Introduction to MPIO, MCS, Trunking, and LACP

Sam Lee

Version 1.0 (JAN, 2010)

QSAN Technology, Inc.
http://www.QsanTechnology.com
White Paper# **QWP201002-P210C**

Introduction

Many users confuse the terms of MPIO, MC/S, Trunking and LACP. In this document, it will describe these multipath methods. Multipath provides two or more paths for redundancy from server to storage and protects against hardware failures. (e.g.: cable, switch, HBA failures and so on.) In addition, it can also provide higher performance by aggregating multiple connections.

Basically, MPIO, MC/S features come from iSCSI initiator and LACP, Trunking functions must be set in **QSAN** controllers and gigabit switches. We will take some demonstrations about how to use LACP, Trunking function with Microsoft iSCSI initiator. At the same time, it will also increase the redundancy and bandwidth for higher availability.

Environment

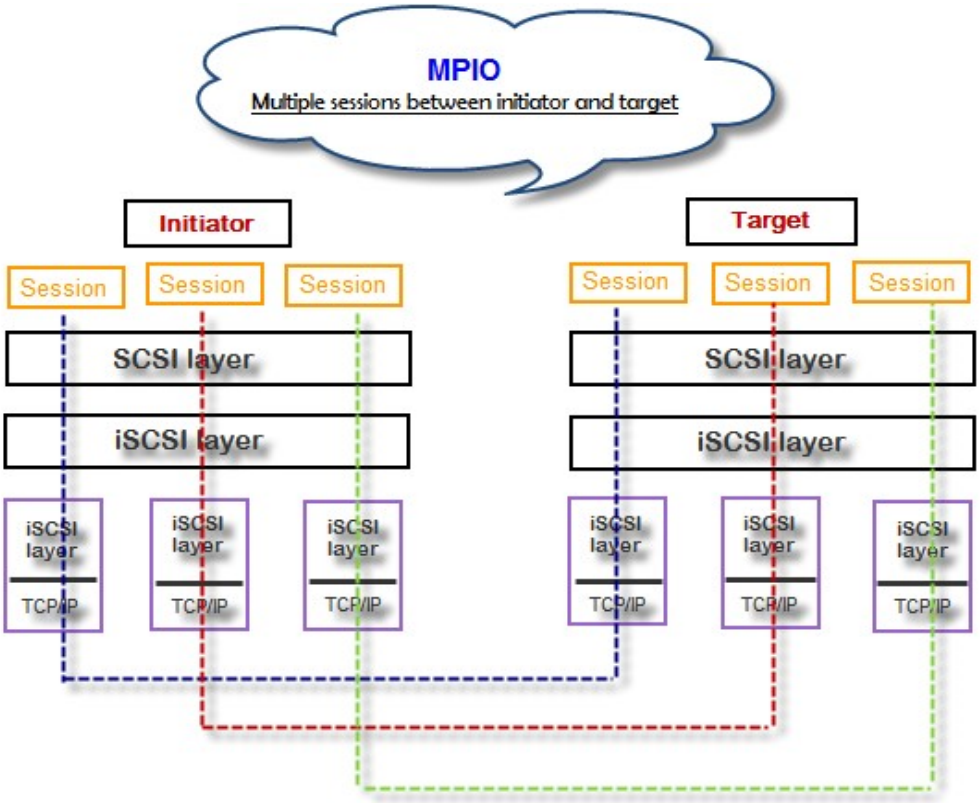
Host OS:	Windows Server 2008 Enterprise edition R2
iSCSI target:	QSAN P210C
RAM:	1GB DDR2-667
Firmware:	1.0.6 (200911113_1500)
iSCSI data port:	192.168.1.1/24; 192.168.2.1/24; 192.168.3.1/24; 192.168.4.1/24
Gigabit Switch:	D-Link DGS-3024

Comparison

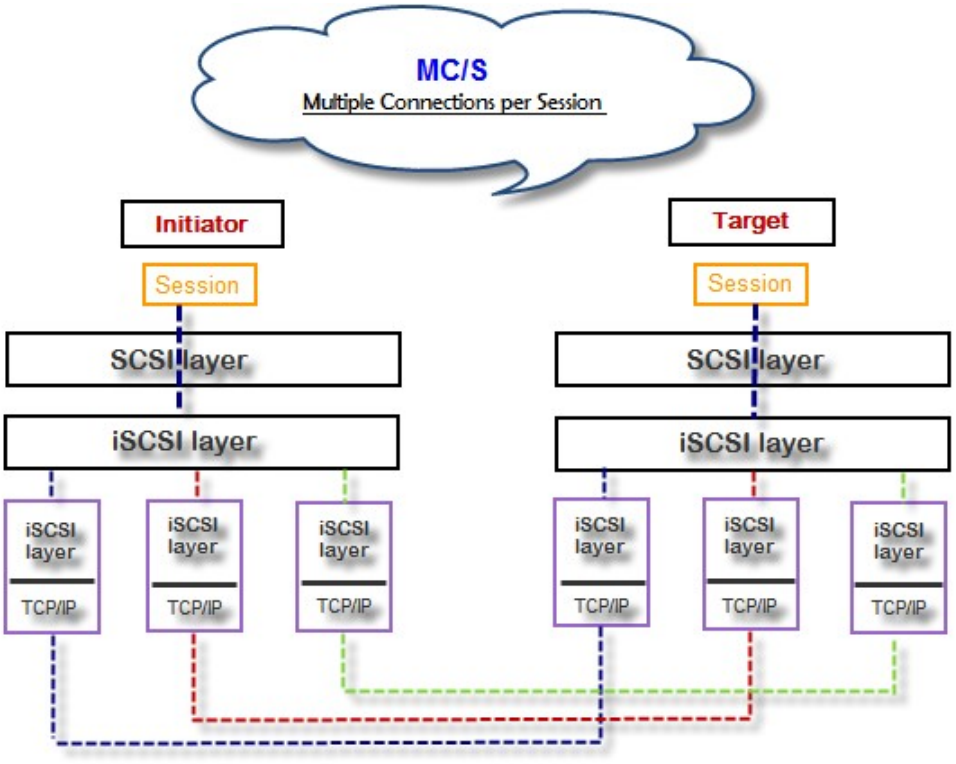
MPIO、MC/S:

