

This was tested on WRT54GL 1.1 with DD-WRT 2.4 mini.

Contents

- [1 What this setup will do](#)
- [2 Creating a second VLAN](#)
 - ◆ [2.1 Commands to set VLANs](#)
- [3 Setting up the scripts](#)
 - ◆ [3.1 Bringing up the second WAN at startup](#)
 - ◇ [3.1.1 wan2up.startup](#)
 - ◆ [3.2 Setting up the scripts responsible for switching to a specific WAN](#)
 - ◇ [3.2.1 wan1.up](#)
 - ◇ [3.2.2 wan2.up](#)
 - ◆ [3.3 Setting up the actual failover script](#)
 - ◇ [3.3.1 failover.startup](#)
 - ◇ [3.3.2 Optionally add automatic switching back to WAN1 once it becomes available](#)
 - ◆ [3.4 Setting up the script responsible for the Easy Setup button](#)
 - ◇ [3.4.1 chwan.sesbutton](#)
- [4 Final polishing](#)

What this setup will do

Create a second WAN port, so you can hook up another provider for backup, and will change the active provider to the secondary, if ping fails to the provider's gateway. It will ping in a predefined interval. If this is not what you need, then you might try:

- [Dual-WAN for simple round-robin load equalization](#) - explains how you can load balance between two WAN connections
- [Dual WAN with one as standby backup](#) - Explains how to set up a secondary WAN connection for backup

Creating a second VLAN

You'll need to create a new VLAN, called vlan2. Enter in the web interface, Settings/VLANs. Set Port 4 to a new VLAN. Enter in the console, and do the following commands

Commands to set VLANs

```
nvrnm set vlan0ports="1 2 3 5"  
nvrnm set vlan2ports="0 5"  
nvrnm set vlan2hwname=et0  
nvrnm commit
```

On a WRT54GL this will be port 4, if you want port 1 for vlan2 then set vlan2ports to "3 5".

Setting up the scripts

We'll have some scripts, which will go in /jffs/etc/config.

Bringing up the second WAN at startup

wan2up.startup

Now this is for static IP, but you can change this situation by modifying the script for udhpcp

```
#!/bin/sh
WAN2_IFNAME=vlan2
WAN2_IPADDR=192.168.2.2
WAN2_GATEWAY=192.168.2.1
WAN2_NETMASK=255.255.255.0
if [ "$(nvram get wan2_ipaddr)" != "$WAN2_IPADDR" ]; then
    nvram set wan2_ifname=$WAN2_IFNAME
    nvram set wan2_ipaddr=$WAN2_IPADDR
    nvram set wan2_gateway=$WAN2_GATEWAY
    nvram set wan2_netmask=$WAN2_NETMASK
    nvram commit
fi
ifconfig $(nvram get wan2_ifname) up $(nvram get wan2_ipaddr) netmask $(nvram get wan2_netmask)
# We will want to set masquerade on WAN2, otherwise after changing active WAN, we'll remain up in
iptables -t nat -A POSTROUTING -o vlan2 -j MASQUERADE
```

Setting up the scripts responsible for switching to a specific WAN

This file makes the changes necessary to activate WAN1. Change the nameservers to your provider's DNS servers or OpenDNS's.

wan1.up

```
#!/bin/sh
# WAN1
DEV=vlan1
GATEWAY=`nvram get wan_gateway`
DNS1=193.226.43.33
DNS2=193.230.175.1

nvram set wan_dns="$DNS1 $DNS2"
ip route delete default
# ip route delete default
ip route add default via $GATEWAY dev $DEV
echo "nameserver $DNS1" >/tmp/resolv.dnsmasq
echo "nameserver $DNS2" >>/tmp/resolv.dnsmasq
pr="$(ps|awk '/dnsmasq/ {print $1}')"
kill -9 $pr
dnsmasq --conf-file=/tmp/dnsmasq.conf
```

