

Microsoft Network Load Balancing and Cisco Catalyst Configuration

OVERVIEW	2
UNICAST MODE.....	2
MULTICAST MODE.....	3
ANALYSIS.....	4
CPU UTILIZATION	4
CAPTURE PACKETS	5
MICROSOFT READING.....	6
MULTICAST NOTES.....	6
<i>RFC1112 Host Extensions for IP Multicasting: August 1989.....</i>	<i>6</i>
<i>RFC 2236 Internet Group Management Protocol, Version 2: November 1997.....</i>	<i>9</i>
<i>RFC 4541 Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches: May 2006</i>	<i>9</i>

OVERVIEW

Microsoft's Network Load Balancing software allows a collection of Windows boxes to share a virtual IP address and to distribute the work of handling incoming and outgoing traffic amongst the members of the cluster. Deploying this product has implications for nearby switches and routers; in this document, I outline the mechanics of configuring Catalyst gear to support this product.

BTW: Cisco now has an excellent document describing the issues; see http://www.cisco.com/en/US/products/hw/switches/ps708/products_configuration_example09186a0080a07203.shtml --sk 2011-05-27

UNICAST MODE

In Unicast Mode, the sys admin clicks the 'unicast' button in the MS NLB configuration GUI. This choice instructs the cluster members to respond to ARP queries for their virtual address using MAC address *abc* ... but to source the packets they emit using MAC address *xyz*. The local Ethernet switch will populate its port-to-MAC-address table (called a 'CAM' table or a 'mac-address-table' in Cisco-speak) using MAC address *xyz* ... and will never know which port connects to MAC address *abc*.

Thus, when local hosts, including the local router, want to send a packet to the virtual IP address, they will send that packet using MAC address *abc*. The switch won't know which port holds MAC address *abc* and will therefore flood the packet to all ports.

When this happens, the cluster gets what it wants: every member receives the incoming packet. The cluster members use some internal algorithm to determine which one of them will process the packet and, possibly, respond to it.