These features come from iSCSi initiator. They can be setup from iSCSI initiator to establish redundant paths for sending I/O from the initiator to the target.

1. **MPIO:** In Microsoft Windows server base system, Microsoft MPIO driver allows initiators to login multiple sessions to the same target and aggregate the duplicate devices into a single device. Each session to the target can be established using different NICs, network infrastructure and target ports. If one session fails, then another session can continue processing I/O without interruption to the application.



2. **MC/S:** MC/S (Multiple Connections per Session) is a feature of iSCSI protocol, which allows combining several connections inside a single session for performance and failover purposes. In this way, I/O can be sent on any TCP/IP connection to the target. If one connection fails, another connection can continue processing I/O without interruption to the application.



Difference:

MC/S is implemented on iSCSI level, while MPIO is implemented on the higher level. Hence, all MPIO infrastructures are shared among all SCSI transports, including Fiber Channel, SAS, etc. MPIO is the most common usage across all OS vendors. The primary difference between these two is which level the redundancy is maintained. MPIO creates multiple iSCSI sessions with the target storage. Load balance and failover occurs between the multiple sessions. MC/S creates multiple connections within a single iSCSI session to manage load balance and failover. Notice that iSCSI connections and sessions are different than TCP/IP connections and sessions. The above figures describe the difference between MPIO and MC/S.

There are some considerations when user chooses MC/S or MPIO for multipathing.

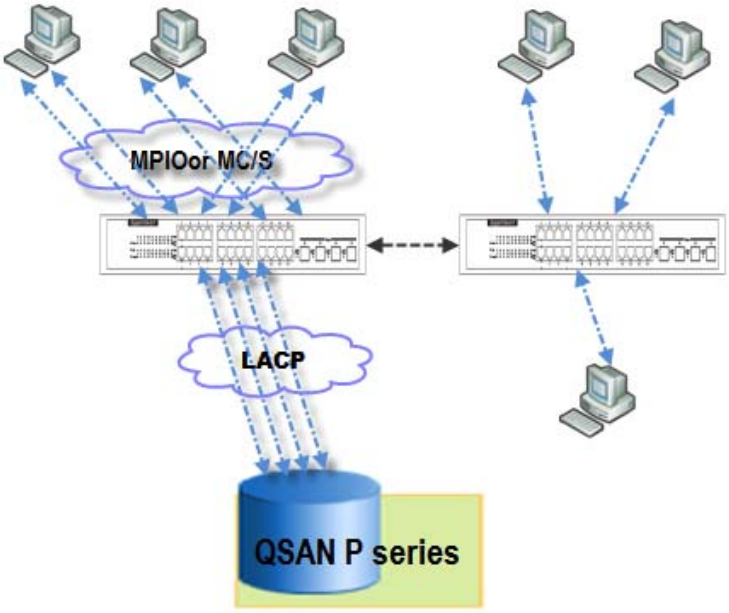
1. If user uses hardware iSCSI off-load HBA, then MPIO is the only one choice.
2. If user needs to specify different load balance policies for different LUNs, then MPIO should be used.
3. If user installs anyone of Windows XP, Windows Vista or Windows 7, MC/S is the only option since Microsoft MPIO is supported Windows Server editions only.
4. MC/S can provide higher throughput than MPIO in Windows system, but it consumes more CPU resources than MPIO.

Link aggregation:

Link aggregation is the technique of taking several distinct Ethernet links to let them appear as a single link. It has a larger bandwidth and provides the fault tolerance ability. Beside the advantage of wide bandwidth, the I/O traffic remains operating until all physical links fail. If any link is restored, it will be added to the link group automatically. **QSAN** implements link aggregation as LACP and Trunking.

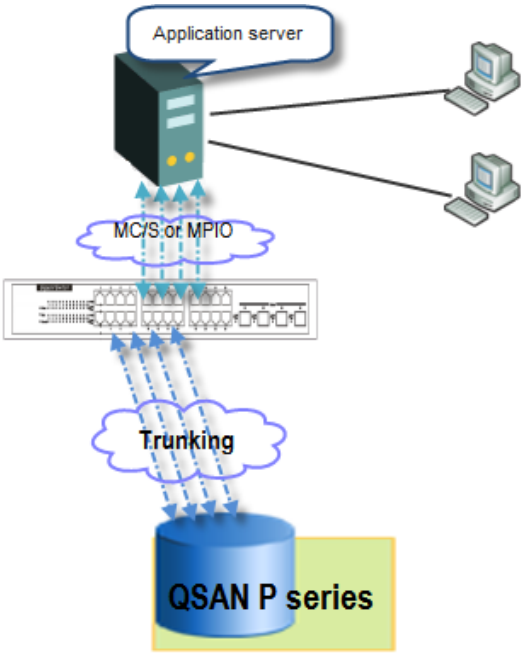
1. **LACP (IEEE 802.3ad):** The Link Aggregation Control Protocol (LACP) is a part of IEEE specification 802.3ad. It allows bundling several physical ports together to form a single logical channel. A network switch negotiates an automatic bundle by sending LACP packets to the peer. Theoretically, LACP port can be defined as active or passive. **QSAN** controller implements it as active mode which means that LACP port sends LACP protocol packets automatically. Please notice that using the same configurations between **QSAN** controller and gigabit switch.

