

Name resolution and network selection for PBSPro network messages on dual-network machines

0. Introduction

PBSPro makes certain assumptions about name resolution that do not always hold in modern clusters when there are machines with more than one interface (and certainly in clusters with nodes on different networks).

In PBSPro 13.1 variables were introduced to cope with this; this document will explore how to use them in typical situations (since the Admin Guide tends to merely document the presence of the variables) together with OS facilities to influence how TCP/IP packets are routed.

We do make assumptions in this document about the sanity of the name resolution framework:

- every name resolves to **only one** IP address (so there is no ambiguity), i.e. no name appears on two lines in `/etc/hosts`;
- All different resolvers (hosts files, NIS, DNS, LDAP) agree (unless all relevant IP addresses are in `/etc/hosts` on all the hosts);
- **Name resolution should be consistent across the entire cluster, i.e. a given IP address resolves to the same canonical name across all hosts that run PBSPro daemons¹**;
- The canonical name for each address (first on the line in `/etc/hosts`) is *usually fully qualified* (with domain name); the output of “hostname” is *usually* the short name.

External machines that are just used to submit jobs or run client commands (e.g. `qstat`, `pbs_rsub`) can resolve the same names to different IP addresses.

TPP and pbs_comm

In PBSPro 13.1, when the default TPP transport for interdaemon messages is enabled, daemons communicate with each other through a `pbs_comm` daemon; that daemon does basic buffering but can also route between networks (daemons can also use one of a set of `pbs_comm` daemons for better fault tolerance, specified using `PBS_LEAF_ROUTERS`; `pbs_comm` network graphs can become quite complex).

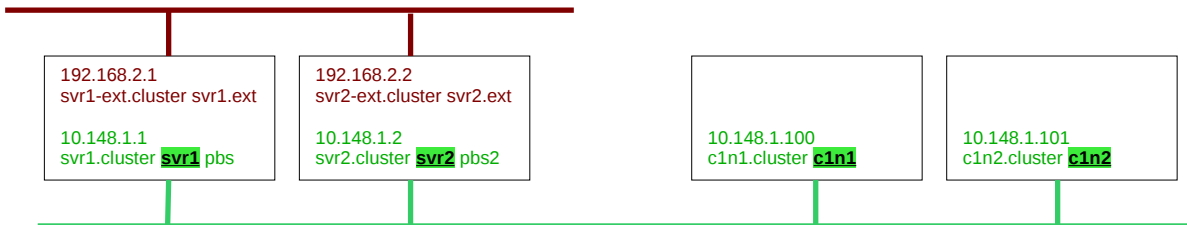
By default, if `PBS_LEAF_ROUTERS` is unspecified, either `PBS_SERVER` or `PBS_PRIMARY` plus `PBS_SECONDARY` are the addresses at which server, scheduler and MoM daemons expect to find a `pbs_comm` daemon (and they will register themselves to all of these `pbs_comm` daemons).

A host will query all leaf routers it knows to find out if a target address has been registered to it and will use the first `pbs_comm` that has seen the target. All connections are set up by the server, scheduler and MoM daemons, not `pbs_comm`.

¹MoMs will often interpret names in `exec_host`, `exec_host2` and `exec_vnode` attributes that were filled in by the server according to their local name resolution framework, so discrepancies can often cause strange failures (sometimes only for multinode jobs).

If the connection to the target is lost, then a daemon will renegotiate anew connection through all leaf routers again (which means the network is resilient against failures of a single pbs_comm if there is a pbs_comm pair).

1. Nominal Situation



Here all hostnames (marked in colour) resolve to an IP address that is on the network that PBSPro is meant to use.

The `/etc/pbs.conf` is the same on all nodes:

```
PBS_SERVER=svr1
PBS_PRIMARY=svr1.cluster
PBS_SECONDARY=svr2.cluster
```