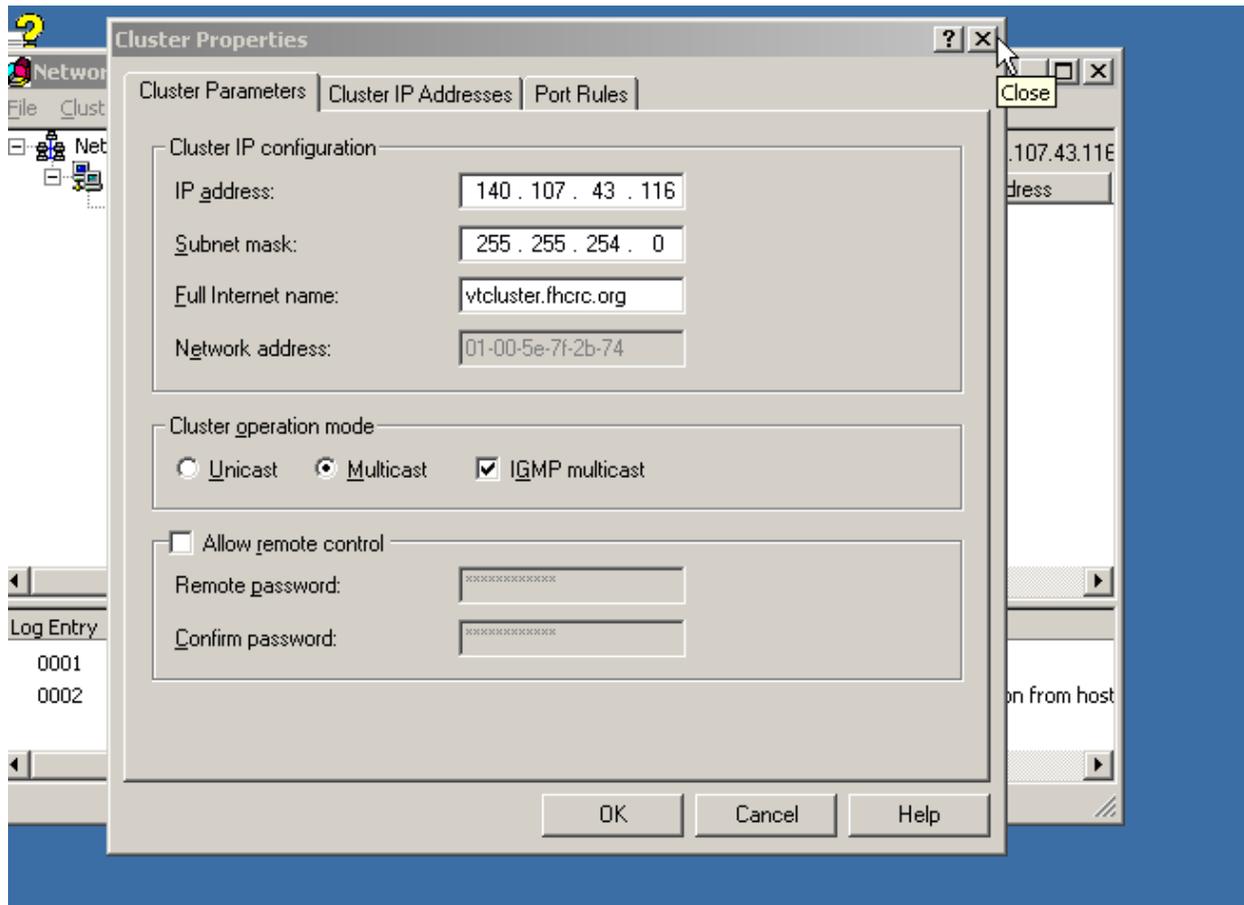
While the cluster is getting what it wants, every other station on this subnet (broadcast domain and VLAN) is also receiving this packet. I call this behavior *anti-social*, a term which I define specifically as meaning that the node is optimizing for its own benefit at the expense of its neighbors. As long as in-bound cluster traffic remains low, I have trouble imagining a scenario under which this behavior would disrupt service to the neighbors.

Nevertheless, I don't like taking these kinds of risks – designs which add a subnet-wide failure mode seem undesirable to me.

One way to restrict NLB's anti-social behavior is to assign the cluster members to their own VLAN, or even to a private VLAN. Another way is to insert static entries into the switch's mac-address-table. I leave these configurations as an exercise to the reader.

MULTICAST MODE

In Multicast Mode (aka IGMP Mode), the sys admin clicks the 'IGMP Multicast' button in the MS NLB configuration GUI. This choice instructs the cluster members to respond to ARPs for their virtual address using a multicast MAC address, say, *pdq*, (01:00:5e:7f:2b:74 in the example below) and to emit IGMP Membership Report packets.



If the local switch also speaks IGMP, it will then populate mac-address-table appropriately, associating the multicast MAC address with each port feeding a cluster member. In this way, when a local station ARPs for the cluster's virtual IP address, the cluster will respond with *pdq*, the local station will address the packet to *pdq*, the local switch will forward the packet out each of the ports feeding cluster members, and everybody is happy: the cluster members get what they want (every single packet destined for the virtual IP address) and the neighbors don't have to hear about it. In our Catalyst-based environment, this approach works seamlessly: our Catalysts shipped with IGMP enabled, no need for configuration changes there.

However, cluster's virtual IP address becomes unreachable from outside the local subnet.

Poking around, I found that the local router (a Layer 3 switch: Catalyst 6500 w/Sup 720) was ARPing for the cluster's virtual IP address, the cluster members were replying ... but the Cat6500 didn't seem to hear. This ARP entry never appeared in its ARP table; it eventually quit

ARPing and returned an ICMP Host unreachable to my test station (located outside the cluster's subnet).

Here's the story as I understand it. Microsoft thinks that associating a unicast IP address with a multicast MAC address is hunky-dory. Cisco thinks that doing this violates an RFC and therefore ignores ARP responses containing such a mapping. I poked briefly at some of the multicast RFCs and wasn't able to resolve the issue. If you can find a reference, drop me a note.

