

Foreman - Bug #2270

Unable to create instances on OpenStack without floating IPs

03/01/2013 07:00 PM - Anonymous

Status: Closed	
Priority: Normal	
Assignee: Greg Sutcliffe	
Category: Host creation	
Target version: 1.5.0	
Difficulty:	Fixed in Releases:
Triaged:	Found in Releases:
Bugzilla link:	Red Hat JIRA:
Pull request:	
Description	
<p>Ohad, after our chat in the channel I cloned the develop branch (ca8e438b88260e9c1fd2847726bbb7d05d92ad18) and with that I'm able to create hosts as long as a floating IP is chosen. But without a floating IP the host creation process does not move past the "running - Acquiring IP address for testwithoutfloatip.ohmy.com" phase. It eventually times out and fails with "Unable to save. Failed to get IP for".</p> <p>From the logs: http://pastie.org/6363087</p> <p>The host is created on the OpenStack side though: http://pastie.org/6363089</p>	
Related issues:	
Related to Foreman - Bug #4616: Problems provisioning hosts in EC2 with 1.4.1-2	Closed 03/10/2014
Related to Foreman - Bug #4710: Problems provisioning hosts in Rackspace with...	Closed 03/19/2014
Has duplicate Foreman - Bug #3252: launching openstack VMs adds the external ...	Duplicate 10/15/2013

Associated revisions

Revision 1b110b37 - 02/19/2014 03:15 PM - Greg Sutcliffe

Fixes #2270 - Detect all OpenStack IPs and test which allows SSH

Revision eb1bba78 - 02/19/2014 08:06 PM - Greg Sutcliffe

Fixes #2270 - Detect all OpenStack IPs and test which allows SSH

(cherry picked from commit 1b110b37dd4a332ee0682e33daaaa848da8bcad7)

Revision 346b20ee - 04/10/2014 08:26 AM - Greg Sutcliffe

Fixes #4710, #2270 - Wait for VM to become ready before looking for IPs

History

#1 - 03/01/2013 07:17 PM - Anonymous

I forgot to mention that Foreman, along with the proxy, puppetmaster, and bind are running on a VM in the same private network as the testwithoutfloatip.ohmy.com host.

#2 - 03/04/2013 03:01 AM - Mateusz Kisielewski

Same problem here:

```
Failed to get IP for ubuntu006.<domain_name>: The specified wait_for timeout (600 seconds) was exceeded/usr/lib/ruby/gems/1.8/gems/fog-1.9.0/lib/fog/core/wait_for.rb:10:in `wait_for'
```

...

```
Rolling back due to a problem: Acquiring IP address for ubuntu006.<domain_name> 2 failed ubuntu006.<domain_name> setComputeIP  
Removing Compute instance for ubuntu006.<domain_name>
```

```
Failed to destroy a compute Openstack (OpenStack) instance ubuntu006.<domain_name>:
```

```
ActiveRecord::RecordNotFound/usr/share/foreman/app/models/compute_resource.rb:104:in `find_vm_by_uuid'
```

...

```
Failed to save: Failed to get IP for ubuntu006.<domain_name>: The specified wait_for timeout (600 seconds) was exceeded,  
Failed to destroy a compute Openstack (OpenStack) instance ubuntu006.<domain_name>: ActiveRecord::RecordNotFound
```

Greetings,
SCR

#3 - 03/04/2013 03:03 AM - Ohad Levy

- Priority changed from High to Normal

what would be the right behavior then? go over all floating ips and fixed ip to try to connect?

#4 - 03/04/2013 05:16 PM - Anonymous

That's a good question. How about Foreman doesn't use a floating IP to communicate with a host unless explicitly instructed to do so (even though the user has selected a floating IP network from the drop down list)? Perhaps with a "Manage this host using its floating IP?" checkbox during the host creation phase.

In a typical OpenStack cluster the fixed address pool is larger than the floating IP one. It's probably reasonable to assume that most hosts will communicate with other hosts using their fixed IPs on the internal network -- for provisioning, monitoring, log collection, DNS, etc. A smaller subset of hosts will have floating IPs attached if they have to provide services to the outside world but those are the exceptions and will still use their fixed IPs.

Thoughts?

#5 - 03/04/2013 05:23 PM - Sam Kottler

The only problem with not using the floating IP is that the VM may not be on a network that's accessible to Foreman. Maybe we could allow the user to choose either globally or on a host by host basis whether to connect to the internal address or floating IP?

#6 - 03/04/2013 05:30 PM - Anonymous

Sam Kottler wrote:

Maybe we could allow the user to choose either globally or on a host by host basis whether to connect to the internal address or floating IP?

I can't comment on how granular the config option should be but I do feel it's reasonable to allow the user to state a preference, either globally or per host, for using the fixed or floating IP. Doing so allows varying deployments of Foreman/OpenStack be catered for.