   The usage occasion of LACP:
   A. It's necessary to use LACP in a network environment of multiple switches. When adding new devices, LACP will separate the traffic to each path dynamically.



2. **Trunking (Non-protocol)**: Defines the usage of multiple iSCSI data ports in parallel to increase the link speed beyond the limits of any single port.

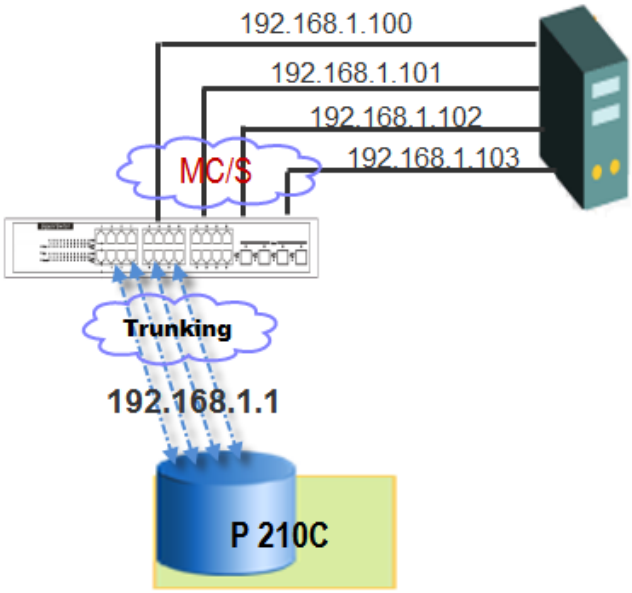
   The usage occasion of Trunking:
   A. This is a simple SAN environment. There is only one switch to connect the server and storage. And there is no extra server to be added in the future.
   B. There is no idea of using LACP or Trunking, uses Trunking first.
   C. There is a request of monitoring the traffic on a trunk in switch.



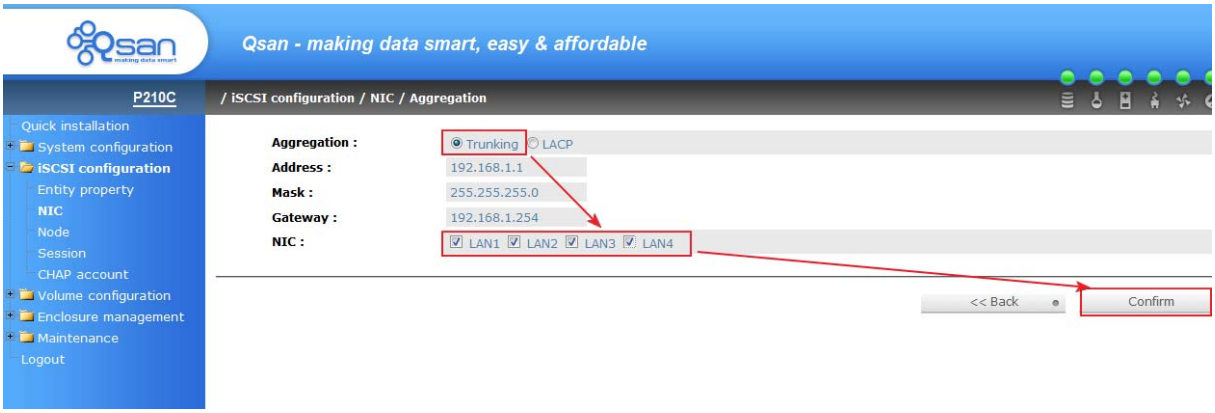
Configuration

Part 1: Using MC/S and Trunking in Windows Server 2008 R2

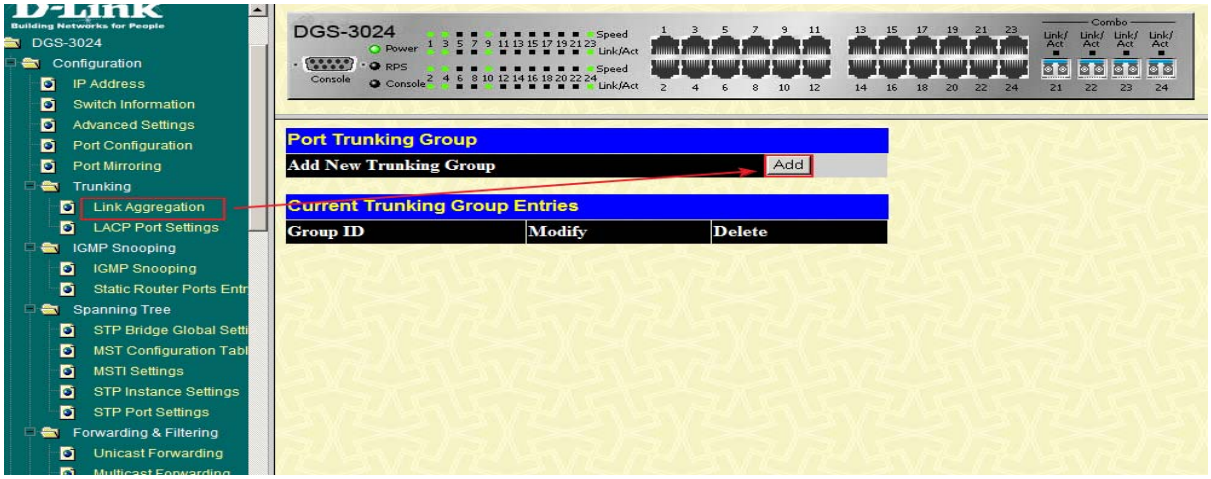
Diagram :