If you want override Cisco's default behavior, here's what you do:

```
arp 10.1.2.15 0100.5301.0200 ARPA
```

This static ARP entry populates the router's ARP table, the router doesn't bother to ARP but just constructs and forwards the packet. Connectivity restored.

However, you aren't out of the woods yet. Via IGMP, the Cat6500 is hearing from the cluster members about their use of multicast address pdq. In theory, this would allow the Cat6500 to populate its mac-address-table. And it does. But, curiously enough, the Cat6500 ignores this entry and process-switches each cluster-bound packet. I'm unclear why, but perhaps again it considers a unicast IP address to multicast MAC address mapping to be invalid. This lead to high CPU utilization on our Cat6500 (spikes to 100% during the day). So, I inserted a static mac-address-table entry, and the Cat6500 started switching cluster-bound packets in hardware once more.

```
mac-address-table static 0100.5301.0200 vlan 2 interface GigabitEthernet3/1 disable-snooping
```

The 'disable-snooping' parameter is essential; without it, the statement does not affect behavior.

In this example:

Cluster virtual IP address:	10.1.2.15
Cluster multicast MAC address:	0100.5301.0200
VLAN on which cluster is located:	2
Interface servicing VLAN 2:	GigabitEthernet3/1

```
arp 10.1.2.15 0100.5301.0200 ARPA
mac-address-table static 0100.5301.0200 vlan 2 interface GigabitEthernet3/1 disable-snooping
```

ANALYSIS

CPU Utilization

To identify major contributors to CPU utilization:

```
Router#show processes cpu | exclude 0.00
CPU utilization for five seconds: 91%/50%; one minute: 89%; five minutes: 47%
  PID Runtime(ms)   Invoked    uSecs   5Sec   1Min   5Min  TTY Process
   5     881160     79142    11133   0.49%  0.19%  0.16%  0 Check heaps
  98    121064    3020704      40  40.53% 38.67% 20.59%  0 IP Input
```

```
245      209336      894828      233  0.08%  0.05%  0.02%  0 IFCOM Msg Hdlr
```

[Cribbed from

http://www.cisco.com/en/US/customer/products/hw/switches/ps708/products_tech_note09186a00804916e0.shtml, a link which also describes, in greater detail that I display below, how to sniff on Route Processor traffic.]

The 'IP Input' line indicates that the Route Processor is spending much of its time process-switching transit packets.

Capture Packets

To watch your C6K process-switch packets, attach a sniffer to a 'shutdown' interface and configure a SPAN port to forward Route Processor traffic to that shutdown interface. In the following example, Gi3/15 is an administratively 'shutdown' interface, and the sniffer is plugged into Gi3/16.

```
Router# config t
Router(config)# monitor session 1 source interface Gi3/15
Router(config)# monitor session 1 destination interface Gi3/16
Router(config)# exit
Router# remote login switch
Trying Switch ...
Entering CONSOLE for Switch
Type "^C^C^C" to end this session
```

```
Router-sp#test monitor add 1 rp-inband both
Router-sp#test monitor show session 1
Session [1] Info
```

```
-----
asic_session    = 0
session_type    = 0
session_status  = 0
dst_ltl_idx     = 0x10000
decap_index     = -1
```

```
Source Port-VLAN Info
```

```
-----
Ingress Source Ports:  3/15 15/1
Egress Source Ports  :  3/15 15/1
Ingress Source Vlan:  <null>
Egress Source Vlan  :  <null>
Ingress Filter Vlan :  <null>
Egress Filter Vlan  :  <null>
Exclude Filter Vlan :  <empty>
Exclude Alt Filter Vlan : <empty>
Ingress Filter Vlan Count: 0
Egress Filter Vlan Count : 0
Exclude Filter Vlan Count: 0
Exclude Alt Vlan Count   : 0
```

Destination ports: 3/16

Router-sp#

Microsoft Reading

Here is a list of Microsoft documents describing this product.

