VMware ESX Server

Ethernet-based Storage Configuration

Using Ethernet for a storage medium is not a new idea. Network file systems such as NFS and SMB/CIFS are entering their third decade of use. Network-based storage is popular because it allows data to be shared among many users using a medium and technology that already exists.

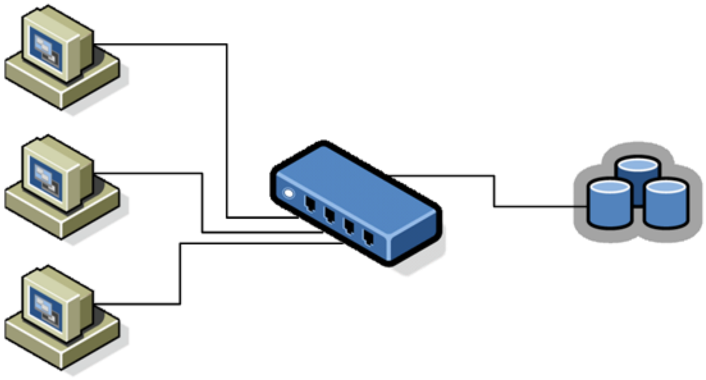
Newer generations of network-based storage attempt to share existing infrastructure, while combining ideas from the storage industry, such as fast, reliable transport of data. Gigabit Ethernet, for speed, and TCP, for reliability, represent this combination for the latest versions of NFS and iSCSI. iSCSI uses the sharing and virtualization capabilities of a traditional SAN, but also includes the low-cost infrastructure of a traditional Ethernet.

For reasons of security and performance, many companies set up isolated networks for network-based storage. Data passed on a shared network is more susceptible to inspection and modification. Storage data will be slowed by competing with other data on the network. For iSCSI configurations, VMware only supports isolated networks.

This technical note examines key issues in configuring iSCSI storage with virtual machines running on an ESX Server system. It covers the following topics:

Storage Networks

A typical storage network consists of a collection of computers connected to a collection of storage devices through a network of switches. It is common to have numerous computers accessing the same storage. The following figure shows several computer systems connected to network storage through an Ethernet switch. In this case, all of the systems communicate with the storage through a single link between the switch and the storage.



It's also common to have a single computer accessing numerous different storage devices, especially in the campus environment, with numerous NFS-mounted file systems.

In the case of systems running ESX Server, each computer can actually represent many virtual machines connected to the storage device. In the previous figure, the number of virtual machines would be a large multiple of the three physical systems, all sharing the storage.

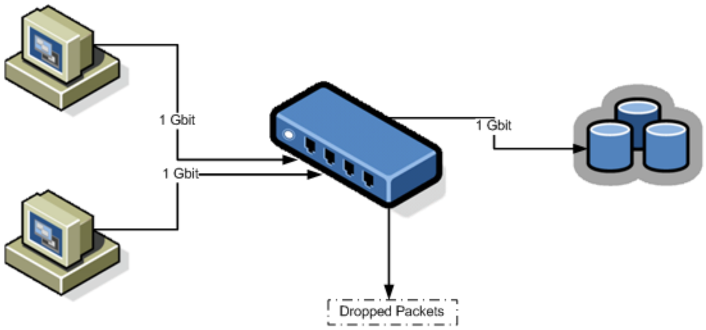
A Potential Configuration Pitfall

A common way of configuring a system is similar to the figure on the previous page. Many systems can access the storage in this case, and storage can potentially be shared across the various systems. A virtual machine can be migrated from one of these systems to another, and data stores are available to any of the systems.

However, the configuration means that each system is connected through a Gigabit Ethernet link to the switch, which is also connected to the storage through one Gigabit Ethernet link. In most configurations, with modern switches and typical traffic, this won't cause a problem.

When systems read data from storage, the best the storage can do is send enough data to fill the gigabit link between the storage and switch. This means it is unlikely that any single system or virtual machine will get gigabit, or wire speed read data transfers, but this is the expected situation when many systems share one storage device.

When writing data to storage, multiple systems or virtual machines might attempt to fill their links. As the following figure shows, when this happens, the switch between the systems and the storage has to drop data because it has a 1Gb connection to the storage device, but more than 1Gb of traffic to send to the storage device. An excess of data on the two connections from the systems on the left that can't be carried on the 1Gb connection to the storage is dropped by the switch.



In this case, network packets are dropped by the switch because it's limited to the amount of data it can transmit by the speed of the link between it and storage.

TCP is a reliable transmission protocol that ensures that dropped packets are retried and eventually reach their destination. Recovering from dropped network packets results in large performance degradation. In addition to time spent determining that data was dropped, the retransmission uses network bandwidth that could otherwise be used for current transactions.

