```
#!/bin/sh

clear
echo
"+=====================================================================
=========+"
echo "| This configuration script is for an HPDL380G6, QLogic HBAs, 4
x NICs       |"
echo "+
---------------------------------------------------------------------
----- +"
echo "| Phycial NIC 1 & 2 (vmnic0 & vmnic1) - Service Console &
vMotion            |"
echo "| Phycial NIC 3 & 4 (vmnic2 & vmnic3) - LAN traffic
|"
echo
"+=====================================================================
=========+"
echo " "
echo -e "Do you want to continue? (Y = yes , N = no) : \c"
read contans
if [ $contans = "Y" ]; then
   echo " "
   # Console memory configuration
   # Check to see if the script has been run before

   if [ -e "/tmp/esx.conf.old" ]; then
      echo "Console memory previously amended, no update required."
   else
      echo "Changing Console Memory to 800mb"
      /bin/sed -e 's/boot\/memSize = \"272\"/boot\/memSize = \"800
\"/g' /etc/vmware/esx.conf >> /etc/vmware/esx.conf.new
      mv /etc/vmware/esx.conf /tmp/esx.conf.old
      mv /etc/vmware/esx.conf.new /etc/vmware/esx.conf
      /bin/sed -e 's/uppermem 277504/uppermem 818176/g' -e 's/mem=
272M/mem=800M/g' /boot/grub/grub.conf >> /boot/grub/grub.conf.new
      mv /boot/grub/grub.conf /tmp/grub.conf.old
      mv /boot/grub/grub.conf.new /boot/grub/grub.conf
      echo "Console Memory amendments completed."
      sleep 2
   fi


   # DNS and Gateway configuration
   # Check to see if the script has been run before
   if [ -e "/tmp/resolv.conf.old" ]; then
      echo "DNS settings previously updated, no update required."
   else
      echo "Updating resolv.conf with DNS configuration."
      mv /etc/resolv.conf /tmp/resolv.conf.old
      echo "search yourdomainname.com" >> /etc/resolv.conf
      echo "nameserver 192.168.0.56" >> /etc/resolv.conf
      echo "nameserver 192.168.0.57" >> /etc/resolv.conf
      echo "Updating of resolv.conf completed."
      sleep 2
   fi


   # Enable and configure ntp
   # Check to see if the script has been run before

   if [ -e "/tmp/ntp.conf.old" ]; then
      echo "Time settings previously amended, no update required."
   else
      echo "Enabling NTP client."
      /usr/sbin/esxcfg-firewall --enableService ntpClient
```

```
      echo "NTP client enabled."
      mv /etc/ntp.conf /tmp/ntp.conf.old
      mv /etc/ntp/step-tickers /tmp/step-tickers.old
      echo "Updating ntp.conf file, please wait."
      echo "restrict 127.0.0.1" >> /etc/ntp.conf
      echo "restrict default kod nomodify notrap" >> /etc/ntp.conf
      echo "server 192.168.0.99" >> /etc/ntp.conf
      echo "driftfile /var/lib/ntp/drift"  >> /etc/ntp.conf
      echo "192.168.0.99"  >> /etc/ntp/step-tickers
      echo "Updated ntp.conf file."
      /sbin/chkconfig --level 345 ntpd on
      echo "Restart ntpd daemon."
      /etc/init.d/ntpd restart > nul
      echo "Synchronise the system and hardware clocks."
      /sbin/hwclock --systohc
      echo "Time settings update complete."
      sleep 2
   fi


   ######
   # Qlogic Queue Depth
   if [ -e "/tmp/qlogic.old" ]; then
      echo "QLogic adaptor previously configured, no update
required."
   else
      echo "QLogic configuration in progress, please wait."
      /usr/sbin/esxcfg-module -s ql2xmaxqdepth=64 qla2300_707_vmw.o >
nul
      /usr/sbin/esxcfg-advcfg -s 64 /Disk/SchedNumReqOutstanding >
nul
      /usr/sbin/esxcfg-boot -b > nul
      echo "QLogic adaptor updated."
      sleep 2
      echo "Configuring LUN & Device parameters, please wait."
      # Enable uselunreset and Disable devicereset
      /usr/sbin/esxcfg-advcfg -s 0 /Disk/UseDeviceReset > nul
      /usr/sbin/esxcfg-advcfg -s 1 /Disk/UseLunReset > nul
      echo "Configured LUN & Device parameters."
      echo "Qlogic update applied" >> /tmp/qlogic.old
   fi


   ######
   # Service Console configuration
   # Check to see if the script has been run before

   if [ -e "/tmp/service-console.old" ]; then
      echo "Service console previously configured, no update
required."
   else
      # Add physical NIC2 to vSwitch0
      echo "Adding physical NIC port 2 to Service Console, please
wait."
