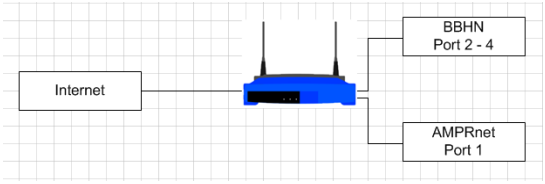
Runbook AMPRnet routing SM7 BBHN

This procedure will route AMPRnet IP subnets, or single IPs, using Policy Based Routing over Vtund tunnel to the BBHN node in different scenarios.

The BBHN node will get switchport 1 dedicated to AMPRnet and the statuspage on the web will show the AMPRnet IP (not in the external firewall scenario).

Here we´ll show you three different, but possible setups that you might encounter.

Directly on the Vtund node only



1. Configure AMPRnet interface and set IP address in **etc/config/network**

```
#### VLAN configuration
config switch eth0
        option vlan0    "2 3 4 5*"
        option vlan1    "0 5"
        option vlan2    "1 5"


#### LAN configuration
config interface ampr
        option ifname           "eth0.2"
        option proto            static
        option ipaddr           44.140.x.x
        option netmask          255.255.x.x
        option dns              "44.140.x.x"
```

2. Configure OLSR to HNA4 publish the new AMPRnet IP network in:
   /etc/config/olsr.conf
   /etc/config.mesh/olsr.conf

```
Hna4
{
# Internet gateway
# 0.0.0.0   0.0.0.0
# specific small networks reachable through this node
# 15.15.0.0 255.255.255.0
44.140.x.x 255.255.x.x
}
```

3. Place **mips-vtund-nossl** and **vtund.conf** in /

Email: sm7i.ham@gmail.com
Packet: sm7i@sk7es.#m.swe.eu

4. Configure **vtund.conf** to match serversettings (in reverse)

```
options {
port 5000;          # Listen on this port.

# Syslog facility
syslog       daemon;

# Path to various programs
ppp          /usr/sbin/pppd;
ifconfig     /sbin/ifconfig;
route        /sbin/route;
firewall     /sbin/ipchains;
ip           /sbin/ip;
}

# Default session options
default {
compress no;         # Compression is off by default
speed 0;             # By default maximum speed, NO shaping
}

#################################################################
nodename-x {
passwd  some-secret-pw;
type  tun;           # IP tunnel
proto tcp;           # UDP protocol
#  comp  lzo:9;        # LZO compression level 9
encrypt  no;         # Encryption
keepalive yes;       # Keep connection alive

up {
ifconfig "%% 172.16.20.x pointopoint 172.16.20.y mtu 1450";
ifconfig "%% multicast";
} ;
}
```

5. Chenge exeflag: **chmod 755 /mips-vtund-nossl**

6. Install kmod-tun package: **ipkg install kmod-tun_2.4.34-brcm-1_mipsel.ipk**

7. Install IP2 package: **ipkg install ip_2.6.20-070313-1_mipsel.ipk**

8. Create **mkdir /etc/iproute2** folder

9. Create new table: **vi /etc/iproute2/rt_tables**

10. Populate **rt_tables** with:

```
255             local
254             main
253             default
0               unspec
10              ampr
```


Email: sm7i.ham@gmail.com
Packet: sm7i@sk7es.#m.swe.eu

11. Edit **/etc/rc.d/S50httpd**

```
#!/bin/sh /etc/rc.common
# Copyright (C) 2006 OpenWrt.org
START=50

start() {
        include /lib/network
        scan_interfaces
        config_get ifname wan hostname
        [ -d /www ] && httpd -p 8080 -h /www -r ${hostname:-OpenWrt}

# firewall rules
        iptables -D FORWARD 11
        iptables -I FORWARD -i tun0 -o eth0.0 -j ACCEPT
        iptables -I FORWARD -i eth0.0 -o tun0 -j ACCEPT
        iptables -I FORWARD -i tun0 -o wl0 -j ACCEPT
        iptables -I FORWARD -i wl0 -o tun0 -j ACCEPT
        iptables -I FORWARD -i tun0 -o eth0.2 -j ACCEPT
        iptables -I FORWARD -i eth0.2 -o tun0 -j ACCEPT

        /mips-vtund-nossl -f /vtund.conf nodename-x serverIP
        sleep 5
        /usr/sbin/ip rule add fwmark 1 table ampr
        /usr/sbin/ip route add default via 172.16.20.x dev tun0 table ampr metric 1
        /usr/sbin/ip rule add from x.x.x.x/x table ampr
        iptables -A forwarding_rule -o tun0 -j ACCEPT
        /usr/sbin/ip route flush cache
        iptables -I INPUT 1 -i tun0 -p udp --dport 53 -j DROP

}

stop() {
        killall httpd
}
```

