock user.

# Debian os

1. apt install ocserv
2. systemctl status ocserv