TCP is designed to recover from occasional dropped packets and retransmits them quickly and seamlessly. When packets are discarded by the switch with any regularity, network throughput suffers significantly. The network becomes congested with requests to resend data and with the resent packets, and less data is actually transferred than in a network without congestion.

Most quality Ethernet switches have some capability of buffering, or storing, data and giving every device attempting to send data an equal chance to get to the destination. This ability to buffer some transmissions, combined with many systems limiting the number of outstanding commands, allows small bursts from several systems to be sent to a storage system in turn.

If the transactions are large, though, and there are multiple systems trying to send data through a single output port, they can exceed a switch's ability to buffer one request while another is transmitted. In this case, again, the switch drops the data it cannot send, and requests to retransmit and retransmitted data packets follow. For example, if an Ethernet switch can buffer 32K on an input port, but the system connected to it thinks it can send 256K to the storage device, some of the data is dropped.

Most managed switches provide information on dropped packets.

```
 *: interface is up
IHQ: pkts in input hold queue       IQD: pkts dropped from input queue
OHQ: pkts in output hold queue      OQD: pkts dropped from output queue
RXBS: rx rate (bits/sec)            RXPS: rx rate (pkts/sec)
TXBS: tx rate (bits/sec)            TXPS: tx rate (pkts/sec)
TRTL: throttle count

  Interface              IHQ   IQD  OHQ   OQD  RXBS RXPS  TXBS TXPS TRTL
-----------------------------------------------------------------------
* GigabitEthernet0/1       3  9922    0     0    476303000    62273
477840000  63677    0
```

In this example from a Cisco switch, the bandwidth used is 476303000 bits/second, which is less than half of gigabit wire speed. Even so, the port is buffering incoming packets and has dropped quite a few. The final line of this interface summary indicates this port has already dropped almost 10,000 inbound packets in the IQD column.

If a network regularly gets into this situation and packets are frequently dropped, performance will be less than optimal.

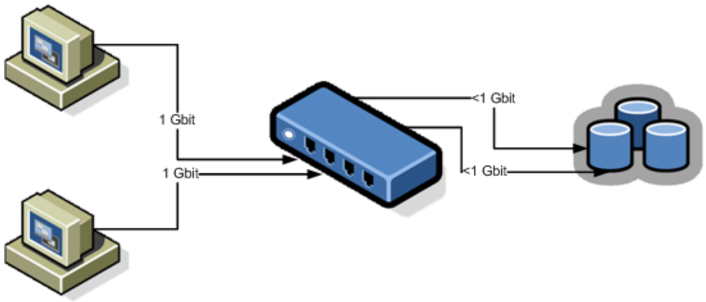
Performance Tuning

Configuration changes to avoid this problem involve making sure several input Ethernet links are not funneled into one output link, resulting in an oversubscribed link. Any time a number of links transmitting near capacity are switched to a smaller number of links, oversubscription is a possibility.

There usually is no problem switching many links into one. A few dropped packets are expected and do not result in serious performance degradation. The problem is a result of switching many heavily used links into one. In this situation, the oversubscribed links can't handle the traffic and packets are dropped. As mentioned in the previous example, two 1Gb links cannot be forced onto a single gigabit link, and the result is dropped packets. Whenever the number of input links exceeds the number of output links, oversubscription and performance degradation can occur.

The problem exists even when links are oversubscribed momentarily, as when two systems attempt to write data to the same storage port, and the write request is larger than the switch's capability to buffer requests. In such situations, Ethernet packets are dropped. Performance degrades in this situation.

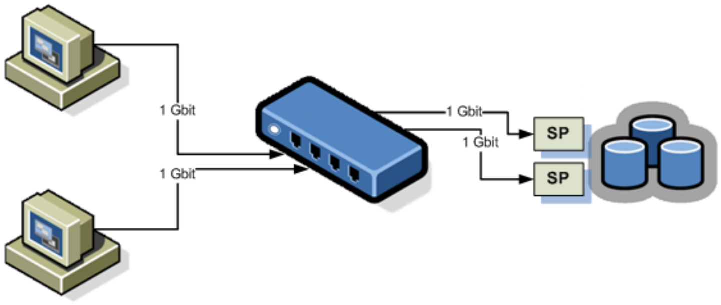
As a rule of thumb, applications or systems that write a lot of data to storage, such as data acquisition or transaction logging systems, should not share Ethernet links to a storage device. These types of applications perform best with multiple connections to storage devices. In the following figure, there are multiple connections from the switch to the storage, allowing more than 1Gb of data to be sent to the storage.



Using VLANs or VPNs does not provide a suitable solution to the problem of link oversubscription in shared configurations. VLANs and other virtual partitioning of a network provide a way of logically designing a network, but don't change the physical capabilities of links and trunks between switches.

