

vpnc is supposed to work with:

- Cisco VPN concentrator 3000 Series
- Cisco IOS routers
- Cisco PIX / ASA Zecurity Appliances
- Juniper/Netscreen

Contents

- [1 This describes how to use DD-WRT to connect to a Cisco VPN Concentrator using vpnc without auto-reconnect and without connect on startup](#)
- [2 This describes how to use DD-WRT to connect to a Cisco VPN Concentrator using vpnc with auto-reconnect](#)
- [3 Multi-Site VPN](#)
 - ◆ [3.1 iptables startup condition](#)
- [4 FAQ Frequently asked Questions](#)

This describes how to use DD-WRT to connect to a Cisco VPN Concentrator using vpnc without auto-reconnect and without connect on startup

This "script" without reconnect is mainly for people to test vpnc and find out the correct settings before they use the script with auto-reconnect. If you want your router to automatically reconnect on connection loss see further down

There is currently no gui for this, but don't worry, it won't be complicated. This script also automatically reconnects, if your vpn connection gets disconnected

To let your router connect to the vpn concentrator and share the connection with its clients follow the steps below:

1. You need to flash DD-WRT VPN build after 08/18/07 (I recommend latest v24 vpn).
2. Paste the code below into a text editor and adjust line 2 - 6.
3. Open a Webbrowser, type the IP of your router, then go to Administration -> Commands
4. Paste the code adjusted in Step 2 into the commands box, then click 'Save Startup'
5. Reboot your router
6. Log in via telnet and enter: `vpnc /tmp/etc/vpnc/vpn.conf`
7. To share the vpn tunnel with the connected pc's, enter the following via telnet:

```
iptables -A FORWARD -o tun0 -j ACCEPT
iptables -A FORWARD -i tun0 -j ACCEPT
iptables -t nat -A POSTROUTING -o tun0 -j MASQUERADE
```

I added also the next rule to share the tunnel tun0 with the LAN connected in br0 interface and added the DNS and WIN servers from my VPN to the local computer connected to LAN:

```
iptables -A FORWARD -i br0 -j ACCEPT
```

To disconnect the tunnel, type `vpnc-disconnect`. Step 6+7 have to be redone after every disconnect or reboot

VPNC

If you have any comment or question on this, you can PM me in the DD-WRT Forum (Username: alain)

```
#!/bin/sh
vpn_concentrator="host" ##enter ip or hostname of your Ipsec vpn concentrator
vpn_groupname="grid" ##enter the group name here
vpn_grouppasswd="grppasswd" ##enter the group password here
vpn_username="username" ##enter your username here
vpn_password="password" ##enter your password here

#--do not edit this--
#Written by Alain R. (alainr /A*T/ gmx. de) 27.Sep.2007
vpnc-disconnect
rm -f /tmp/etc/vpnc/vpn.conf
mkdir /tmp/etc/vpnc
echo "
IPSec gateway $vpn_concentrator
IPSec ID $vpn_groupname
IPSec secret $vpn_grouppasswd
Xauth username $vpn_username
Xauth password $vpn_password
" >> /tmp/etc/vpnc/vpn.conf
```

If your VPN also requires your NTDomain to be submitted, modify the above 2 sections to include: (This also applies to the 'persistent' section below)
under "vpn_password"

```
vpn_domain="domain" ##enter your NTDomain here
```

and under "Xauth password \$vpn_password" add:

```
Domain $vpn_domain
```

This describes how to use DD-WRT to connect to a Cisco VPN Concentrator using vpnc with auto-reconnect

This script does not give out any (useful) error messages. If you have trouble establishing a tunnel, first look at the script without reconnect

There is currently no gui for this, but don't worry, it won't be complicated. This script also automatically reconnects, if your vpn connection gets disconnected

To let your router connect to the vpn concentrator and share the connection with its clients follow the steps below:

1. You need to flash DD-WRT VPN build after 08/18/07 (I recommend latest v24 vpn).
2. Paste the code below into a text editor and adjust line 5 - 11.
3. Open a Webbrowser, type the IP of your router, then go to Administration -> Commands
4. Paste the code adjusted in Step 2 into the commands box, then click 'Save Startup'
5. Reboot your router

If everything worked fine, your router has now established a vpn tunnel.