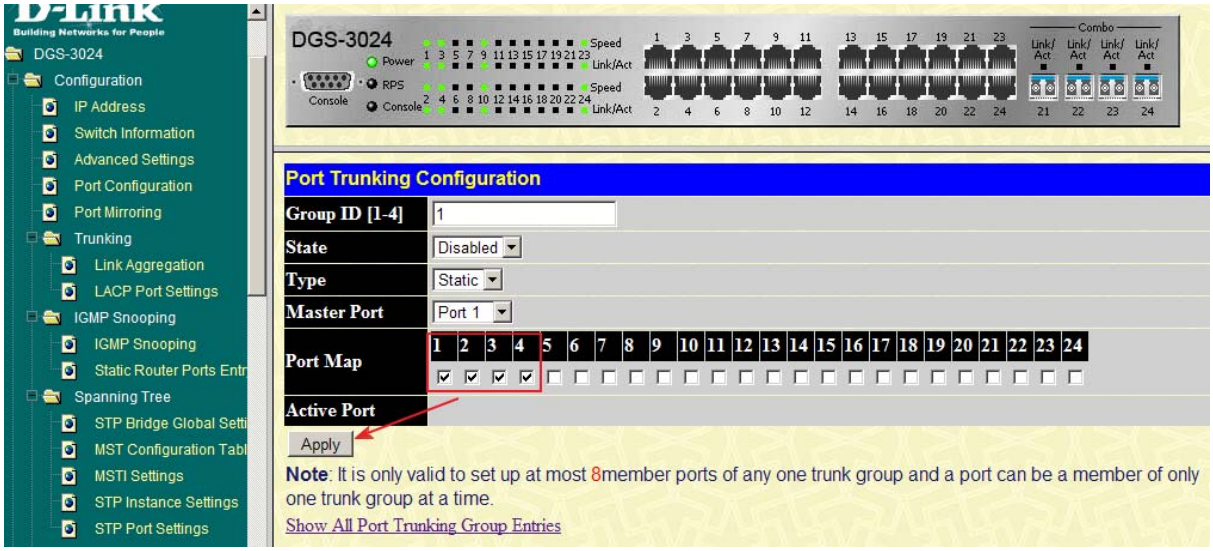
1. Set Trunking on **P210C**.



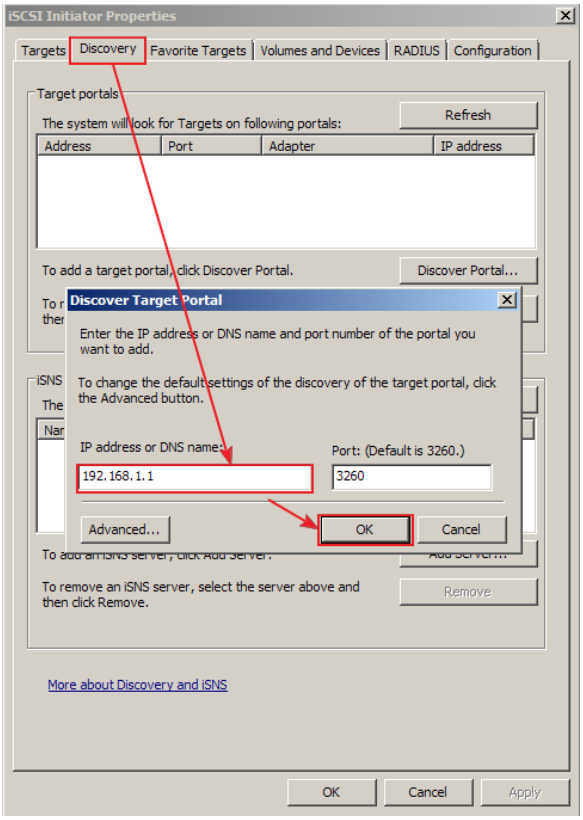
2. Add a Trunking Group on the gigabit switch.