Nodes are created using

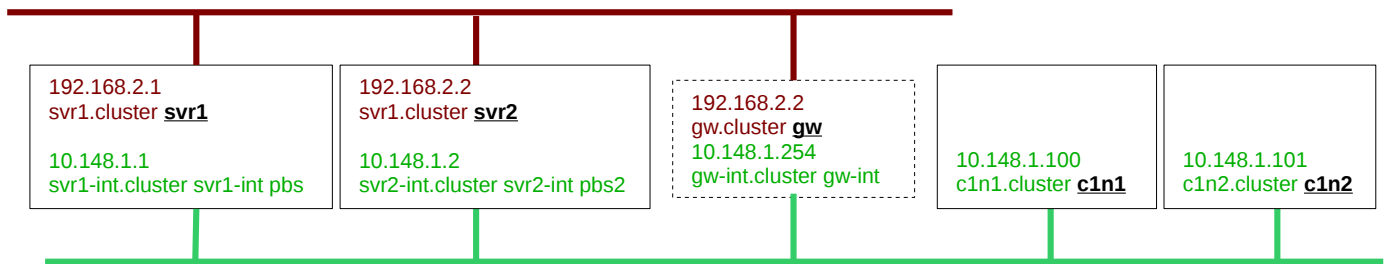
```
qmgr -c "create node c1n1"
```

Note that ***the name used to create a node must be the hostname of the node (output of "hostname")***. If not, hooks and MoM v2 config files may not work as expected, since MoM will silently assume the natural vnode name to be its hostname when merging information from hooks or v2 config files.

This configuration file will result in the following:

- pbs_comm on svr1 and svr2 will listen on all network interfaces
- svr1 and svr2 will register to both pbs_comm daemons using their 10.148.1.x address
- execution nodes will register to both pbs_comm daemons using their 10.148.1.10x address
- When trying to contact a MoM at server startup, the server will send a "hello" to the 10.148.1.10x addresses of the execution hosts, which will match the node's address registered to pbs_comm
- When MoM boots it will contact PBS_SERVER (i.e. 10.148.1.1). If the active server is the secondary, the node will remain down until the secondary tries to recover it in its "ping down nodes" routines (in less than 5 minutes).
- PBS commands on the execution host will be directed to svr1 (PBS_SERVER) and then to svr2.cluster (PBS_SECONDARY) if it does not respond.

2. Server Hostname on Different Interface



Here the same default `/etc/pbs.conf` will lead to very different results:

- pbs_comm daemons on svr1 and svr2 will still listen on all network interfaces
- svr1 and svr2 will register to pbs_comm using their external network address 192.168.2.x
- Execution nodes will typically resolve PBS_SERVER and PBS_SECONDARY to either:
 - the 10.148.1.x addresses (if e.g. they are not on the external network), in which case pbs_comm will deny having “seen this host”; it will thus simply not work
 - the 192.168.2.x addresses, which are unreachable without passing through a gateway, which becomes a single point of failure (or worse — if it is a NAT gateway it may drop connections if its connection tracking table is full).

Please note that NAT gateways are not supported by Altair due to the complex tuning requirements to avoid dropping packets and connections and because PBSPro client commands on execution nodes will not work if they have to cross a NAT gateway unless Munge authentication is used.

The dependency on the gateway can be removed in more than one way; this is **always recommended by Altair**. Implement any of these three solutions:

- host routes can be installed on the execution nodes so that they send packets destined for svr1 to svr1-int, and for svr2 to svr2-int.
- `/etc/pbs.conf` on execution nodes can use the aliases “pbs” and “pbs2” in PBS_SERVER, PBS_PRIMARY and PBS_SECONDARY.
In that case, **only on the servers**:
 - PBS_SERVER_HOST_NAME needs to be set to the hostname of the machine (since it no longer matches PBS_SERVER);
 - PBS_LEAF_NAME needs to be set to the “pbs” or “pbs2” alias so that the servers register to pbs_comm with the internal interface address;
 - `/etc/init.d/pbs` needs to be changed to supply a clientsfile to the scheduler (see `man pbs_sched`) using the `-c` file, in which either the hostname or the addresses that will be used by the primary and possibly secondary host as TPP leaf names are mentioned on `$clienthost` lines; alternatively, supply the leaf names as PBS_PRIMARY and PBS_SECONDARY but **only** in front of the scheduler process startup line in `/etc/init.d/pbs`, or make a separate `pbs.conf` file for the scheduler and supply its location using PBS_CONF_FILE only to the scheduler. Note that it is an error to supply only PBS_PRIMARY and not PBS_SECONDARY; if you do not have two servers just supply the same name for both.

and on execution nodes:

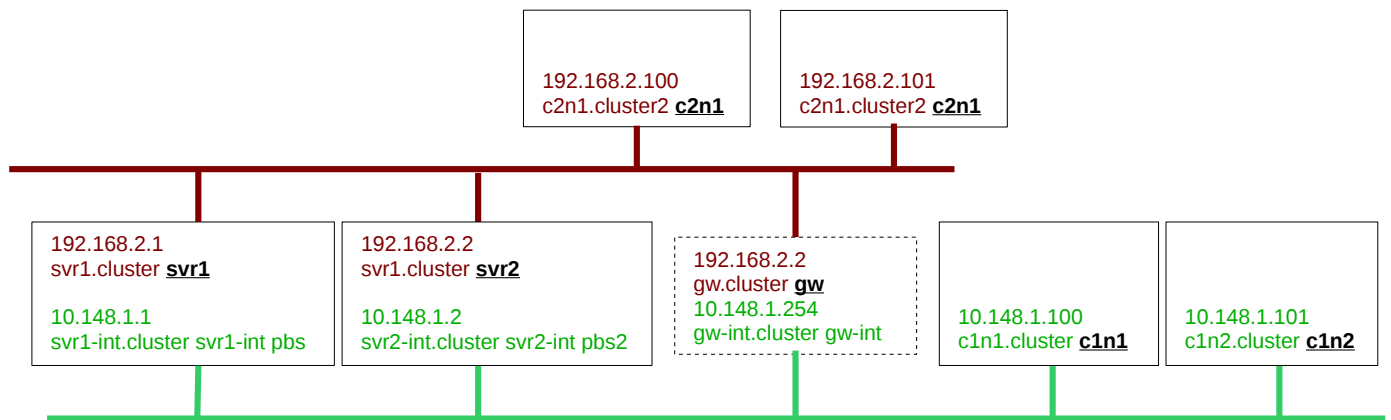
- PBS_SERVER needs to be set to “pbs” to be consistent across the cluster;
PBS_PRIMARY and PBS_SECONDARY can be set to pbs and pbs2 (or svr1-int.cluster and svr2-int.cluster).

iii. On execution nodes, PBS_LEAF_ROUTERS can be set to “svr1-int,svr2-int” or “pbs,pbs2” so that MoMs know to approach the pbs_comm from the internal network. In contrast to the two previous solutions client commands on execution nodes will still use the gateway (and fail if it is down or if it is a NAT gateway)

Solution ii cannot be implemented after the fact without *also changing the job suffix* (the last old job will be called <N>.svr1 and the first new job will be <N+1>.pbs). That is extremely messy if client commands are used that do not specify job suffixes and/or when job dependencies are used, and can only be fixed by updating the suffix of the old jobs in the data store.

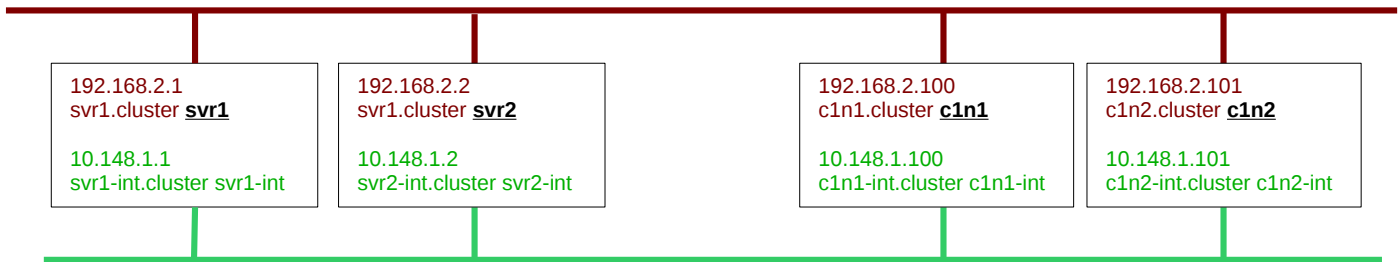
3. Multinetwork Cluster

Sometimes a situation as described in the previous section cannot be avoided, if e.g. there are execution nodes on more than one network connected to the server:



The same solutions used in the previous section apply here, except that the second solution cannot be used on its own (since the PBS leaf address for the server and PBS_SERVER cannot be on two networks at the same time).

4. Multihomed Execution Nodes with Hostname bound to “wrong” interface



In this situation the execution nodes also have a hostname bound to the “wrong” interface, on the network where you do not want messages between pbs_comm and MoM daemons to pass.

It should simply be possible to use PBS_LEAF_NAME to tell the node to register with pbs_comm using its internal IP address 10.148.1.100.

However, until PBS-13894 is fixed, it will introduce problems for multihost jobs (secondary nodes may refused to join the job).

In this situation, it is currently advisable to let the MoM daemons register the corresponding “external” IP address to pbs_comm and install host routes or use PBS_LEAF_ROUTERS on the execution nodes so that they contact pbs_comm through the internal network.