This describes how to use DD-WRT to connect to a CiscoVPN Concentrator using vpnc without auto-reconnect

VPNC

If you have any comment or question on this, you can PM me in the DD-WRT Forum (Username: alain)

```
mkdir /tmp/etc/vpnc
rm -f /tmp/etc/vpnc/vpnc.sh
echo '
#!/bin/sh
vpn_concentrator="host" ##enter ip or hostname of your Ipsec vpn concentrator
vpn_keepalive_host1="keepalive1" ##enter the ip or hostname of a computer that is only rea
vpn_keepalive_host2="keepalive2" ##enter the ip or hostname of a computer that is only rea
vpn_groupname="grid" ##enter the group name here
vpn_grouppasswd="grppasswd" ##enter the group password here
vpn_username="username" ##enter your username here
vpn_password="password" ##enter your password here

#--do not edit this--
#Written by Alain R. 28.Sep.2007
vpnc-disconnect
rm -f /tmp/etc/vpnc/vpn.conf
echo "
IPSec gateway $vpn_concentrator
IPSec ID $vpn_groupname
IPSec secret $vpn_grouppasswd
Xauth username $vpn_username
Xauth password $vpn_password
" >> /tmp/etc/vpnc/vpn.conf

pingtest1 () {
ping -q -c1 $param1 >> /dev/null
if [ "$?" == "0" ]; then
    echo 0 #reachable

else
    echo 1 #not reachable
fi
}

pingtest2 () {
ping -q -c2 $param2 >> /dev/null
if [ "$?" == "0" ]; then
    echo 0 #reachable

else
    echo 1 #not reachable
fi
}

while [ true ]; do
param1=$vpn_concentrator;
if [ "`pingtest1`" == "0" ]; then #Vpn concentrator reachable
doloop=1;
while [ $doloop -gt 0 ]; do
param1=$vpn_keepalive_host1;

if [ "`pingtest1`" == "0" ]; then
sleep 300
else
param2=$vpn_keepalive_host2;
if [ "`pingtest2`" == "0" ]; then
sleep 300
else
doloop=0;
```

This describes how to use DD-WRT to connect to a CiscoVPN Concentrator using vpnc with auto-reconnect

VPNC

```
vpnc-disconnect
vpnc /tmp/etc/vpnc/vpn.conf --dpd-idle 0
sleep 1
if [ "`pingtest1`" != "0" ]; then
    sleep 10
fi
tundev="`ifconfig |grep tun |cut -b 1-4`"
iptables -A FORWARD -o $tundev -j ACCEPT
iptables -A FORWARD -i $tundev -j ACCEPT
iptables -t nat -A POSTROUTING -o $tundev -j MASQUERADE
sleep 9
fi
done
else
sleep 10;
fi
done

return 0;
' >> /tmp/etc/vpnc/vpnc.sh
chmod 700 /tmp/etc/vpnc/vpnc.sh
/tmp/etc/vpnc/vpnc.sh&
```

Multi-Site VPN

There was a problem with the previous version of this script pid was not set correctly and the iptables were not updated properly upon termination. If you used it please copy and paste this new version.

What happens if your company has more than one location that you need access to? Use this:

```
mkdir /tmp/etc/vpnc

# get rid of left over scripts
if [ -f /tmp/etc/vpnc/vpnc-tun0.sh ]; then
    rm -f /tmp/etc/vpnc/vpnc-tun0.sh
fi
if [ -f /tmp/etc/vpnc/vpnc-tun1.sh ]; then
    rm -f /tmp/etc/vpnc/vpnc-tun1.sh
fi

cp /etc/resolv.conf /tmp/etc/resolv.conf

# get rid of any leftover connections
vpnc-disconnect

echo '
#!/bin/sh
vpnc_interface="tun0"
vpnc_concentrator="xxx"
vpnc_keepalive_host="xxx"
vpnc_groupname="xxx"
vpnc_grouppasswd="xxx"
vpnc_username="xxx"
vpnc_password="xxx"
vpnc_domain=""
vpnc_nat_host="xxx"
```

