Subject: SOLVED * Veth on startup (again) + bridge config - venet config
Posted by sylvain on Tue, 21 Nov 2006 03:40:49 GMT
View Forum Message <> Reply to Message

Hi,

This is solved. Here is the step I took to solve this problem. I am aware that bugs have been filed and fix my obsolete my configuration.

This is not too intrusive into openVZ tools and configurations. However it will not appeal to people who wants to control their VE ip configuration from the HN.

First, a bridge start script: /etc/init.d/bridge (to link at proper runlevel)

```
#!/bin/sh
#

# Source function library.
. /etc/init.d/functions

BRCTL=brctl
IFCONFIG=ifconfig

if [ ! -x /usr/sbin/$BRCTL ]; then
    echo -n $"/usr/sbin/$BRCTL does not exist."; warning; echo
    exit 0
fi

if [ ! -x /sbin/$IFCONFIG ]; then
    echo -n $"/sbin/$IFCONFIG does not exist."; warning; echo
    exit 0
fi


# Default bridge configuration:
BR_NAME="vzbr0"
BR_ADDRESS="0"
FORWARDING="yes"
PROXY_ARP="yes"

start() {

    echo -n $"Strating bridge $BR_NAME: "

    $BRCTL addbr $BR_NAME
    if [ $? -eq 0 ]; then
 success; echo
    else
 failure; echo; return 1
    fi

    echo -n $"Configuring bridge $BR_NAME: "
    $IFCONFIG $BR_NAME $BR_ADDRESS
    if [ $? -eq 0 ]; then
 success; echo
    else
 failure; echo; return 1
    fi

    if  [ $FORWARDING == "yes" ] ; then
 echo -n $"Configuring bridge $BR_NAME forwading: "
 if [ -f $"/proc/sys/net/ipv4/conf/$BR_NAME/forwarding" ] ; then
     echo 1 > /proc/sys/net/ipv4/conf/$BR_NAME/forwarding
     success; echo
 else
     failure; echo; return 1
 fi
    fi

    if  [ $PROXY_ARP == "yes" ] ; then
 echo -n $"Configuring bridge $BR_NAME proxy arp: "
 if [ -f $"/proc/sys/net/ipv4/conf/$BR_NAME/proxy_arp" ] ; then
     echo 1 > /proc/sys/net/ipv4/conf/$BR_NAME/proxy_arp
     success; echo
 else
     failure; echo; return 1
 fi
    fi

    return 0
}

stop() {

    echo -n $"Stopping bridge $BR_NAME: "

    $BRCTL delbr $BR_NAME
    if [ $? -eq 0 ]; then
 success; echo
    else
 failure; echo; return 1
    fi

    return 0
}
```

```
status() {

    if $BRCTL show 2>/dev/null | grep -q $BR_NAME ; then
 echo $"Bridge $BR_NAME started."
    else
 echo $"Bridge $BR_NAME stopped."
    fi

    return 0
}

restart() {
    stop
    start
}

case "$1" in
    start)
 stop
 start
 RETVAL=$?
 ;;
    stop)
 stop
 RETVAL=$?
 ;;
    restart)
 restart
 RETVAL=$?
 ;;
    status)
 status
 RETVAL=$?
 ;;
    *)
 echo $"Usage: $0 {start|stop|restart|status}"
 exit 1
 ;;
esac

exit $RETVAL
```

I created a new distribution template configuration file: /etc/vz/dists/nonetdevice.conf

```
# distribution configuration that inibits:
```

```
#  * vzctl --addip
#  * vzctl --delip
#  * vzctl --hostname
#
ADD_IP=null.sh
DEL_IP=null.sh
SET_HOSTNAME=null.sh
SET_DNS=set_dns.sh
SET_USERPASS=set_userpass.sh
SET_UGID_QUOTA=set_ugid_quota.sh
POST_CREATE=postcreate.sh
```

with the corresponding /etc/vz/dists/scripts/null.sh script

```
#!/bin/bash

# Null script. It just does not take any actions.
```

Now change the OSTEMPLATE variable for you VE in /etc/vz/conf/X.conf to be OSTEMPLATE="nonetdevice"

I changed a bit /usr/sbin/vznetcfg, in the function init_veth() only

```
function init_veth()
{
 local dev=$1

 /sbin/ifconfig ${dev} up

 if [ "${VZNET_BRIDGE}" != "" ]
 then
  /usr/sbin/brctl addif ${VZNET_BRIDGE} ${dev}
 fi
}
```

and added the following line in /etc/vz/vznet.conf

```
VZNET_BRIDGE=vzbr0
```

The remaining task consists in configuring the network inside your VE. Enter your VE and, regarding the distribution, setup your configuration files for your Ethernet device (/etc/sysconfig/network-scripts/ifcfg-ethX for centos/redhat/fedora, /etc/network/interfaces for debian based)

This does the job as I wanted.