12. Edit **/www/cgi-bin/status**

```
if(system "ifconfig br-lan >/dev/null 2>&1") # lan not bridged
{
   # show the wifi address
   ($ip, $mask, $bcast, $net, $cidr) = get_ip4_network("wl0");
   $cidr = "/ $cidr" if $cidr;
   $str  = "<th align=right><nobr>WiFi address</nobr></th><td>$ip
<small>$cidr</small><br>";
   $str .= "<small><nobr>" . get_ip6_addr("wl0") . "</nobr></small></td>";
   push @col1, $str;

# show the ampr address
   ($ip, $mask, $bcast, $net, $cidr) = get_ip4_network("eth0.2");
   $cidr = "/ $cidr" if $cidr;
   $str  = "<th align=right><nobr>AMPRnet address</nobr></th><td>$ip
<small>$cidr</small><br>";
   $str .= "<small><nobr>" . get_ip6_addr("eth0.2") . "</nobr></small></td>";
   push @col1, $str;
```

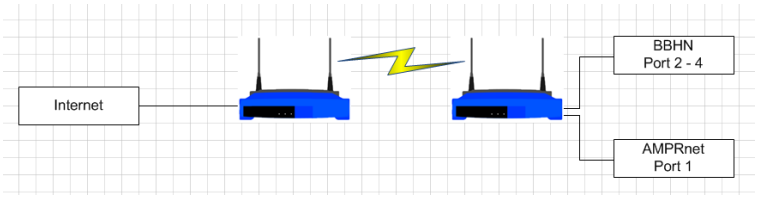

Email: sm7i.ham@gmail.com
Packet: sm7i@sk7es.#m.swe.eu

```
# find out if the browser is on this node's lan
# if not, hide the local network details
($ip, $mask, $bcast, $net, $cidr) = get_ip4_network("eth0.0");
```

13. Reboot

Email: sm7i.ham@gmail.com
Packet: sm7i@sk7es.#m.swe.eu

On WiFi connected node behind Vtund node



1. Configure AMPRnet interface and set IP address in **etc/config/network** on the WiFi connected node

```
#### VLAN configuration
config switch eth0
        option vlan0    "2 3 4 5*"
        option vlan1    "0 5"
        option vlan2    "1 5"


#### LAN configuration
config interface ampr
        option ifname               "eth0.2"
        option proto                static
        option ipaddr               44.140.x.x
        option netmask              255.255.x.x
        option dns                  "44.140.x.x"
```

2. Configure OLSR to HNA4 publish the new AMPRnet IP network on the WiFi connected node in:
   /etc/config/olsr.conf
   /etc/config.mesh/olsr.conf

```
Hna4
{
# Internet gateway
# 0.0.0.0   0.0.0.0
# specific small networks reachable through this node
# 15.15.0.0 255.255.255.0
44.140.x.x 255.255.x.x
}
```

3. Edit **/etc/rc.d/S50httpd** on the WiFi connected node

```
#!/bin/sh /etc/rc.common
# Copyright (C) 2006 OpenWrt.org
START=50

start() {
        include /lib/network
        scan_interfaces
        config_get ifname wan hostname
        [ -d /www ] && httpd -p 8080 -h /www -r ${hostname:-OpenWrt}

# firewall rules
        iptables -D FORWARD 11
```


Email: sm7i.ham@gmail.com
Packet: sm7i@sk7es.#m.swe.eu

```
        iptables -I FORWARD -i wl0 -o eth0.2 -j ACCEPT
        iptables -I FORWARD -i eth0.2 -o wl -j ACCEPT

}

stop() {
        killall httpd
}
```