VPNC

```
# remove any old config files
rm -f /tmp/etc/vpnc/vpnc-tun0.conf

# create the new config file
echo "
Local Port 0
Interface name ${vpnc_interface}
IPSec gateway ${vpnc_concentrator}
IPSec ID ${vpnc_groupname}
IPSec secret ${vpnc_grouppasswd}
Xauth username ${vpnc_username}
Xauth password ${vpnc_password}
Domain ${vpnc_domain}
" > /tmp/etc/vpnc/vpnc-tun0.conf

source /tmp/etc/vpnc/vpnc-manager.sh

return 0;
' >> /tmp/etc/vpnc/vpnc-tun0.sh

echo '
#!/bin/sh
vpnc_interface="tun1"
vpnc_concentrator="xxx"
vpnc_keepalive_host="xxx"
vpnc_groupname="xxx"
vpnc_grouppasswd="xxx"
vpnc_username="xxx"
vpnc_password="xxx"
vpnc_nat_host="xxx"

# remove any old config files
rm -f /tmp/etc/vpnc/vpnc-tun1.conf

# create the new config file
echo "
Local Port 0
Interface name ${vpnc_interface}
IPSec gateway ${vpnc_concentrator}
IPSec ID ${vpnc_groupname}
IPSec secret ${vpnc_grouppasswd}
Xauth username ${vpnc_username}
Xauth password ${vpnc_password}
" > /tmp/etc/vpnc/vpnc-tun1.conf

source /tmp/etc/vpnc/vpnc-manager.sh

return 0;
' >> /tmp/etc/vpnc/vpnc-tun1.sh

echo '
#!/bin/sh

# Some timing parameters, adjust to your preference
sleep_short=10
sleep_long=300
pid=0

while [ true ]; do
    loss=`ping -q -c3 ${vpnc_keepalive_host} | awk "NR == 4 { print \\$7 }"`
    if [ ${loss} != "100%" ]; then
        sleep ${sleep_long}
    fi
done
```

VPNC

```
else
    if [ ${pid} != "0" ]; then
        echo "Killing daemon with pid: ${pid}"
        # kill the daemon
        kill ${pid}
        # reverse the iptables rules
        iptables -D FORWARD -o ${vpnc_interface} -j ACCEPT
        iptables -D FORWARD -i ${vpnc_interface} -j ACCEPT
        iptables -t nat -D POSTROUTING -o ${vpnc_interface} -j MASQUERADE
        # Tunnel nat
        iptables -D PREROUTING -t nat -i ${vpnc_interface} -p tcp --dport 20 -j DNAT --to-destination ${vpnc_nat_host}
        iptables -D PREROUTING -t nat -i ${vpnc_interface} -p tcp --dport 21 -j DNAT --to-destination ${vpnc_nat_host}
        iptables -D PREROUTING -t nat -i ${vpnc_interface} -p tcp --dport 3389 -j DNAT --to-destination ${vpnc_nat_host}
        iptables -D PREROUTING -t nat -i ${vpnc_interface} -p tcp --dport 5900 -j DNAT --to-destination ${vpnc_nat_host}
        iptables -D FORWARD -p tcp -d ${vpnc_nat_host} --dport 20 -j ACCEPT
        iptables -D FORWARD -p tcp -d ${vpnc_nat_host} --dport 21 -j ACCEPT
        iptables -D FORWARD -p tcp -d ${vpnc_nat_host} --dport 3389 -j ACCEPT
        iptables -D FORWARD -p tcp -d ${vpnc_nat_host} --dport 5900 -j ACCEPT
    fi

    # Establish the link
    vpnc /tmp/etc/vpnc/vpnc-${vpnc_interface}.conf --dport-idle 0

    # Record the process id
    pid=`ps | grep "vpnc .*${vpnc_interface}" | awk "{print \\$1}"`
    sleep 1

    # Ping the remote host to see if we are live
    loss=`ping -q -c3 ${vpnc_keepalive_host} | awk "NR == 4 { print \\$7 }"`
    if [ ${loss} == "100%" ]; then
        sleep ${sleep_short}
    fi

    # Make sure we can talk with the interface from within the LAN
    iptables -A FORWARD -o ${vpnc_interface} -j ACCEPT
    iptables -A FORWARD -i ${vpnc_interface} -j ACCEPT
    iptables -t nat -A POSTROUTING -o ${vpnc_interface} -j MASQUERADE

    # Tunnel nat, adjust to your preference
    iptables -A PREROUTING -t nat -i ${vpnc_interface} -p tcp --dport 20 -j DNAT --to-destination ${vpnc_nat_host}
    iptables -A PREROUTING -t nat -i ${vpnc_interface} -p tcp --dport 21 -j DNAT --to-destination ${vpnc_nat_host}
    iptables -A PREROUTING -t nat -i ${vpnc_interface} -p tcp --dport 3389 -j DNAT --to-destination ${vpnc_nat_host}
    iptables -A PREROUTING -t nat -i ${vpnc_interface} -p tcp --dport 5900 -j DNAT --to-destination ${vpnc_nat_host}
    iptables -I FORWARD -p tcp -d ${vpnc_nat_host} --dport 20 -j ACCEPT
    iptables -I FORWARD -p tcp -d ${vpnc_nat_host} --dport 21 -j ACCEPT
    iptables -I FORWARD -p tcp -d ${vpnc_nat_host} --dport 3389 -j ACCEPT
    iptables -I FORWARD -p tcp -d ${vpnc_nat_host} --dport 5900 -j ACCEPT
    # fix the resolv.conf, bug in vpnc where it ignores DNSUpdate
    cat /tmp/etc/resolv.conf > /etc/resolv.conf
    sleep ${sleep_short}
fi

done
' >> /tmp/etc/vpnc/vpnc-manager.sh

# Set the permissions
chmod 700 /tmp/etc/vpnc/vpnc-tun0.sh
chmod 700 /tmp/etc/vpnc/vpnc-tun1.sh
chmod 700 /tmp/etc/vpnc/vpnc-manager.sh

# Fire off the connectors
/tmp/etc/vpnc/vpnc-tun0.sh&
/tmp/etc/vpnc/vpnc-tun1.sh&
```

