Debugging SystemVm agents

You can use Eclipse or any Java IDE to debug the cloudstack systemVM services as described below:

Debug a live agent

1. login into ssvm, either console proxy or ssh(port 3922)
2. kill all the processes named as(run.sh/_run.sh, and java)
3. cd /usr/local/cloud/systemvm
4. add parameters "-agentlib:jdwp=transport=dt_socket,server=y,suspend=n,address=*:8787" (or "-Xdebug -Xrunjdwp:transport=dt_socket,server=y,suspend=n,address=8787" for previous JDK versions) after "java " in the last line of _run.sh
5. ./_run.sh, the java agent will start, with debug port 8787 is listened on.
6. allow port 8787 in ssvm, "iptables -I INPUT -i eth1 -p tcp -m state --state NEW -m tcp --dport 8787 -j ACCEPT", either eth1 or eth2 is ok.

Then you can connect to agent in ssvm, based on the ip address based on on eth1 or eth2, and port 8787

Debug a mock agent

Or if the above steps sound too complicated, you can debug the systemvm resource locally using the cloudstack codebase:

For example, There is unit test code, called, LocalNfsSecondaryStorageResourceTest, you can mock the commands send to resource, then test it locally, but the requirement is that your local machine needs to be able to mount nfs(Both Linux/Mac works, Windows can't). I'd prefer unit test, as it's easier to test /debug/fix.