Network Load Balancing Technical Overview

<http://www.microsoft.com/technet/prodtechnol/windows2000serv/deploy/confeat/nlbovw.mspx>

Network Load Balancing FAQ

<http://technet2.microsoft.com/WindowsServer/en/library/884c727d-6083-4265-ac1d-b5e66b68281a1033.mspx?mfr=true>

Configuration options for WLBS hosts connected to layer 2 switches

<http://support.microsoft.com/kb/193602>

IGMP Multicast support in Microsoft's Network Load Balancing product under Windows 2003

<http://technet2.microsoft.com/WindowsServer/en/library/bf3a1c95-f960-4ed3-b154-3586631fb0061033.mspx?mfr=true>

Network Load Balancing parameters (how to do it, from the Windows point of view)

<http://technet2.microsoft.com/WindowsServer/en/library/57c24429-0268-4ed8-afdf-fd4b0b6539b71033.mspx?mfr=true>

Network Load Balancing: Configuration Best Practices for Windows 2000 and Windows 2003

<http://technet2.microsoft.com/windowsserver/en/library/c7da3162-2055-438d-87c1-c1086c694c9f1033.mspx?mfr=true>

Multicast Notes

RFC1112 HOST EXTENSIONS FOR IP MULTICASTING: AUGUST 1989

Useful for understanding how IP multicasting impacts hosts.

[...]

4. HOST GROUP ADDRESSES

Host groups are identified by class D IP addresses, i.e., those with "1110" as their high-order four bits. Class E IP addresses, i.e., those with "1111" as their high-order four bits, are reserved for future addressing modes.

In Internet standard "dotted decimal" notation, host group addresses range from 224.0.0.0 to 239.255.255.255. The address 224.0.0.0 is guaranteed not to be assigned to any group, and 224.0.0.1 is assigned to the permanent group of all IP hosts (including gateways). This is used to address all multicast hosts on the directly connected network. There is no multicast address (or any other IP address) for all hosts on the total Internet. The addresses of other well-known, permanent groups are to be published in "Assigned Numbers".

[...]

6.4. Extensions to an Ethernet Local Network Module

The Ethernet directly supports the sending of local multicast packets by allowing multicast addresses in the destination field of Ethernet packets. All that is needed to support the sending of multicast IP datagrams is a procedure for mapping IP host group addresses to Ethernet multicast addresses.

An IP host group address is mapped to an Ethernet multicast address by placing the low-order 23-bits of the IP address into the low-order 23 bits of the Ethernet multicast address 01-00-5E-00-00-00 (hex). Because there are 28 significant bits in an IP host group address, more than one host group address may map to the same Ethernet multicast address.

[...]

7.4. Extensions to an Ethernet Local Network Module

To support the reception of multicast IP datagrams, an Ethernet module must be able to receive packets addressed to the Ethernet multicast addresses that correspond to the host's IP host group addresses. It is highly desirable to take advantage of any address filtering capabilities that the Ethernet hardware interface may have, so that the host receives only those packets that are destined to it.

Unfortunately, many current Ethernet interfaces have a small limit on the number of addresses that the hardware can be configured to recognize. Nevertheless, an implementation must be capable of listening on an arbitrary number of Ethernet multicast addresses, which may mean "opening up" the address filter to accept all multicast packets during those periods when the number of addresses exceeds the limit of the filter.

For interfaces with inadequate hardware address filtering, it may be desirable (for performance reasons) to perform Ethernet address filtering within the software of the Ethernet module. This is not mandatory, however, because the IP module performs its own filtering based on IP destination addresses.

[...]

APPENDIX I. INTERNET GROUP MANAGEMENT PROTOCOL (IGMP)

The Internet Group Management Protocol (IGMP) is used by IP hosts to report their host group memberships to any immediately-neighbor-ing multicast routers. IGMP is an asymmetric protocol and is specified here from the point of view of a host, rather than a multicast router. (IGMP may also be used, symmetrically or asymmetrically, between multicast routers. Such use is not specified here.)