      /usr/sbin/esxcfg-vswitch -L vmnic1 vSwitch0
      echo "Physical NIC 2 added to vSwitch0."
      echo "Service console configuration script run" >>
/tmp/service-console.old
      sleep 3
   fi


   ######
   # vSwitch0 configuration for VMotion
   # Check to see if the script has been run before
```

```
   if [ -e "/tmp/vmotion.old" ]; then
      echo "vMotion previously configured, no update required."
   else
      # Add vMotion port group to vSwitch0
      echo -e "Please type in the vMotion IP address for this server
: \c"
      read vmotionip
      echo -e "Please type in the vMotion Subnet Mask for this server
: \c"
      read vmotionnetm
      echo -e "Please type in the vMotion Default Gateway for this
server : \c"
      read vmotiongw
      echo "You entered:"
      echo "   $vmotionip"
      echo "   $vmotionnetm"
      echo "   $vmotiongw"
      echo -e "Is this OK? (Y = yes , N = no) : \c"
      read vmotionans
      sleep 2
      if [ $vmotionans = "Y" ]; then
         echo "Adding port group vMotion to vSwitch0, please wait."
         /usr/sbin/esxcfg-vswitch -A vmotion vSwitch0
         sleep 1
         /usr/sbin/esxcfg-vswitch -p vmotion -v 80 vSwitch0
         echo "vMotion Port Group using VLAN 80 added to vSwitch0."
         sleep 2
         /usr/sbin/esxcfg-vmknic -a vmotion -i $vmotionip -n
$vmotionnetm
         sleep 1
         /usr/sbin/esxcfg-route $vmotiongw
         sleep 5
         # Turn on vMotion
         vmware-vim-cmd "hostsvc/vmotion/vnic_set" `esxcfg-vmknic -l
| grep "vmotion" | awk '{print $1}'`
         echo "vMotion now enabled."
         echo "vMotion configuration script run" >> /tmp/vmotion.old
         sleep 2
      else
         rm -f /tmp/vmotion.old > nul
         echo " "
         echo "Re-run this script immediately after to configure
vMotion"
         echo " "
       fi
   fi


   ######
   # vSwitch 1 configuration
   # Check to see if the script has been run before

   if [ -e "/tmp/vswitch1.old" ]; then
      echo "vSwitch 1 previously configured, no update required."
   else

      # Create vswitch1 with physical NIC3 & NIC4 and name it
      echo "Adding physical NIC ports 3 and 4 to vSwitch1, please
wait."
      /usr/sbin/esxcfg-vswitch -a vSwitch1
      /usr/sbin/esxcfg-vswitch -L vmnic2 vSwitch1
      /usr/sbin/esxcfg-vswitch -L vmnic3 vSwitch1
      echo "Physical NIC ports added to vSwitch1."
      sleep 2
      echo "Adding Port Group names to vSwitch1, please wait."
      /usr/sbin/esxcfg-vswitch -A "desktop_lan" vSwitch1
```

```
      /usr/sbin/esxcfg-vswitch -A "server_lan" vSwitch1
      echo "Port Groups added to vSwitch1."
      echo "Assign VLAN IDs to Port Groups on vSwitch1, please wait."
      sleep 2
      /usr/sbin/esxcfg-vswitch -p "desktop_lan" -v 40 vSwitch1
      /usr/sbin/esxcfg-vswitch -p "server_lan" -v 50 vSwitch1
      echo "Assigned VLAN IDs to Port Groups on vSwitch1."
      echo "vSwitch1 configuration script run" >> /tmp/vswitch1.old
      sleep 2
   fi


   ######
   # vSwitch2 for internal traffic only
   # Check to see if the script has been run before

   if [ -e "/tmp/vswitch2.old" ]; then
      echo "vSwitch 2 previously configured, no update required."
   else
      # Create vSwitch2 for internal traffic no NIC assignment
      echo "Creating vSwitch2 for host only traffic only, please
wait."
      /usr/sbin/esxcfg-vswitch -a vSwitch2
      /usr/sbin/esxcfg-vswitch -A internal-host vSwitch2
      sleep 2
      echo "vSwitch2 created."
      echo "vSwitch2 configuration script run" >> /tmp/vswitch2.old
   fi


   ######
   # Enable SSH Client for WinSCP client
   # Check to see if the script has been run before

   if [ -e "/tmp/sshclient.old" ]; then
      echo "SSH Client previously enabled, no update required."
   else
      echo "Enabled the SSH Client, please wait."
      esxcfg-firewall -e sshClient
      echo "SSH Client enabled."
      echo "SSH Client enabled" >> /tmp/sshclient.old
   fi


   ######
   # Reboot server

   echo " "
   echo "This ESX server must be rebooted for all changes to take
effect"
   echo " "
   echo -e "Do you want to reboot now? (Y = yes , N = no) : \c"
   read bootans
   if [ $bootans = "Y" ]; then
       echo "Reboot command issued..."
       sleep 2
       reboot
   else
       echo " "
       echo "Re-configuration tasks completed"
       echo " "
   fi

fi
```