VPNC

It is important to note the ping line for the keepalive host assumes the loss percentage is on line 4 of the output column 7, if your routers output of ping looks different you need to adjust that line.

iptables startup condition

For some reason on the newer builds iptables will get flushed after the startup script is executed. Just put in a sleep on top of the vpnc manager sh, make it sleep for 10 seconds.

FAQ Frequently asked Questions

- Can some attacker steal my VPN password

Yes, if you don't secure your router...

- Is it possible to get this to work when using the router as a wireless repeater?

Yes, I use it here on a wireless repeater. If you experience problems, flash latest v24 and follow [Wlan Repeater](#).

- I cannot connect if my password contains ", \$, `, or \.

The reason is that the shell always interprets these characters (even between quotation marks). If your password contains such characters, just put a \ (escape character) before each of the above characters. If your password is for example abc\$123xyz you need to enter abc\\$123\xyz

- Can I use split-tunneling to access my local network (and local internet connection) while still accessing my corporate network via VPN tunnel?

Yes. On a Windows machine (which I have tested), you need to edit the TCP/IP configuration and explicitly identify your DNS servers via IP. Add in your corporate DNS server first in the list, then your ISP DNS servers. If you access corporate SMB network shares, be sure to add in your corporate WINS server.

For split tunneling, use the following code:

```
#!/bin/sh
# This is a wrapper for the vpnc-script overriding some variables needed
# for setting up split-tunneling
# this effectively disables changes to /etc/resolv.conf
INTERNAL_IP4_DNS=

# This sets up split networking regardless of the concentrators specifications.
# You can add as many routes as you want, but you must set the counter
# CISCO_SPLIT_INC accordingly. All requests to IP ranges NOT listed
# in the code below will NOT go through the VPN tunnel.

CISCO_SPLIT_INC=2
CISCO_SPLIT_INC_0_ADDR=147.0.0.0 #IP range to go into first tunnel
CISCO_SPLIT_INC_0_MASK=255.0.0.0 #Subnet Mask for first tunnel
CISCO_SPLIT_INC_0_MASKLEN=8      #Mask length
CISCO_SPLIT_INC_0_PROTOCOL=0
CISCO_SPLIT_INC_0_SPORT=0
```

VPNC

```
CISCO_SPLIT_INC_0_DPORT=0
CISCO_SPLIT_INC_1_ADDR=172.0.0.0      #IP range to go into the second tunnel
CISCO_SPLIT_INC_1_MASK=255.0.0.0     #Subnet mask
CISCO_SPLIT_INC_1_MASKLEN=8          #Mask length
CISCO_SPLIT_INC_1_PROTOCOL=0
CISCO_SPLIT_INC_1_SPORT=0
CISCO_SPLIT_INC_1_DPORT=0

# run the original script
. /etc/vpnc/vpnc-script
```

