

Foreman - Bug #7031

Provisioning to EC2 is not working and not logging

08/11/2014 07:01 PM - Ashton Davis

Status: Closed	
Priority: Normal	
Assignee:	
Category: Compute resources - EC2	
Target version:	
Difficulty:	Fixed in Releases:
Triaged:	Found in Releases: 1.5.2
Bugzilla link:	Red Hat JIRA:
Pull request:	
Description	
<p>I've set up an EC2 (VPC) compute resource, and assigned an image, userdata file, etc - but it won't provision. See this Foreman-Users post for reference https://groups.google.com/forum/#!searchin/foreman-users/ec2/foreman-users/KeVB6fGaMUk/1N9uD0Ap0Y8J</p> <p>That's the exact same error I have. The keys I'm using in the resource are full-access keys, the AMI is accessible, they're all in the same region - it's failing on SOMETHING, but not logging what.</p> <p>I did also follow Lukas' advice and enabled REST debugging - no additional output.</p> <p>This is a blocker for progress at the moment - any help would be most appreciated. Here are my package versions:</p> <pre>foreman-libvirt-1.5.2-1.el6.noarch ruby193-rubygem-foreman_bootdisk-2.0.5-1.el6.noarch ruby193-rubygem-foreman_setup-2.0.3-1.el6.noarch foreman-release-1.5.0-0.2.RC2.el6.noarch rubygem-foreman_api-0.1.11-1.el6.noarch foreman-installer-1.5.0-0.2.RC2.el6.noarch foreman-1.5.2-1.el6.noarch foreman-vmware-1.5.2-1.el6.noarch foreman-postgresql-1.5.2-1.el6.noarch foreman-proxy-1.5.2-1.el6.noarch ruby193-rubygem-foreman_xen-0.0.1-1.el6.noarch foreman-compute-1.5.2-1.el6.noarch foreman-console-1.5.2-1.el6.noarch rubygem-hammer_cli_foreman-0.1.1-1.el6.noarch</pre>	

History

#1 - 08/11/2014 07:04 PM - Anonymous

- Category set to Compute resources - EC2

#2 - 08/18/2014 11:42 AM - Ashton Davis

I was able to get this fixed last week - after I got debug working, I was able to get a Fog output about a missing key.

Per the linked Google Groups chain, Lukas and Ohad think that Foreman should check for keys before using them. Perhaps handling fog errors a bit better would help as well - currently they don't seem to appear under any type of header in the log output - like they're just being appended to the output from stdin. That's probably why it doesn't get reported anywhere. I ended up encountering a second silent failure - a security group had been renamed, and it killed Foreman's ability to provision (since it was a default SG for the compute profile).

```
--
CACHE (0.0ms) SELECT puppetclass_id FROM "hostgroup_classes" WHERE "hostgroup_classes"."hostgroup_id" IN (3
)
CACHE (0.0ms) SELECT puppetclass_id FROM "host_classes" WHERE "host_classes"."host_id" = 74
Puppetclass Load (1.0ms) SELECT DISTINCT "puppetclasses".* FROM "puppetclasses" INNER JOIN "environment_cla
sses" ON "puppetclasses"."id" = "environment_classes"."puppetclass_id" WHERE "environment_classes"."environmen
t_id" = 1 AND "puppetclasses"."id" IN (36, 28, 29, 30, 23, 10) ORDER BY puppetclasses.name
Rendered text template (0.0ms)
Completed 200 OK in 176ms (Views: 4.9ms | ActiveRecord: 46.0ms)
```

Fog error: The security group 'sg-[number]' does not exist in VPC 'vpc-[number]'

```
/opt/rh/ruby193/root/usr/share/gems/gems/excon-0.32.1/lib/excon/middlewares/expects.rb:10:in `response_call'  
/opt/rh/ruby193/root/usr/share/gems/gems/excon-0.32.1/lib/excon/middlewares/response_parser.rb:26:in `response_call'  
/opt/rh/ruby193/root/usr/share/gems/gems/excon-0.32.1/lib/excon/connection.rb:398:in `response'  
/opt/rh/ruby193/root/usr/share/gems/gems/excon-0.32.1/lib/excon/connection.rb:268:in `request'  
/opt/rh/ruby193/root/usr/share/gems/gems/fog-1.21.0/lib/fog/xml/sax_parser_connection.rb:36:in `request'  
/opt/rh/ruby193/root/usr/share/gems/gems/fog-1.21.0/lib/fog/xml.rb:22:in `request'  
/opt/rh/ruby193/root/usr/share/gems/gems/fog-1.21.0/lib/fog/aws/compute.rb:447:in `_request'  
/opt/rh/ruby193/root/usr/share/gems/gems/fog-1.21.0/lib/fog/aws/compute.rb:442:in `request'  
/opt/rh/ruby193/root/usr/share/gems/gems/fog-1.21.0/lib/fog/aws/requests/compute/run_instances.rb:140:in `run_instances'  
/opt/rh/ruby193/root/usr/share/gems/gems/fog-1.21.0/lib/fog/aws/models/compute/server.rb:195:in `save'  
/opt/rh/ruby193/root/usr/share/gems/gems/fog-core-1.21.1/lib/fog/core/collection.rb:51:in `create'  
/usr/share/foreman/app/models/compute_resource.rb:150:in `create_vm'  
/usr/share/foreman/app/models/compute_resources/foreman/model/ec2.rb:48:in `create_vm'  
/usr/share/foreman/app/models/concerns/orchestration/compute.rb:74:in `setCompute'
```

#3 - 08/18/2014 12:02 PM - Dominic Cleal

I think Foreman 1.6 fixes the hidden errors via [#6099](#), it should now show Fog errors (esp around EC2) on the new host page.

#4 - 10/24/2015 10:01 AM - Anonymous

- Status changed from New to Closed

closing this as [#6099](#) will log errors now.