4. Edit **/etc/rc.d/S50httpd** on the Vtund node

```
#!/bin/sh /etc/rc.common
# Copyright (C) 2006 OpenWrt.org
START=50

start() {
        include /lib/network
        scan_interfaces
        config_get ifname wan hostname
        [ -d /www ] && httpd -p 8080 -h /www -r ${hostname:-OpenWrt}

# firewall rules
        iptables -D FORWARD 11
        iptables -I FORWARD -i tun0 -o eth0.0 -j ACCEPT
        iptables -I FORWARD -i eth0.0 -o tun0 -j ACCEPT
        iptables -I FORWARD -i tun0 -o wl0 -j ACCEPT
        iptables -I FORWARD -i wl0 -o tun0 -j ACCEPT

        /mips-vtund-nossl -f /vtund.conf nodename-x serverIP
        sleep 5
        /usr/sbin/ip rule add fwmark 1 table ampr
        /usr/sbin/ip route add default via 172.16.20.x dev tun0 table ampr metric 1
        /usr/sbin/ip rule add from x.x.x.x/x table ampr
        iptables -A forwarding_rule -o tun0 -j ACCEPT
        /usr/sbin/ip route flush cache
        iptables -I INPUT 1 -i tun0 -p udp --dport 53 -j DROP

}

stop() {
        killall httpd
}
```

5. Edit **/www/cgi-bin/status** on the WiFi connected node

```
if(system "ifconfig br-lan >/dev/null 2>&1") # lan not bridged
{
   # show the wifi address
   ($ip, $mask, $bcast, $net, $cidr) = get_ip4_network("wl0");
   $cidr = "/ $cidr" if $cidr;
   $str  = "<th align=right><nobr>WiFi address</nobr></th><td>$ip
<small>$cidr</small><br>";
   $str .= "<small><nobr>" . get_ip6_addr("wl0") . "</nobr></small></td>";
   push @col1, $str;

# show the ampr address
```

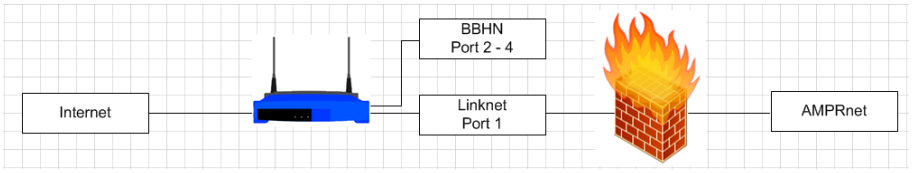
```
    ($ip, $mask, $bcast, $net, $cidr) = get_ip4_network("eth0.2");
    $cidr = "/ $cidr" if $cidr;
    $str  = "<th align=right><nobr>AMPRnet address</nobr></th><td>$ip
<small>$cidr</small><br>";
    $str .= "<small><nobr>" . get_ip6_addr("eth0.2") . "</nobr></small></td>";
    push @col1, $str;

    # find out if the browser is on this node's lan
    # if not, hide the local network details
    ($ip, $mask, $bcast, $net, $cidr) = get_ip4_network("eth0.0");
```

6. Reboot both the Vtund node & WiFi connected node

Email: sm7i.ham@gmail.com
Packet: sm7i@sk7es.#m.swe.eu

Routing through external firewall



1. Configure Linknet interface and set IP address in **etc/config/network** on the node.

   #### VLAN configuration
   config switch eth0
   option vlan0 "2 3 4 5*"
   option vlan1 "0 5"
   option vlan2 "1 5"

   #### LAN configuration
   config interface link
   option ifname "eth0.2"
   option proto static
   option ipaddr 192.168.1.1
   option netmask 255.255.255.252
   option dns "44.140.x.x"

2. Configure any interface, of your choice, on the firewall that will be part of the linknet, to use IP **192.168.1.2** with netmask **255.255.255.252**

   (**Tip**: for ease of administration, name your interfaces or vlans properly according to function)

3. Configure any other interface, of your choice, or create a vlan on the firewall that will be part of the AMPRnet subnet. Best practice is to use the first assigned IP address of your AMPRnet subnet as interface address of your firewall. Do not configure any firewall-policies yet.

4. Configure your firewall to route any packets from source **44.140.x.x / 255.255.x.x** to your node, ie. **192.168.1.1**. Most modern firewalls are capable of doing this using Policy Based Routing.

5. Create a network-object on your firewall to reflect your AMPRnet subnet, but do not configure any kind of NAT.
   The soul purpose of this is to get pure L3 routing running, meaning that the services you provide should be directly avaliable from AMPRnet and Internet using your firewall for security as to which ports are reachable or not.