#7 - 03/11/2013 11:48 AM - Ohad Levy

sounds like we have a few options (the same solution is probably true to all cloud providers)

1. use all ip addresses of the system to try and find it
2. let the user select the network per compute resource (not makes a whole lot of sense per vm imho)

another option which i have not yet thought much of, is using similar idea to multiple nics on Bare metal? and mapping those networks to something foreman can consume?

#8 - 03/11/2013 02:28 PM - Anonymous

Being able to specify the network for each compute resource makes sense.

Just to clarify, being able to allocate a floating IP to an instance is still a great feature to have in Foreman even if the private address is being used for managing it. Otherwise one would need to use external tools (nova CLI, for example) to assign a floating IP to an instance that Foreman created. Currently that's what I use to attach Cinder volumes to instances which means Foreman isn't used entirely for the provisioning process.

#9 - 07/11/2013 02:42 AM - Cédric Jeanneret

Hello,

We're hitting the very same problem in our infra using foreman 1.2. Just one (probably) stupid question, but why can't foreman ask the IP directly to openstack? Openstack API is well formed, and can return the wanted information.

Some details: when we create a new instance (using nova commands), we get the instance id, like "90fe2237-fab3-4519-b4be-b64ca8a358ee" - this can be used later in order to get the dynamic IP provisioned by openstack... This nova command may be of some use in order to find how to ask it I guess:

```
nova interface-list <hostname|instance_id>
```

This will return:

```
+-----+-----+-----+-----+-----+-----+-----+-----+
| Port State | Port ID | Net ID | IP addresses | MA
C Address |
```

ACTIVE	8e8958ba-b5ff-4373-9324-26b862d352e4	5b67e329-266c-4553-a6f8-fb7793cb5454	10.27.70.16
--------	--------------------------------------	--------------------------------------	-------------

There's no "we have to search the whole subnet/network" stuff... Just use the openstack restful API.

NOTE: the nova interface-list will not return floating IP attached to the host

This is a major problem I think, as we cannot provision instances with non-floating IP directly from Foreman - we have to start it in Openstack, then do all the foreman stuff by hand in order to make the instance known... not really suitable I think.

Cheers,

C.

#10 - 10/15/2013 10:37 AM - Greg Sutcliffe

- Has duplicate Bug #3252: launching openstack VMs adds the external network added

#11 - 10/21/2013 04:21 PM - Andy Taylor

As we've started using Openstack we've run into the same issue.

From our perspective the simplest fix would be a setting in the Compute Resource - whether to use the fixed or floating IP of the VM. That way you can always have multiple Compute Resources if you want some VMs to have fixed, some floating.

Also I think automatic determination of the VM IP addresses could end up being fairly messy. Foreman would be connecting to some VMs via the fixed and some via the floating.

#12 - 11/22/2013 09:52 AM - cristian falcas

I would prefer to choose which network is for foreman. In cases with multiple private networks per vm, probably only one will be for foreman management.

Also, the fix for our case (flat network, so no floating ip) is very simple. In /usr/share/foreman/app/models/compute_resources/foreman/model/openstack.rb change:
 super.merge({ :ip => :floating_ip_address })
 to
 super.merge({ :ip => :private_ip_address })

Sucks if you have other compute resources where foreman is accessing the machines only through the external network.

#13 - 01/14/2014 05:33 PM - Greg Sutcliffe

- Description updated
- Status changed from New to Assigned
- Assignee changed from Ohad Levy to Greg Sutcliffe

<https://github.com/foreman/foreman/pull/1149>

#14 - 01/15/2014 08:57 AM - Dominic Cleal

- Status changed from Assigned to Ready For Testing
- Target version set to 1.9.2

#15 - 02/06/2014 12:12 PM - Anonymous

- Target version changed from 1.9.2 to 1.9.1

#16 - 02/19/2014 03:15 PM - Dominic Cleal

- translation missing: en.field_release set to 5

#17 - 02/19/2014 03:31 PM - Greg Sutcliffe

- % Done changed from 0 to 100
- Status changed from Ready For Testing to Closed

Applied in changeset [1b110b37dd4a332ee0682e33daaaa848da8bcad7](#).

#18 - 03/10/2014 03:21 PM - Dominic Cleal

- Related to Bug #4616: Problems provisioning hosts in EC2 with 1.4.1-2 added

#19 - 03/24/2014 08:39 AM - Dominic Cleal

- translation missing: en.field_release changed from 5 to 4

We've had a couple of follow-on issues ([#4616](#), [#4710](#)) which means I'm reverting this fix in Foreman 1.4.2. It seems the code actually had errors which meant it probably didn't work. We're working on updating this for Foreman 1.5.0 via [#4710](#) and to ensure we don't break other compute resource providers.

#20 - 04/09/2014 10:12 AM - Dominic Cleal

- Related to Bug #4710: Problems provisioning hosts in Rackspace with 1.4.1-2 added