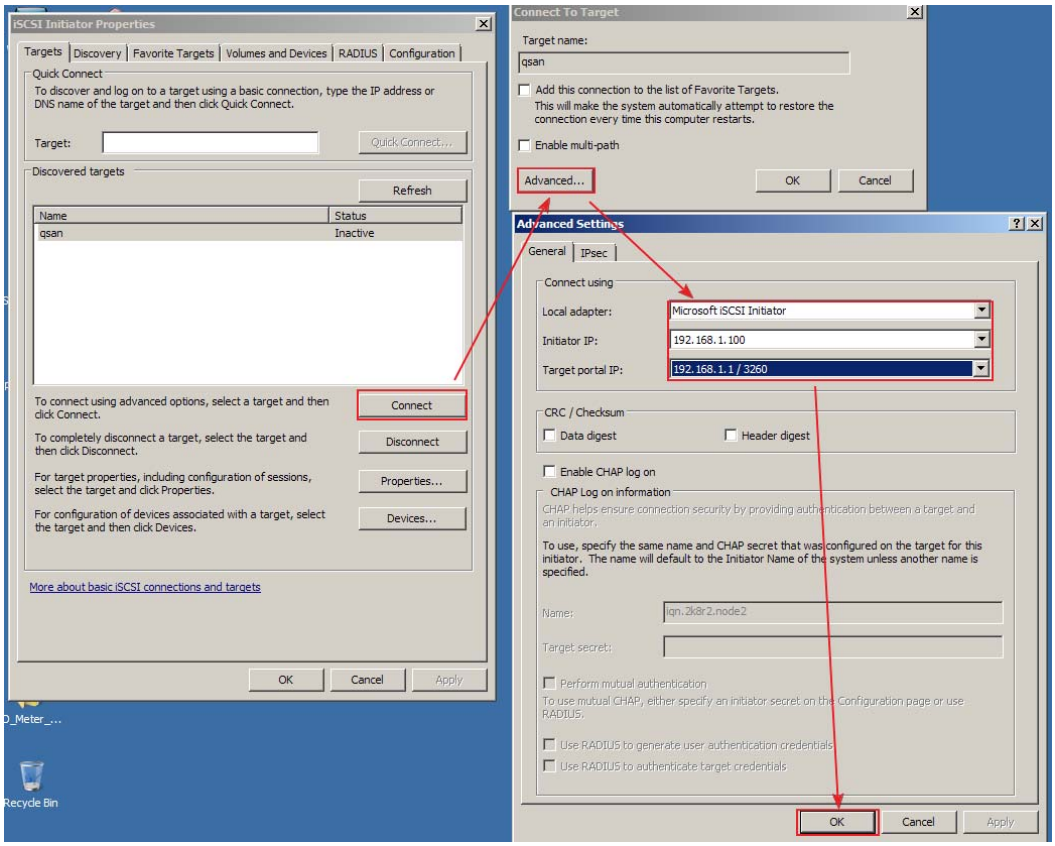
3. Select port 1, 2, 3, 4, and then press “Apply”. These four ports are aggregated to a group and are mapped to the iSCSI data port of **P210C**.



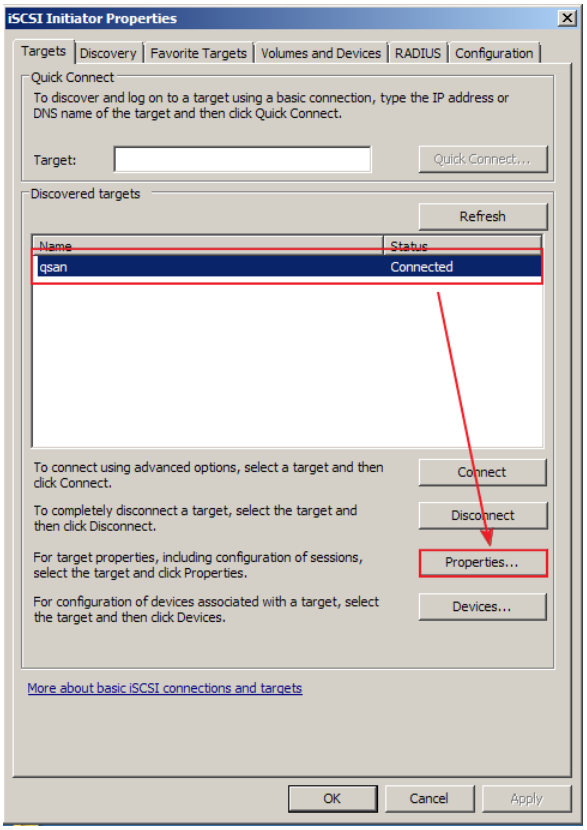
4. Add a “Discover” target portal in Microsoft iSCSI initiator.



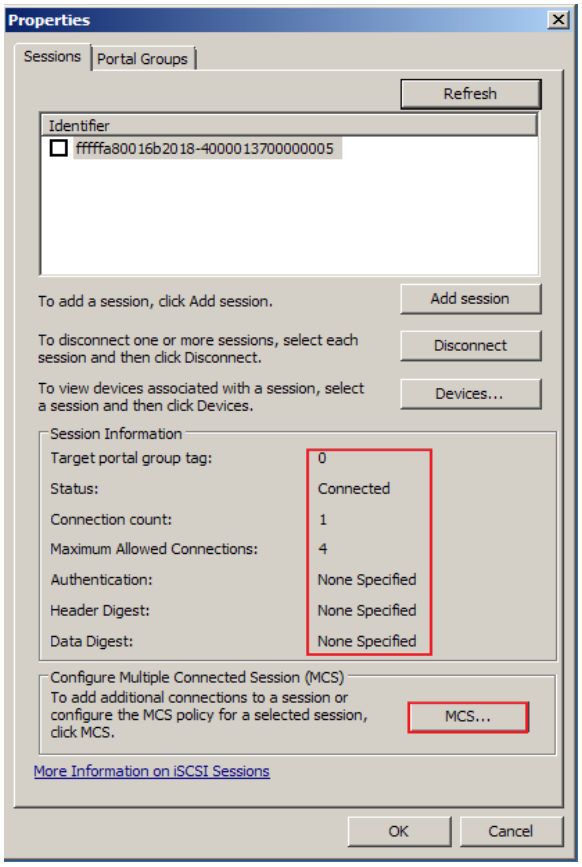
5. Here is the example of creating four connections within one session. It is necessary add each Initiator IP to target portal respectively.