6. Configure OLSR to HNA4 publish the new AMPRnet IP network on the node in:
   /etc/config/olsr.conf
   /etc/config.mesh/olsr.conf

Email: sm7i.ham@gmail.com
Packet: sm7i@sk7es.#m.swe.eu

```
Hna4
{
# Internet gateway
# 0.0.0.0   0.0.0.0
# specific small networks reachable through this node
# 15.15.0.0 255.255.255.0
44.140.x.x 255.255.x.x
}
```

7. Place **mips-vtund-nossl** and **vtund.conf** in /

8. Configure **vtund.conf** to match serversettings (in reverse)

```
options {
port 5000;          # Listen on this port.

# Syslog facility
syslog        daemon;

# Path to various programs
ppp           /usr/sbin/pppd;
ifconfig      /sbin/ifconfig;
route         /sbin/route;
firewall      /sbin/ipchains;
ip           /sbin/ip;
}

# Default session options
default {
compress no;         # Compression is off by default
speed 0;             # By default maximum speed, NO shaping
}

#####################################################################
nodename-x {
passwd  some-secret-pw;
type  tun;           # IP tunnel
proto tcp;           # UDP protocol
#  comp  lzo:9;        # LZO compression level 9
encrypt  no;         # Encryption
keepalive yes;       # Keep connection alive

up {
ifconfig "%% 172.16.20.x pointopoint 172.16.20.y mtu 1450";
ifconfig "%% multicast";
} ;
}
```

9. Chenge exeflag: **chmod 755 /mips-vtund-nossl**

10. Install kmod-tun package: **ipkg install kmod-tun_2.4.34-brcm-1_mipsel.ipk**

11. Install IP2 package: **ipkg install ip_2.6.20-070313-1_mipsel.ipk**

Email: sm7i.ham@gmail.com
Packet: sm7i@sk7es.#m.swe.eu

12. Create **mkdir /etc/iproute2** folder

13. Create new table: **vi /etc/iproute2/rt_tables**

14. Populate **rt_tables** with:

```
255     local
254     main
253     default
0       unspec
10      ampr
20      link
```

15. Edit **/etc/rc.d/S50httpd**

```
#!/bin/sh /etc/rc.common
# Copyright (C) 2006 OpenWrt.org
START=50

start() {
        include /lib/network
        scan_interfaces
        config_get ifname wan hostname
        [ -d /www ] && httpd -p 8080 -h /www -r ${hostname:-OpenWrt}

# firewall rules
        iptables -D FORWARD 11
        iptables -I FORWARD -i tun0 -o eth0.0 -j ACCEPT
        iptables -I FORWARD -i eth0.0 -o tun0 -j ACCEPT
        iptables -I FORWARD -i tun0 -o wl0 -j ACCEPT
        iptables -I FORWARD -i wl0 -o tun0 -j ACCEPT
        iptables -I FORWARD -i tun0 -o eth0.2 -j ACCEPT
        iptables -I FORWARD -i eth0.2 -o tun0 -j ACCEPT

        /mips-vtund-nossl -f /vtund.conf nodename-x serverIP
        sleep 5
        /usr/sbin/ip rule add fwmark 1 table ampr
        /usr/sbin/ip rule add fwmark 1 table link
        /usr/sbin/ip route add default via 172.16.20.x dev tun0 table ampr metric 1
        /usr/sbin/ip route add 44.140.x.x/x via 192.168.1.2 dev eth0.2 table link metric 1
        /usr/sbin/ip rule add from x.x.x.x/x table ampr
        /usr/sbin/ip rule add to x.x.x.x/x table link
        iptables -A forwarding_rule -o tun0 -j ACCEPT
        /usr/sbin/ip route flush cache
        iptables -I INPUT 1 -i tun0 -p udp --dport 53 -j DROP

}

stop() {
        killall httpd
}
```

16. Reboot the node

Email: sm7i.ham@gmail.com
Packet: sm7i@sk7es.#m.swe.eu

17. Configure necessary firewall-policies according to your needs.

Configure pfSense firewall with Vtund and OLSR to route AMPRnet

(Attention: In the following instructions the BBHN (10.x.x.x/x) network has been removed)

To be written….

Email: sm7i.ham@gmail.com
Packet: sm7i@sk7es.#m.swe.eu