

OpenNebula - Feature #3409

Provide different message transport systems next to SSH

12/05/2014 03:42 PM - Arnold Bechtoldt

Status:	Pending	Start date:	12/05/2014
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:		Pull request:	
Resolution:			
Description			
<p>The TCP driven <i>overhead</i> given in SSH talking with a huge amount of compute nodes might be a bottleneck in the future. I think this can be a problem for users running very large ONE setups.</p> <p>One way to improve this is to use a message queue based concept like ZeroMQ, RabbitMQ and others. A great example for this is SaltStack which allows a user to choose between ZeroMQ, SSH and RAET (http://docs.saltstack.com/en/latest/topics/development/raet/index.html). Thanks to this concept, Salt is able to manage tens of thousands of clients.</p> <p>I can image having OpenNebula implementing such a transport system as an optional (!) alternative. User may also be able to use the infrastructure of SaltStack itself to send commands (deploy VM, move VM, undeploy VM, etc.) to compute nodes by creating new MADs. The missing part will be having OpenNebula handling the results of these commands and doing further processing (rollback, etc.). Also think of using SaltStack or message queue to integrate with oneflow and friends.</p> <p>Another benefit using SaltStack would be the access to the hundreds of execution modules developed by a very large open source community (http://docs.saltstack.com/en/latest/ref/modules/all/index.html).</p> <p>http://docs.saltstack.com/en/latest/ref/modules/all/salt.modules.cmdmod.html http://docs.saltstack.com/en/latest/ref/modules/all/salt.modules.virt.html#module-salt.modules.virt http://icee3.com/wp-content/uploads/2014/11/SaltStack-Thomas-Hatch-IC3.pdf http://docs.saltstack.com/en/latest/ref/clouds/all/salt.cloud.clouds.opennebula.html</p> <p>Looking forward to get your thoughts on this.</p>			