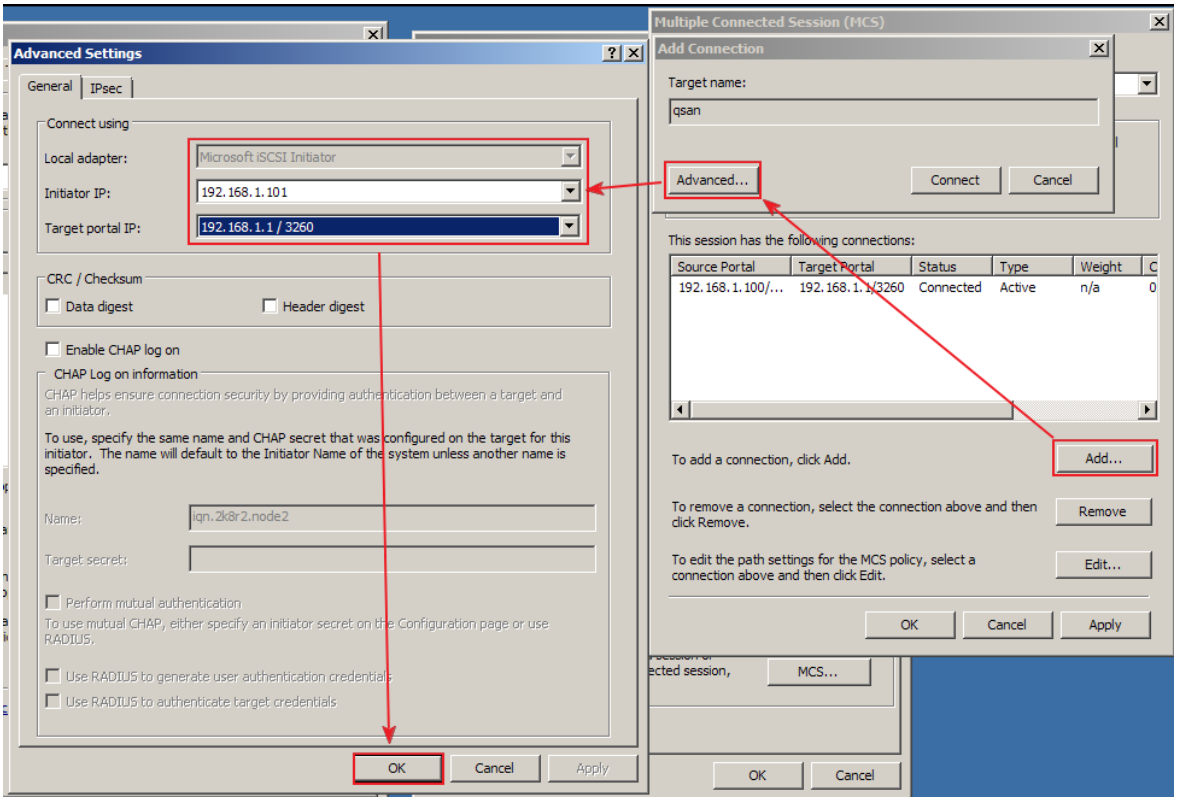
6. Create another connection. Click “Properties”.



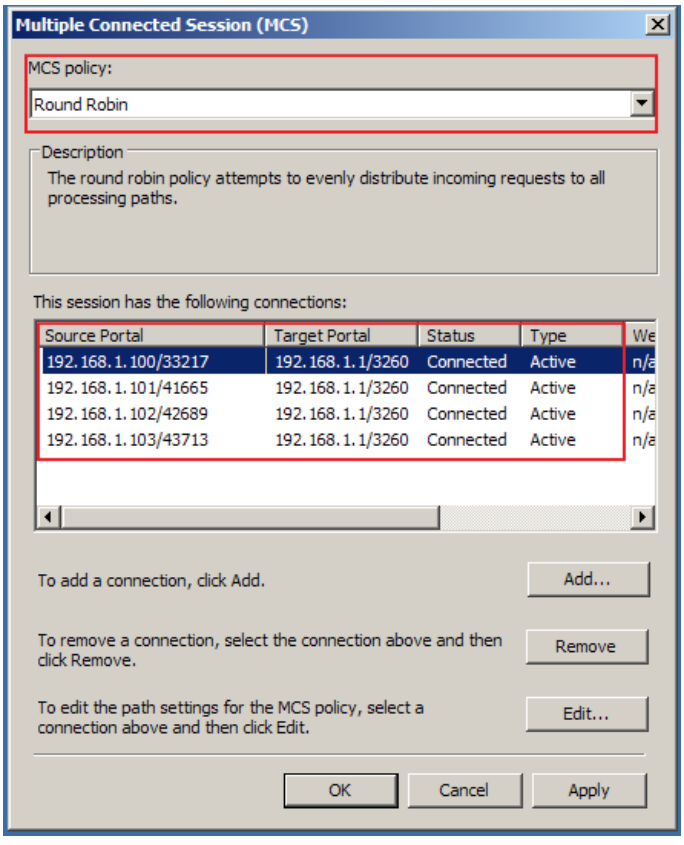
7. Select MCS. (There is only one connection currently.)