When storage traffic and other network traffic end up sharing physical connections, as they would with a VPN, the possibility for oversubscription and lost packets exists. The same is true of VLANs that share interswitch trunks. Performance design for a storage network must take into account the physical constraints of the network, not logical allocations.

Even under heavy loads in this configuration, the storage processor might limit the total throughput. If you are working with systems in which the write load is heavy, you may need to assign separate storage processors to separate systems to handle the amount of traffic bound for the storage. As the following figure shows, splitting the load across storage processors might improve performance.



Difference from Fibre Channel

Fibre Channel SANs do not suffer from the congestion and oversubscription problems as readily because of the Fibre Channel protocol. Fibre Channel includes a mechanism for port-to-port throttling, not just end-to-end throttling. This means a Fibre Channel switch can limit the amount of data a connected system sends and avoids situations in which information must be dropped. Therefore, Fibre Channel does not degrade into a retransmission recovery scheme that saps the storage network's performance, and is more capable of operating near wire speed.

Adapters That Do Not Support Fast Retransmission

Some iSCSI adapters that do not support fast retransmission can exhibit slow performance in congested networks or when retransmissions are common. The symptom of this is I/O stalling, or stopping completely, for a few seconds at a time when these conditions exist. In extreme cases, this results in instances of ESX Server or the guest operating system attempting to abandon outstanding commands by issuing SCSI aborts. These aborts are logged in the VMkernel log. The aborted commands do not represent lost data or any compromise of data integrity.

In the worst case, these adapters might wait more than half a second between retransmission requests, effectively freezing I/O until all requests are fulfilled. In some VMware observations, the number of retransmission requests was more than 10, meaning no I/O happened for five seconds. The following trace shows TCP activity between an adapter and a storage array, with the adapter waiting longer than half a second between each request. This is an extreme case that happens only on a congested network with frequently dropped packets. The Time column represents elapsed time in seconds in the trace. The Info column shows ACKs corresponding to each previous retransmission request. Some output has been removed from the Info column for readability.

```
No.     Time      Source          Destination   Protocol Info
  52831 4.158316  <ESX adapter>   <storage>     TCP      29812 > 3260 [ACK]
  52832 4.158339  <ESX adapter>   <storage>     TCP      29812 > 3260 [ACK]
  52833 4.549148  <ESX adapter>   <storage>     TCP      [TCP Retransmission]
          29812 > 3260 [ACK]
  52834 4.549404  <storage>       <ESX adapter> TCP      3260 > 29812 [ACK]
  52835 5.149050  <ESX adapter>   <storage>     TCP      [TCP Retransmission]
          29812 > 3260 [ACK]
  52836 5.149252  <storage>       <ESX adapter> TCP      3260 > 29812 [ACK]
  52837 5.749079  <ESX adapter>   <storage>     TCP      [TCP Retransmission]
          29812 > 3260 [ACK]
  52838 5.749083  <storage>       <ESX adapter> TCP      3260 > 29812 [ACK]
  52839 6.348860  <ESX adapter>   <storage>     TCP      [TCP Retransmission]
          29812 > 3260 [ACK]
  52840 6.348972  <storage>       <ESX adapter> TCP      3260 > 29812 [ACK]
  52841 6.948882  <ESX adapter    <storage>     TCP      [TCP Retransmission]
          29812 > 3260 [ACK]
  52842 6.948888  <storage>       <ESX adapter> TCP      3260 > 29812 [ACK]
  52843 7.548775  <ESX adapter>   <storage>     TCP      [TCP Retransmission]
          29812 > 3260 [ACK]
  52844 7.548780  <storage>       <ESX adapter> TCP      3260 > 29812 [ACK]
  52845 8.148679   <ESX adapter>  <storage>     TCP      [TCP Retransmission]
          29812 > 3260 [ACK]
  52846 8.148683  <storage>       <ESX adapter> TCP      3260 > 29812 [ACK]
```

In this example, nearly four seconds elapse while requesting data retransmissions. Because the adapter is stalled (waiting for retransmitted data), no other I/O is going on in this case.

If stalls described here are seen when using iSCSI with ESX Server, follow suggestions described in the Performance Tuning section earlier in this document. Ensure there are not multiple systems performing frequent, large-block write operations to a single storage port. In environments where this type of activity is likely to happen, dedicate additional links between the switch and the storage or additional storage processors to those systems. Note that the some adapters demonstrate poor characteristics that make the problem worse, but frequently dropped packets and network congestion reduce the performance of any network.

VMware, Inc. 3145 Porter Drive Palo Alto, CA 94304 www.vmware.com