Like ICMP, IGMP is an integral part of IP. It is required to be implemented by all hosts conforming to level 2 of the IP multicasting specification. IGMP messages are encapsulated in IP datagrams, with an IP protocol number of 2.

[...]

Informal Protocol Description

Multicast routers send Host Membership Query messages (hereinafter called Queries) to discover which host groups have members on their attached local networks. Queries are addressed to the all-hosts group (address 224.0.0.1), and carry an IP time-to-live of 1.

Hosts respond to a Query by generating Host Membership Reports (hereinafter called Reports), reporting each host group to which they belong on the network interface from which the Query was received. In order to avoid an "implosion" of concurrent Reports and to reduce the total number of Reports transmitted, two techniques are used:

1. When a host receives a Query, rather than sending Reports immediately, it starts a report delay timer for each of its group memberships on the network interface of the incoming Query. Each timer is set to a different, randomly-chosen value between zero and D seconds. When a timer expires, a Report is generated for the corresponding host group. Thus, Reports are spread out over a D second interval instead of all occurring at once.
2. A Report is sent with an IP destination address equal to the host group address being reported, and with an IP time-to-live of 1, so that other members of the same group on the same network can overhear the Report. If a host hears a Report for a group to which it belongs on that network, the host stops its own timer for that group and does not generate a Report for that group. Thus, in the normal case, only one Report will be generated for each group present on the network, by the member host whose delay timer expires first. Note that the multicast routers receive all IP multicast datagrams, and therefore need not be addressed explicitly. Further note that the routers need not know which hosts belong to a group, only that at least one host belongs to a group on a particular network.

There are two exceptions to the behavior described above. First, if a report delay timer is already running for a group membership when a Query is received, that timer is not reset to a new random value, but rather allowed to continue running with its current value. Second, a report delay timer is never set for a host's membership in the all-hosts group (224.0.0.1), and that membership is never reported.

[...]

Multicast routers send Queries periodically to refresh their knowledge of memberships present on a particular network. If no Reports are received for a particular group after some number of Queries, the routers assume that that group has no local members and that they need not forward remotely-originated multicasts for that group onto the local network. Queries are normally sent infrequently (no more than once a minute) so as to keep the IGMP overhead on hosts

and networks very low. However, when a multicast router starts up, it may issue several closely-spaced Queries in order to build up its knowledge of local memberships quickly.

When a host joins a new group, it should immediately transmit a Report for that group, rather than waiting for a Query, in case it is the first member of that group on the network. To cover the possibility of the initial Report being lost or damaged, it is recommended that it be repeated once or twice after short delays. (A simple way to accomplish this is to act as if a Query had been received for that group only, setting the group's random report delay timer. The state transition diagram below illustrates this approach.)

Note that, on a network with no multicast routers present, the only IGMP traffic is the one or more Reports sent whenever a host joins a new group.

[...]

APPENDIX II. HOST GROUP ADDRESS ISSUES

This appendix is not part of the IP multicasting specification, but provides background discussion of several issues related to IP host group addresses.

Group Address Binding

The binding of IP host group addresses to physical hosts may be considered a generalization of the binding of IP unicast addresses. An IP unicast address is statically bound to a single local network interface on a single IP network. An IP host group address is dynamically bound to a set of local network interfaces on a set of IP networks.

It is important to understand that an IP host group address is NOT bound to a set of IP unicast addresses. The multicast routers do not need to maintain a list of individual members of each host group. For example, a multicast router attached to an Ethernet need associate only a single Ethernet multicast address with each host group having local members, rather than a list of the members' individual IP or Ethernet addresses.

RFC 2236 INTERNET GROUP MANAGEMENT PROTOCOL, VERSION 2: NOVEMBER 1997

Useful for understanding how IGMP works.

RFC 4541 CONSIDERATIONS FOR INTERNET GROUP MANAGEMENT PROTOCOL (IGMP) AND MULTICAST LISTENER DISCOVERY (MLD) SNOOPING SWITCHES: MAY 2006

Useful for trouble-shooting issues between multi-vendor IGMP implementations.