8. Add each connection one by one.

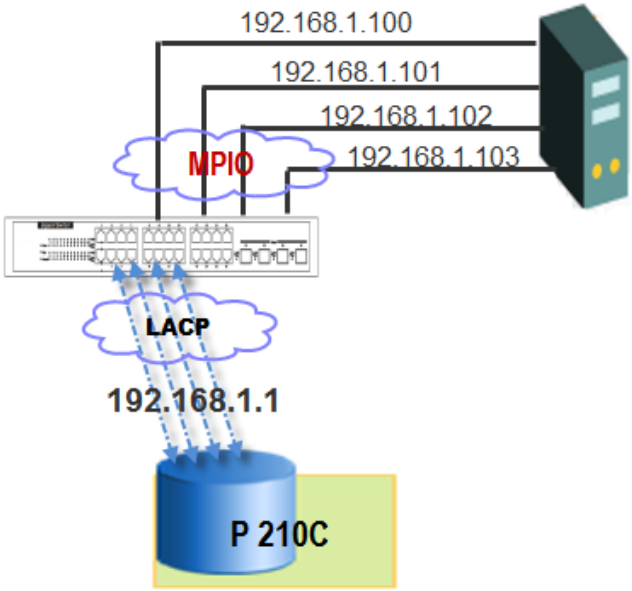


9. Finally, there are four connections in this session, and MCS policy is set to “Round Robin”.



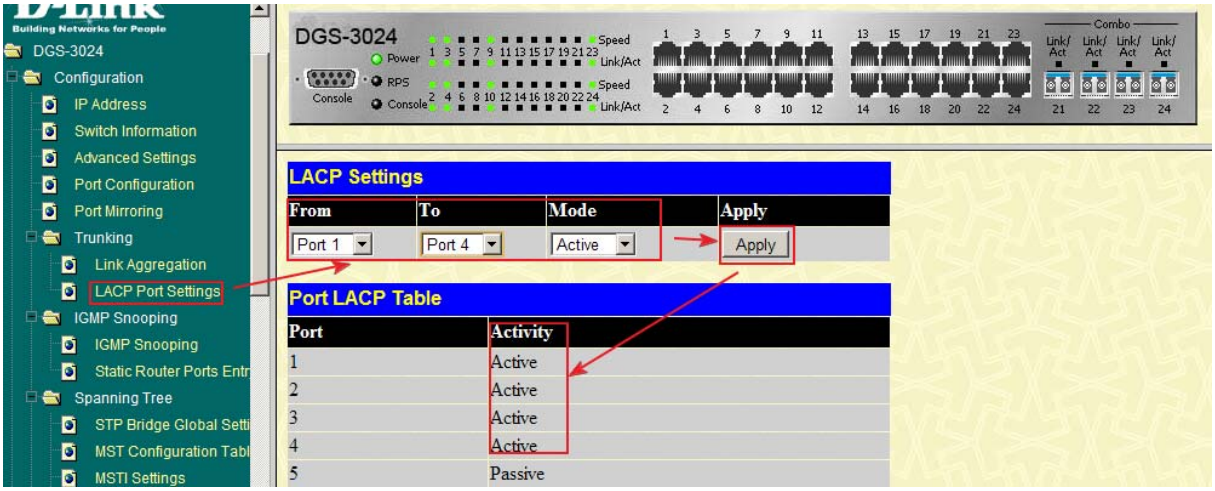
Part 2: Using MPIO and LACP in Red Hat Enterprise Linux 5

Diagram :



1. Set LACP on **P210C**.

2. Add a LACP Group on the gigabit switch. Set port 1, 2, 3, 4 as activite. These four ports are mapped to the iSCSI data port of **P210C**.



3. In RHEL 5, start iSCSI service firstly.

```
[root@staff-60 ~]# service iscsi start

Turning off network shutdown. Starting iSCSI daemon:       [  OK  ]
                                                           [  OK  ]
```

4. Add iSCSI service in startup demand.

```
[root@staff-60 ~]# chkconfig iscsi on
[root@staff-60 ~]#
```

5. Edit multipath.conf.

```
[root@staff-60 ~]# vi /etc/multipath.conf
```

6. Add “#” in blacklist, add a line “path_grouping_policy”, and modify the value of “rr_min_io” to 512.

```
# This is a basic configuration file with some examples, for device mapper
# multipath.
# For a complete list of the default configuration values, see
# /usr/share/doc/device-mapper-multipath-0.4.7/multipath.conf.defaults
# For a list of configuration options with descriptions, see
# /usr/share/doc/device-mapper-multipath-0.4.7/multipath.conf.annotated


# Blacklist all devices by default. Remove this to enable multipathing
# on the default devices.
blacklist {
#        devnode "*"
}

## By default, devices with vendor = "IBM" and product = "S/390.*" are
## blacklisted. To enable mulitpathing on these devies, uncomment the
## following lines.
#blacklist_exceptions {
#       device {
#               vendor  "IBM"
#               product "S/390.*"
#       }
#}

## Use user friendly names, instead of using WWIDs as names.
defaults {
        user_friendly_names yes
        path_grouping_policy    multibus
        getuid_callout          "/sbin/scsi_id -g -u -s /block/%n"
        prio_callout            /bin/true
        path_checker            readsector0
        rr_min_io               512
        rr_weight               priorities
        failback                immediate
        features                "1 queue_if_no_path"
```

Add #

7. Start multipathd sevice and add it in startup demand.

```
[root@staff-60 ~]# service multipathd start
Starting multipathd daemon:                                [  OK  ]
[root@staff-60 ~]# chkconfig multipathd on
```