As you probably already figured out, this will make the changes necessary to activate WAN2 :)

wan2.up

```
#!/bin/sh
# WAN2
DEV=vlan2
GATEWAY=`nvram get wan2_gateway`
DNS1=193.226.43.33
DNS2=193.230.175.1

nvram set wan_dns="$DNS1 $DNS2"
ip route delete default
# ip route delete default
ip route add default via $GATEWAY dev $DEV
echo "nameserver $DNS1" >/tmp/resolv.dnsmasq
echo "nameserver $DNS2" >>/tmp/resolv.dnsmasq
pr="$(ps|awk '/dnsmasq/ {print $1}')"
kill -9 $pr
dnsmasq --conf-file=/tmp/dnsmasq.conf
```

Setting up the actual failover script

Now, this is the core of our goal, this script will monitor the active connection, and if connection fails, it will call one of wan1.up or wan2.up to change the active connection.

failover.startup

```
#!/bin/sh
INTERVAL=10
PACKETS=1
USINGWAN=0
WAN1=vlan1
WAN2=vlan2
WAN1GW=`nvram get wan_gateway`
WAN2GW=`nvram get wan2_gateway`

# We'll run this in the intervals given above
while sleep $INTERVAL
do
    # Try to figure out the current route
    TARGET=`ip route | awk '/default via/ {print $3}'`
    # Set the variable, so let the script now which connection is it dealing with
    if [ "$WAN1GW" = "$TARGET" ]; then
        USINGWAN=1;
    else if [ "$WAN2GW" = "$TARGET" ]; then
        USINGWAN=2;
    fi;
fi
# We'll ping as many times the $PACKETS variable tells, and test if we have connection:
RET=`ping -c $PACKETS $TARGET 2>/dev/null | awk '/packets received/ {print $4}'`
# If we don't have connection, change the active WAN port (If there is any loss with mult
if [ "$RET" -ne "$PACKETS" ]; then
    if [ "$USINGWAN" = "1" ]; then
        /jffs/etc/config/wan2.up
        USINGWAN=2
        echo "Changed active WAN port to 2!"
    else
        /jffs/etc/config/wan1.up
```

Dual_WAN_with_failover

```
        USINGWAN=1
        echo "Changed active WAN port to 2!"
    fi
fi
done;
```

Optionally add automatic switching back to WAN1 once it becomes available

When WAN2 is in use as the active WAN port the script will check the status of WAN1 at the regular interval and switch back to it when it becomes available. Useful if WAN2 is a slower or paid-use connection.

Add these lines to failover.startup just before the last line (done;).

```
if [ "$USINGWAN" = "2" ]; then
WAN1STAT=`ping -c $PACKETS $WAN1GW 2>/dev/null | awk '/packets received/ {print $4}'`
if [ "$WAN1STAT" = "$PACKETS" ]; then
/jffs/etc/wan1.up
USINGWAN=1
echo "Changed active WAN port to 1!"
fi
fi
```

Setting up the script responsible for the Easy Setup button

Optionally, if you want to be able to switch the active WAN port with the Easy Setup button, you can add:

chwan.sesbutton

```
#!/bin/sh
USINGWAN=0
WAN1=vlan1
WAN2=vlan2
WAN1GW=`nvram get wan_gateway`
WAN2GW=`nvram get wan2_gateway`

TARGET=`ip route | awk '/default via/ {print $3}'`
if [ "$WAN1GW" = "$TARGET" ]; then
    USINGWAN=1;
else if [ "$WAN2GW" = "$TARGET" ]; then
    USINGWAN=2;
fi;
fi
ip route delete default
if [ "$USINGWAN" = "1" ]; then
    /jffs/etc/config/wan2.up
    USINGWAN=2
    echo "Changed active WAN port to 2!"
else
    /jffs/etc/config/wan1.up
    USINGWAN=1
    echo "Changed active WAN port to 1!"
fi
```

Final polishing

Create above files in /jffs/etc/config and then do a

```
chmod +x /jffs/etc/config/*
```

If you made everything as written above right, you should reboot the router and enjoy!