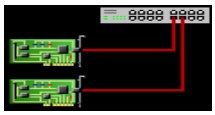
Express-Guide

~to~

Load Balancing the Network Traffic



Ethernet Bonding

Using Multiple NICs on Linux

by, **ABK** ~ http://www.twitter.com/**aBionic**

::Task Detail::

- Load Balancing the Network Traffic on Multiple Ethernet cards attached on a Linux Box.

::Background::

- Linux allows binding multiple NICs into single NIC using special in-built kernel module 'bonding'.
- The behavior of bonded interfaces depend upon 'modes', providing either hot-standby or load-balancing services. Additionally, link integrity monitoring can be performed.
- Settings for 'modes' are:

```
mode=0 (balance-rr)
: Round-Robin Policy {load balancing and fault-tolerance}
mode=1 (active-backup)
: Fault-tolerance, one active at a time
mode=2 (balance-XOR)
: Same slave for same Target {load balancing & fault-tolerance}
mode=3 (broadcast)
: Fault Tolerance
mode=4 (802.3ad)
: IEEE 802.3ad, Dynamic Link Aggregation
mode=5 (balance-tlb)
: Load balancing outgoing traffic
mode=6 (balance-alb)
: Load Balancing In/Out traffic
```

Settings for 'miimon' will be in milliseconds monitoring the traffic.

::Execution Method::

Here, a new Bonded NIC is been configured in Linux-Kernel-based Machines (as it is a Linux Kernel Module) representing the both physical NICs in machine.

It can provide with load-balancing and failure-recovery.
{it's RHEL/Fedora/CentOS sepcific, use them according to your distro}

- Creating new software-level Bonding NIC acting as a master
  - Create a new file '/etc/sysconfig/network-scripts/ifcfg-bond0' if want to name it bond0 with content below; the IP configuration changes as per your network
    - for static IP settings
      - #####start of ifcfg-bond0 content####
        DEVICE=bond0
        IPADDR=192.168.1.3
        NETWORK=192.168.1.1
        NETMASK=255.255.255.0
        USERCTL=no
        BOOTPROTO=static
        ONBOOT=yes
        BONDING_OPTS="mode=balance-tlb miimon=100"
        ####stop of ifcfg-bond0 content########
    - for dynamic IP settings
      - #####start of ifcfg-bond0 content#######
        DEVICE=bond0
        USERCTL=no
        BOOTPROTO=dhcp
        ONBOOT=yes
        BONDING_OPTS="mode=balance-tlb miimon=100"
        ######stop of ifcfg-bond0 content######
    - here, the line with 'BONDING_OPTS' specifies it to be bonded

- Modify the existing NIC configuration's as below
  - say for eth0 settings '/etc/sysconfig/network-scripts/ifcfg-eth0',
  - similar for other NICs configuration files
    - #####start of ifcfg-eth0 content#########
      DEVICE=eth0
      USERCTL=no
      BOOTPROTO=none
      ONBOOT=yes
      MASTER=bond0
      SLAVE=yes
      #######stop of ifcfg-eth0 content#####

- Loading bond driver/module
  - open '/etc/modprobe.conf' in any editor and add following 2 lines to it
    - #####below 2 lines#####
      alias bond0 bonding
      options bond0 mode=balance-alb miimon=100

- Loading all changes made
  - at shell
    #modprobe bonding
    #service network restart
    #less /proc/net/bonding/bond0

::Tools/Technology Used::

Bonding Module of Linux Kernel:
http://www.kernel.org/doc/Documentation/networking/bonding.txt

---

::Inference::

- This is a nice built-in facility provided in Linux to be used.
- Generic Mode for input/output traffic load balancing 'balance-alb'.

::Troubleshooting/Updates::

- ***Problem:*** initial setting gave error 'dev_set_max_address of dev eth0 failed'
  Solution:
  ALB mode implemented requires a suitable HWAddress to be specified; there were already present HWADDRESS entries in all configuration files. Just for a trial removing those lines worked.