8. Modify iscsi config file for increasing performance.

```
[root@staff-60 ~]# vi /etc/iscsi/iscsid.conf
```

9. Modify two values on the following.

```
##################################
# session and device queue depth
##################################

# To control how many commands the session will queue set
# node.session.cmds_max to an integer between 2 and 2048 that is also
# a power of 2. The default is 128.
node.session.cmds_max = 1024

# To control the device's queue depth set node.session.queue_depth
# to a value between 1 and 1024. The default is 32.
node.session.queue_depth = 128
```

10. Create four iSCSI interfaces for easy management.

```
[root@staff-60 ~]# iscsiadm -m iface -I ieth1 -o new
New interface ieth1 added
[root@staff-60 ~]# iscsiadm -m iface -I ieth2 -o new
New interface ieth2 added
[root@staff-60 ~]# iscsiadm -m iface -I ieth3 -o new
New interface ieth3 added
[root@staff-60 ~]# iscsiadm -m iface -I ieth4 -o new
New interface ieth4 added
[root@staff-60 ~]#
```

11. Map each iSCSI interface to physical Ethernet port.

```
[root@staff-60 ~]# iscsiadm -m  iface -I ieth1 -o update -n iface.net_ifacename -v eth1
ieth1 updated.
[root@staff-60 ~]# iscsiadm -m  iface -I ieth2 -o update -n iface.net_ifacename -v eth2
ieth2 updated.
[root@staff-60 ~]# iscsiadm -m  iface -I ieth3 -o update -n iface.net_ifacename -v eth3
ieth3 updated.
[root@staff-60 ~]# iscsiadm -m  iface -I ieth4 -o update -n iface.net_ifacename -v eth4
ieth4 updated.
```

12. Discovery target for each interface.

```
[root@staff-60 ~]# iscsiadm -m discovery -t sendtargets -p 192.168.1.1 -I ieth1 -I ieth2 -I ieth3 -I ieth4
192.168.1.1:3260,0 qsan
192.168.1.1:3260,0 qsan
192.168.1.1:3260,0 qsan
192.168.1.1:3260,0 qsan
```

13. Login to all nodes.

```
[root@staff-60 ~]# iscsiadm -m node -L all
Logging in to [iface: ieth3, target: qsan, portal: 192.168.1.1,3260]
Logging in to [iface: ieth1, target: qsan, portal: 192.168.1.1,3260]
Logging in to [iface: ieth4, target: qsan, portal: 192.168.1.1,3260]
Logging in to [iface: ieth2, target: qsan, portal: 192.168.1.1,3260]
Login to [iface: ieth3, target: qsan, portal: 192.168.1.1,3260]: successful
Login to [iface: ieth1, target: qsan, portal: 192.168.1.1,3260]: successful
Login to [iface: ieth4, target: qsan, portal: 192.168.1.1,3260]: successful
Login to [iface: ieth2, target: qsan, portal: 192.168.1.1,3260]: successful
```

14. Here are four sessions to the target.

```
[root@staff-60 ~]# iscsiadm -m session
tcp: [10] 192.168.1.1:3260,0 qsan
tcp: [11] 192.168.1.1:3260,0 qsan
tcp: [8] 192.168.1.1:3260,0 qsan
tcp: [9] 192.168.1.1:3260,0 qsan
```

15. Finally, create a multipath device dm-2 which includes four active paths.

```
[root@staff-60 ~]# multipath -ll
mpath4 (327b9001378a6d2cc) dm-2 Qsan,P210C
[size=100G][features=1 queue_if_no_path][hwhandler=0][rw]
\_ round-robin 0 [prio=0][active]
 \_ 14:0:0:0 sdd 8:48  [active][ready]
 \_ 12:0:0:0 sdb 8:16  [active][ready]
 \_ 13:0:0:0 sdc 8:32  [active][ready]
 \_ 15:0:0:0 sde 8:64  [active][ready]
```

Four active path

Summary

Both of MPIO and MC/S are the ways to achieve the multipath on server. So far, MC/S is only implemented in Microsoft iSCSI initiator. But MPIO can be used in Windows Server system, Linux, and even in Mac OS X with globalSAN iSCSI initiator.

MC/S can provide higher throughput than MPIO in Windows system, but it consumes more CPU resources than MPIO.

LACP and Trunking are the aggregation methods between **QSAN** controller and gigabit switch. Because LACP is a standard IEEE 802.3ad protocol, it is more flexible to use in a complex network infrastructure to achieve the load balance.

It is hard to say which multipath solution is the best. The following table summarizes the capabilities of the possible architectures. It will assist user to make the best decision.

Description	LACP	Trunking	MPIO	MC/S
Need switch support	Yes	Yes	No	No
Flexible on complex network environment	Yes	No	Yes	Yes
Support of Hardware initiators (HBA)	No	No	Yes	No
Multiple operating system support			Yes	No
Performance Compare			Good	Better

Applies to

- All **QSAN P series** controllers FW (20091113_1500)

References

- Title: Microsoft User's Guide for iSCSI initiator
  http://download.microsoft.com/download/A/E/9/AE91DEA1-66D9-417C-ADE4-92D824B871AF/uGuide.doc

- Title: Wikipedia -- Link aggregation
  http://en.wikipedia.org/wiki/Link_aggregation

- Implement iSCSI multipath in RHEL5
  ftp://ftp.qsan.com.tw/Qsan_Documents/White_Paper/QWP200801-P150C-Implement_iSCSI_multipath_in_RHEL5.pdf