An example of the whole re-connect script with the split-tunnel built in:

```
mkdir /tmp/etc/vpnc
rm -f /tmp/etc/vpnc/vpnc.sh
echo '
#!/bin/sh
vpn_concentrator="host" ##enter ip or hostname of your Ipsec vpn concentrator
vpn_keepalive_host1="keepalive1"      ##enter the ip or hostname of a computer that is only reachable
vpn_keepalive_host2="keepalive2"      ##enter the ip or hostname of a computer that is only reachable
vpn_groupname="grid" ##enter the group name here
vpn_grouppasswd="grppasswd" ##enter the group password here
vpn_username="username" ##enter your username here
vpn_password="password" ##enter your password here

#--do not edit this--
#Written by Alain R. 28.Sep.2007
vpnc-disconnect
rm -f /tmp/etc/vpnc/vpn.conf

echo "
#!/bin/sh
# This is a wrapper for the vpnc-script overriding some variables needed
# for setting up split-tunneling
# this effectively disables changes to /etc/resolv.conf
INTERNAL_IP4_DNS=

# This sets up split networking regardless of the concentrators specifications.
# You can add as many routes as you want, but you must set the counter
# CISCO_SPLIT_INC accordingly. All requests to IP ranges NOT listed
# in the code below will NOT go through the VPN tunnel.

CISCO_SPLIT_INC=2
CISCO_SPLIT_INC_0_ADDR=147.0.0.0 #IP range to go into first tunnel
CISCO_SPLIT_INC_0_MASK=255.0.0.0 #Subnet Mask for first tunnel
CISCO_SPLIT_INC_0_MASKLEN=8      #Mask length
CISCO_SPLIT_INC_0_PROTOCOL=0
CISCO_SPLIT_INC_0_SPORT=0
CISCO_SPLIT_INC_0_DPORT=0
CISCO_SPLIT_INC_1_ADDR=172.0.0.0 #IP range to go into the second tunnel
CISCO_SPLIT_INC_1_MASK=255.0.0.0 #Subnet mask
CISCO_SPLIT_INC_1_MASKLEN=8      #Mask length
CISCO_SPLIT_INC_1_PROTOCOL=0
CISCO_SPLIT_INC_1_SPORT=0
CISCO_SPLIT_INC_1_DPORT=0

# run the original script
. /etc/vpnc/vpnc-script
" > /tmp/etc/vpnc/wrapper.sh

chmod 700 /tmp/etc/vpnc/wrapper.sh
```


VPNC

```
echo "  
IPSec gateway $vpn_concentrator  
IPSec ID $vpn_groupname  
IPSec secret $vpn_grouppasswd  
Xauth username $vpn_username  
Xauth password $vpn_password  
Script /tmp/etc/vpnc/wrapper.sh  
" >> /tmp/etc/vpnc/vpn.conf  
  
pingtest1 () {  
    ping -q -c1 $param1 >> /dev/null  
    if [ "$?" == "0" ]; then  
        echo 0 #reachable  
  
    else  
        echo 1 #not reachable  
    fi  
}  
  
pingtest2 () {  
    ping -q -c2 $param2 >> /dev/null  
    if [ "$?" == "0" ]; then  
        echo 0 #reachable  
  
    else  
        echo 1 #not reachable  
    fi  
}  
  
while [ true ]; do  
    param1=$vpn_concentrator;  
    if [ "`pingtest1`" == "0" ]; then #Vpn concentrator reachable  
        doloop=1;  
        while [ $doloop -gt 0 ]; do  
            param1=$vpn_keepalive_host1;  
  
            if [ "`pingtest1`" == "0" ]; then  
                sleep 300  
            else  
                param2=$vpn_keepalive_host2;  
                if [ "`pingtest2`" == "0" ]; then  
                    sleep 300  
                else  
                    doloop=0;  
                    vpnc-disconnect  
                    vpnc /tmp/etc/vpnc/vpn.conf --dpd-idle 0  
                    sleep 1  
                    if [ "`pingtest1`" != "0" ]; then  
                        sleep 10  
                    fi  
                    tundev="`ifconfig |grep tun |cut -b 1-4`"  
                    iptables -A FORWARD -o $tundev -j ACCEPT  
                    iptables -A FORWARD -i $tundev -j ACCEPT  
                    iptables -t nat -A POSTROUTING -o $tundev -j MASQUERADE  
                    sleep 9  
                fi  
            fi  
        done  
    else
```

VPNC

```
    sleep 10;
    fi

done

return 0;
' >> /tmp/etc/vpnc/vpnc.sh
chmod 700 /tmp/etc/vpnc/vpnc.sh
/tmp/etc/vpnc/vpnc.sh&
```